NLN LLC

2nd Edition

Visual Basic

A Step-by-Step Guide to Learn, in an Easy Way the Fundamentals of Visual Basic Programming Languag

By Emma William & Ariadna Moore & Rufus Stewart

Visual Basic

A Step-by-Step Guide to Learn, in an Easy Way, the Fundamentals of Visual Basic Programming Language

2nd Edition

2021

By

Emma William

For information contact :
(alabamamond@gmail.com, memlnc)
http://www.memlnc.com

2nd Edition: 2021

Learn Visual Basic

***

memlnc

Content

As is led s

Visual Basic . NET - is a powerful new language in programming mode of Bani, created to rp of rac iey Microsoft at the beginning of the millennium. Save as many apparent simplicity of its predecessor - the language of the Visual Basic 6.0 - it dramatically on a Gatilov and complicated with its domestic holding, giving the programmer a tremendous about lichestvo brand new, brand-new, tools. The most important of them - .now, for l fledged object-landmark of Bathing programming

Visual Basic . NET course is still very strong in all areas of tr and tional programming for conventional desktop computers , ie there, where there was strong, and the Visual Basic 6.0 . But , responding to the call of the times, he has acquired and the new district and the Management Board - the programming networks . Now, about Mr. allows convenient and easy to write programs for local area networks and the Internet , to PDAs and even mobile .phones of the new

Visual Basic . NET - is productive and reliable n th weapon in the hands of professiona about -regional programmer. In addition, a very comfortable and despite the power, enough, etc. on -empty . P oetomu the Visual Basic . NET more other professional language suited to have mastered e Nia beginning programmers. It takes a minimum of time to write a program . After reading this SOI g y, you can quickly write such , for example, applications as are proper e calculator or player MP 3 -files, shifrovschik secret wo b scheny or Service , a web browser or a simple little game and .... With a mustache is sufficient and Leah, you can create quite solid products, such , for example, as an ASIC theme of sound control in Lenia databases , editing or complex , beautiful game that is not ashamed to publish in and n .ternete

In what follows, for brevity, I will often refer to Visual Basic . NET to aka **VB** . John of the GDS , however, I will write *the Visual Basic* or even . just *BASIC*

?Who is this book for

: **The book is designed for two categories of readers**

- **For novice programmers, including even those readers who have not programmed a single line in their lives and just want to learn how to program. Not only that - in the end of the book you will find two applications that will prepare to Play and yatiyu .books, even those who have never sat down at the computer**
- **For programmers with intermediate programming experience who want to master a good object-oriented language and learn what object-oriented programming is . They will get what they wanted - they will feel what objects " at their fingertips" mean . But they may yet to show Xia, I am too, "Dr. of zhevyvayu" material . Well, I confess : I am always of the opinion that it is ."better, "the change of notation b yasnit" than "nedoobyasnit**

This is a tutorial . The book is written with the expectation that, having studied it without any outside help , you will be able, without outside help , to maintain programs on VB and run them on a computer also without outside help . To put advertising lang s com: " The Behold, you need to know in advance, it is where there is a power button Compute e p as ! " . Everything else in the book is explained , including installation instructions VB on your Comp w ter , to work in Windows and on entering the program text into a . computer

The author has made special efforts to make the presentation clear. Wherever possible, I have tried to explain not only the fact that the computer does, but also why he does it. Everything is explained with examples. Consideration of almost every topic ends with tasks for drawing up a .(program (there are 146 such tasks in total

The book is full , that is, despite the fact that it caters to beginners, etc. of programmers, it completely contains traditional material, it is presented in textbooks for VB such as an Kommersant EMA. You can verify .this by looking at the table of contents

P After mastering simple reader training program book goes on and so forth , and as you create real projects explains the method of writing complex

x , salt etc. us x programs which give the reader a sense of power and . confidence in the St. of their power

Examples and tasks

When creating the book, it was very important to decide which examples to choose, on the basis of which to build the presentation of the material. It operates harsh rule: bad note e ra - a bad book, good examples - good book, I & C h ex- amples - boring book, interesting examples - an .interesting book

All the examples analyzed in this book give solid, solid knowledge, suitable for both the pupil and the student. But about one-third of these examples, I have tried to make of a We recommend that you. Here are some :of them

P The examples of tasks on programming decisions in the : book

- **The game " Auto racing " , where you control unpretentious car**
- **Game " Catcher " , where you have to catch a lot of moving balls**
- **Guess the number game**
- **Objective: To ak put a password on your program, so that nobody except you could not start it**
- **Objective: To ak program saving games**
- **Objective: To go to someone else's computer (pre-ak Knock (online in shis**
- **Objective: To ak the Internet to place on your page game**
- **Own calculator (if desired - from your photos, music, (skipping and E keys and other jokes**
- **Own stopwatch alarm**
- **.Own player for in Playback I musics and and video**
- **Own cartoon**
- **Own graphic editor**

- **Drawing a variety of designs and patterns (circles on the (.water, starry sky, bubbles, co m pact-disc tower, carpet, etc**
- **Placement of menus, buttons and other controls in Windows applications**
- **Encrypting and decrypting secret files (explained simply shi (f operated and decryption**
- **Meaningful" dialogue with the computer"**
- **Determinant of the speed of reaction of you and your friends**
- **" The program defines " Are you a psychic**

II Examples for tasks The offered x for the independent : solutions

- **And zgotov Lenie self th CD and with your favorite songs**
- **" And gra-shooter " Torpedo Attack**
- **(Tic-tac-toe on an endless field (five in a line**
- **Sea battle game**
- **The game "Tank Battle" on a stationary field (top view), where expected simultaneous e Menno move and shoot the little figures, and your enemy is the n Cove**

If you are an enthusiast of game programming , I must tell you about what the game will be beyond the power to create a beginner programs istu after leaving at cheniya only my book. First of all, these are three-dimensional (3 D) games like *Quake* or *Unreal* . Ucht and those that create these games are not novices and professional programmers , and one hour ku, but entire businesses. And they are used for that special software (eg as an example, libraries in Direct X) , to find out which can only be eaten pud salt in geometry etc. of programming. Study of a specific programming language - it's just the first step in the creation of solid products, such as an n naya 3 D -game. So it is, If your friend a week mastered are editing tion scenarios s game *Heroes III* and now the important states that he programs, .you can quietly SMILE and Yas, ck and to show him that he was a bit wrong

But what and g ry, you can create: *Gloomy city street. Appears in the car. He is rapidly approaching the main character, hiding behind an advertising curbstone with a blister in his hands. Shots are heard from the car. Bullets shte l mess on the pedestal. At the last moment, brakes screech and the car freezes. T e per tarry not - beat on the keyboard keys, shoot until narco godfathers of the mafia before you did not get! Car breaks out of it popping " lads " and saved as Yas from your shots to catch the next shelter, ... where opening and are harden n ny fire. The fight continues*

And from that you can actually do in the described game : *on the screen - picture g of sorts, no matter how beautiful, you draw yourself or anywhere d of bytaya. You can , by the way, following the plot of the game , to change the image of streets, quarts and fishing, putting e Nij at the right time by placing the screen on the right picture . Cars move around the city, passers-by, bullets fly, laser beams flash. You will be able to e they create the character's voice, the sound of gunfire and more. However, movement of persons and ments will not be as realistic as 3 D -game. You will be easy to do, Thu to be characters in the figure of a styvshey position moved in any direction you want, and even with this decrease in size (to run away) or increased (approaching and Yas). It will be a little more difficult to get them to move their legs while doing this. B olshe time it takes to program more complex movements, such as turning sideways, etc. , and to bend, lie down, charge the rifle, to smile, to get into the pocket, and the like. If the hero pleases a long walk to the right, you can make a city helpfully scroll to the .left, as is the case in, and Mr. rah-arcades*

If you think that starting to learn programming not with VB , but with Delphi or C ++, you will come to programming more complex games and other tasks faster and easier , then you are deeply mistaken. If you are asked for advice on where to begin, to the profession of tional programs istu to about tory whole life to write games and other programs in C ++, then the answer would almost certainly have heard: "Start with the BASIC , etc. have "!Joc

Briefly about the .NET Framework and for Visual Studio .NET

Visual Basic . NET is not a separate standalone development. He CREATE and was located in the part of the new ideology proposed by the Microsoft . This ideology was named **. NET** (exactly like that - with the preceding dot). The essence of ideology is that programming is gradually shifting from individual to go COMP th tera network (NET in English - "Network") and the Internet, so you need to focus on programming in the networks, the Internet . About tsyuda also implies that special Watch yo and of the need to draw on without a pasnost s program delivery. As conceived by its creators, ideology. NET should in s lead the information world to a new level of using computers, networks, the Internet and the development .prospect of the gram

Implementing the ideology of .NET, Microsoft has developed a set of software tools for its implementation. This comprehensive range of made-called s Vat **platform .NET** . It is assumed that this platform will be .supported not only by Windows , but also by other operating systems

An important part of the .NET platform is the **.NET Framework** . It is designed specifically for the development prospect of grams and to support their implementation on a computer e . You cannot program in VB if the . .NET Framework is not installed on the computer

:The .NET Framework has two main parts

- **NET Framework class library . For a beginner I would say that if your prospect of a gram - a house, then the classes - it blocks out of which a significant part of the house is built. E quietly classes in the library contains tons of camping a lot, enough .to build any prospect of a gram**
- **Common Language Runtime (CLR). It is part of the .NET Framework , which manages the execution of your program to m pewter and ensures reliability and bezop with governing this implementation. You can not perform on another computer program created on the VB , if it is not the mouth of a Credited the CLR . Microsoft is going to include .NET Framework in a future version of the Wi n dows , so that soon problems with performance, etc. of the gram created on the VB , on the .computers where the CLR is not , should disappear**

To Programming be for the platform .NET , necessary tool s. The company Microsoft has developed as these tools are a series of closely related languages , etc. for programming . Here they are:

- **Vi s ual Basic . NET**
- **Visual C ++ . NET**
- **Visual C # . NET**
- **Visual J # . NET**

They are all included in a single program called **Visual Studio . NET** and is sold on multiple discs . This program includes in Kutch e stve his inseparable e my parts and a common development environment for all these languages and common aids. This is convenient and should facilitate ease of retraining from language to language. And Mr. stallyatsionny package of the Visual Studio . NET includes the .NET Framework and all other required system programs. T and Kim, the setting on your computer the Visual Studio . NET , you can Programmable about Vat on any of the mentioned languages and even on all four simultaneous n but.

? ++ Why Visual Basic and not C

Why Visual Basic ? This is the question for beginners who don't know how to start with a language. Here is a brief overview of the possible in a Rianta.

Now the most famous languages вЂ ‹вЂ‹ are BASIC , Pascal, C, Java (Java) Before the appearance of the same names . NET languages, they all existed and now exist in their very different versions. ... NET languages differ from their predecessor namesake , even though they are their direct descendants . They differ not so much in the appearance of the programs as in the internal content (using the mechanisms of the .NET Framework). Visual B a sic stands out especially sharply . NET .

As for the time lichy that existed between themselves tongues, they mostly remained in their . NET - versions .

Consider how a particular language for the study of beginners, etc. of programmers.

Let's start with Java. Java is mostly used on the Internet so far. Since we

are still interested in ordinary programming, we move Java aside. There are . BASIC , Pa s kal, Si

C - a highly professional language programs written in it, have excellent performance, among programmers C Prevalence and nen very widely . N on it too complicated for novice perception , programs on it are written with considerable difficulty and it is better not to start. If you really really want to progra m grammed in C, I recommend to start with BASIC . After n its .development Xi go mountain of up easier

According to professionals, BASIC is the easiest and simplest of the languages mentioned . Professional programmers will love it for the fact that for CREATE and Nia on it, etc. , and personally of about THE PRODUCT requires much less time than , say, C . Therefore, his popularity is enormous. Of course, for all we have to pay, and the Visual Basic has its Insufficient ok - programs written on it , not as fast s e as in C , but a beginner will feel the .difference is not very ck of ro

Pascal for simplicity programs ation and effectively get th schihsya programs occupies an intermediate position between the BASIC th and Xi . Pascal to have exists right now in the form of the modern version of the Object Pascal and programs ation it is conducted in an environment Delphi . . But Pascal is not included with Visual Studio . NET

.Do you Bor I leave to the reader

Why Visual Basic . NET , and not old the ? Visual Basic

Why it is necessary to study precisely the Visual Basic . NET , and not another version of the Visual Basic , n as an example, the latest version of the ? pre NET ovsky era - the Visual Basic 6.0

:Consider the pros and cons

Pros : Because this is the latest and most powerful version of Visual Basic . She predosta to wish to set up many features not available in previous .ve p Siyah

For : Because this is the first truly object-oriented ve r , This the Visual .++ Basic . In this it is catching up with Delphi and C

Pros : Old Visual Basic will be in use for a few more years, then it will

.probably go away

Cons : Visual Basic . NET is more complex than the old Visual Basic . .Therefore, beginners in easy out at the chat old

.The choice is yours

You ask, p hy I wrote the book is on the Visual Basic . NET ? - Because of the SOI Well ku on the Visual Basic 6.0 I wrote earlier . You can find it at . www . learncomp . narod . ru

Differences between Visual Studio .NET 200 5 from 2003 and from Visual Studio . NET

Differences are virtually absent and reduced mainly to the interface , so this book is a primer on the avenue of programming language Visu al Basic . NET as in Visual Studio . NET 200 3 and in the Visual Studio environment . NET 2005 and Visual Studio . NET . Although the description is specifically . for the 2003 version

As s you nuzh us the computer and the ? Windows

Visual Studio . NET makes quite high demands on the computer and op e ration of the system. Full functionality s programming Nosta could ut ensure only the Windows Vista , the Windows 2000, the Windows XP Profe s sional and the Windows Server 2003 . To install Visual Studio . NET You .need a few gigabytes of free disk on the first prospect of space

Summary with recommendations

: The book consists of three parts and two n reel of stress th

. Part I . Programming without Program ation

:This part has two purposes

- **To teach the most basic and elementary techniques of**

. working in VB

- **Show that in VB you can achieve impressive results pra to .matically without any Programming and Niya**

Proof of this is the projects "Calculator" and "Player", and the "Calculator" will turn out to be much brighter and more fun than the standard . Wi n do ws calculator

Part II . VB programming is the first level. Here begins the present I present programming. The purpose of this part - to take you from the foundation of the easiest prospect of grams to the complex. Here you learn Programming of on VB the most convenient th FPIC of bong - with examples, that is, on the principle of " do as I do " . You will learn to create Neboli s Chiyah prospect of grams, including loops, branching procedures and uses graphics and sound. But the matter is not limited to simple programs. In this part of the p Detailed Info dil and raetsya creation of three pretty solid projects. The part ends with a task for the independent creation of another such project . It is assumed that after the in s complements this mission you must be a feeling of omnipotence, you should Pochuev to update themselves, now you on the shoulder, a program of any size, and everything you might need in the future - this is additional information about working . with a whole wealth of tools n taria VB

Part III . VB programming is the second level. Destination s this part - with at b dit you these same information about the tools the VB . You will meet dis t viyami over arrays Collection in E strings, files , and other objects Elem n Tammy VB . You will learn the procedures and functions with parameters, modules, know what recursion is and with the p tran sportation. There is also work on the Internet, databases , and, of course, create your own .classes of Kommersant OBJECTS , properties, methods and events

Appendix 1 . This is a theoretical educational program. Designed for . those who do not know how the computer is arranged and works

Appendix 2 . This is a practical educational program. Designed for those who have never been to a dilsya for the computer or sit only for games. Here you learn the slave of a thief with windows, folders and files the Windows , enter into computer text, ie acquire all neo b walk skills to calm . Mr. and chat work among the Visual Studio . NET

Not knowing the material application, you will be in VB stumble on ka .Well house step

Then, in the book drive I tsya solutions to the solid and the alphabet
.pleasing decree and Tel

From the author

I would like to express my gratitude to my sons: Alexei - for a valuable discussion of theoretical aspects of programming; Leonid, whose need to advise on the game The Siege he was creating prompted me to study objects in more depth. But the biggest thanks - my wife, Lyuba, which weighi and Lila the whole load of the household and takes it for many years, watching as
. I enjoy sitting at the computer and working on this book

To the reader

I would be glad to hear your feedback on the book. Especially as I will be grateful for the critical e skie comments. My e - mail : lukin 63 @ mail . ru
.

Part I . Programming and of the free programs Irova and Nia

:This part has two purposes

- **To teach the most basic and elementary techniques of working in VB .**
- **Show that in VB you can achieve impressive results pra to matically without any Programming and Niya.**

Proof of this is the projects "Calculator" and "Player", and "Calculator" will turn out to be much brighter and more fun than the standard Wi n dows calculator .

This part of the need to ensure that, after reading it, you could say to yourself: " I am quite SALT about ilsya and can do in VB interesting things, not especially strained and Yas " . That is (I will explain) almost without programming. Almost, but not absolutely. Simple program text still having come so camping write.

In Chapter 1, you will create your first VB project .

In Chapter 2, you will dive into the depths of the .NET Framework libraries and classes . After all, in order to build a house from classes of cubes, you need to be able to find the necessary cubes and put them in place.

The 3 chapter, you will create your own calculator, providing it with a photo, musical 's Coy and even video. The purpose of the head - to open before you the basic simple and simultaneous n but bright opportunities VB , and show that to create simple projects really le d to. At the end of this chapter, you will also create your own player.

The part ends with Chapter 4 “ Working in Visual Studio . NET » , which sets out the main at e we work on a computer is, for programming the VB .

I tried to write a book like this to sit down at the computer and software th e to VB might even those who never in his life for the computer did not sit down. Especially for such a person I have to write an application I 1 and 2 in

which s explain all elementary and qs on sary to do things: how the computer , what is the structure of folders on the drive as a slave on the thief with the text in a text editor, how to cope with the windows Windows . If you chuvs t exists gaps in at least one of these areas, then in any case do not get behind to . m pewter and stop reading, but first read Ap e or i

By the way, you read Introduced s ? It contains the required further .Therminol of Guia

Chapter 1. First steps

In this chapter, you will create and will try to use the computer is, my .very first progra m th on the VB . For this, the chapter was written

? What is a VB program .1.1

What is a program ? From the point of view of VB and most other object-oriented programming languages , it is most convenient to consider by analogy. Imagine that your sign of mine, who has never left his village, has come to visit you, living in a big city . He wants to go to football, but you have no time to go with him. So that he could get to the stadium and back alive and healthy fo about vym, you write it on a piece of paper so ins t .ruktsiyu

What to do if you feel like going to football

1. Take the elevator down to the courtyard
2. Get to the subway
3. "Before beris station "Sport
4. Buy a ticket to the Luzhniki stadium
5. Go to the stands and watch football
6. Return by metro to Otradnoye station
7. Come to our house and entrance
8. Take the elevator
9. Ring the doorbell

How to take the elevator down

1. Come to the elevator and press the button
2. When the door opens, check if there is a booth
3. If there is, come in
4. Press the button with number 1
5. When the door opens, come out

How to reach forth to the subway

1. Turn left and walk to the corner
2. Cross the street
3. .. ……

How to get to the Sportivnaya station

1. .. ……
2. ... ……

How to cross the street

1. Find the transition
2. Look at the traffic light
3. .. ……

.. ……

What to do if an elevator gets stuck

1. Click on the dispatcher call button
2. .. ……

What if t s get lost Xia

1. Ask passers-by where a pay phone is nearby
2. Call home

.. ……

As you can see, there are several instructions on the sheet . They are of two types. Some early and nayutsya with the words " What if ... " , the other - . " ... with the words " How to

The uppermost primary instruction consists of 9 teams and prescribes strict n of a row of action to achieve the goal. The instructions begin with the word " How " , describing s vayut each of these actions more undermining used but. Thus, the instruction " How to get to the subway ," describes in detail the execution to about Manda " Reaching the metro " . Since this instruction is found team " across the street " , which is itself in need of .explanation, they e etsya guide " How to navigate Street and tsu " . Etc

Why did I write so many " How " instructions ? Do not it be easier to n and write one long main instructions of the " couple thousand " command, which would be asked by smacking d ku all small steps from the beginning to the end of the campaign, Mr. and starting with " Come to the elevator and press the button " and ending with " Come to the door of our apartment and call " ? - It is possible and easier, but that's the instruction in this case, get familiarize be too long. Why long? P about the fact that, for example, to cross the street will have eight times, and it turns out that in the instru to tion Come t Xia eight times to write the same explanation, how to do it. And the reason it is not necessary to write long instruction that a person is much more .pleasant and convenient to take short-ins so ruktsii than long

The order of execution of commands in the instruction s is strictly defined. Try the first ones to break it and see what happens . For example, first try to go to the stands, and only then buy a ticket. Or first, go to the door .of the elevator and then checks s ones if there cabin

But life is complicated, and it can occur in *the event* that it is difficult to anchor me to show to any particular stage of the statement's execution. For example, you can Zabol in ditsya (and, in any place, both on the way there and on the way back), or your elevator is stuck. In that case, the instructions are written, starting with the words "What d f lat if ..." and prescribing how .you react on the Vat on this or that event

In VB programming, all of these instructions are called **procedures** . The commands from which they are composed are called **operators** . The whole set of instructions on the sheet will be called a **program** . And about . life and will be called s Vat **events**

The program should not contain commands that the executor of the program is unable to execute. For example, "Fly over the street." When it comes to feasible about stey h e Lovek, there is more or less clear. What if we are writing a program for a computer? We need Che t to know that the computer is able to, and that he can not. Best of all, if we face is a directory that lists all the commands that the computer FPIC about bin execute. And this reference is - is a system of *the Help* , which you can uses about vatsya, . programs and Rui on the VB

But you will be surprised when you see how “weak”, “shallow”, “weak” all the commands that a computer can do. Whatever language you program in, nowhere else will you find such gorgeous commands as "Calculate the trajectory of a flight to the moon" or "Win chess at Kasparov." In any language you can give only a very primitive in nye commands such as "Multiply 5 by 4" or "Draw a circle on the screen." But you know, that the computer for a long time and calculated the trajectory of the moon , and beat Kasparov! What's the magic here? There is no magic, there are wizards. - Comes gloomy and programmer of the primitive commands n programming language and Sheth long program that calculates the trajectory of the moon , .or even longer program wins the s vayuschuyu Kasparov

Based on the above, we can conclude that programming knowledge :consists of two areas

- **Knowledge of the list of commands from which programs are composed**
- **At IU of rules on write commands one after the other so as .to obtain but the p -formal, etc. of a gram**

Not a program, but a project .1.2

Visual Studio . NET belongs to the so-called **visual development environments** . This means that it takes over a significant portion of the boring command work, leaving the programmer to "build a house of blocks" .instead

To clarify, consider an analogy. Imagine that you are not the programs have on VB NSP is, those and bring to the table toy railway. At your disposal are so camping **on the b JECTS** : table, rails, locomotives, cars, traffic lights, machinists, scapegoats, etc. To the toy to work, you need to run 3 :floors and pas

1. **Assemble it by hand, that is, place it in its place and correctly .connect and thread rails, wagons, etc**
2. **H asa toy very intelligence from -trivial, that is, the behavior of many objects (such as machinists, scapegoats, traffic lights, etc.) - is complex th and implemented a camping on the prospect of grams of e. Therefore, you need to write for each of these facilities programs in CEMs e denia. For example , in the program is, for the driver, one of the commands might be: " E to whether ahead at the traffic lights red light - brake " , and in the program is, for traffic - this - " If two trains closer dangerous - . light the edge of a ny light " and so on**
3. **Launch the toy and see how it works. If there was Cruz e of or anything else went wrong, then or you wrong with a take a toy , or wrote the wrong program s , or both together . You fix bugs there and there and start the toy again. Again, look, explained about the fix Islands. And so on until everything works fine. This . process is called debugging**

Modern programs ist, working in visual media bit and Botko programs , performs the same three stages. At 1 stage hand he puts on e to wound the computer objects that will be performed yat its programs have, on the 2nd stage of writing programs have, on the 3rd floor and ne runs it. What exactly do I mean by the placement of objects, you will learn in the next section, but now remains to say that due to the Phase 1 work program ISTA is now reminiscent of the work-designer designer and the product work progra m m ISTA is now called not program the second and **project** . Although cha

hundred used and the same term . I, too, will probably confuse them. .However, an e schayu that is not at the expense of meaning

The finished project will be called **a Windows application** or simply an **application** . This means that the program will work on computers running the Operations n dimensional system the Windows (say, the program runs *from under* the Windows) . It will not work out of other operations n GOVERNMENTAL systems that number hardly be called a disadvantage. These are almost all modern, etc. of a gram. And "omnivorous" programs . have their disadvantages

Another term for a finished project is **product** . Indeed: the product of .the Ista programs is a product

First steps - by the handle .1.3

The best way to get acquainted with the VB - quickly create and run to and Coy any project . P ust it is small, and no one, except us, is not necessary, but with it we about th dem all stages of work with the VB . This is what we .will do now

Sit down at your computer and let's get started. Without a computer, this section will bring you m and lo good. Learn BASIC for the book without a computer and - to behold like learning to ride a bike on the self-instruction manual without a bicycle. From now on, you must immediately execute everything that I say on your computer . This need for training and also because the plural of Goa which seems so obvious in the text, it is strange and incomprehensible, when n and you press the keys. Also, do not forget that the author is not the Lord God and can also be wrong. The computer did .not err and Byetta and always catch the author's hand

Sometimes I'll just say, " Click this button, then " or " Make to to m drunk e p e so-and-so " . But more often I will simply present the material, without repeating a hundred times that I need to check on a computer every word I say. For example, I say yes n naya procedure for this reason, prints the number 36 , and immediately move on to another t e IU. Stop! Don't rush . after me. First, check to be really PRIN and Tano 36

I remind you that e If you do not have enough experience on the computer, at your service Annex 2, where you will learn everything you need . to priest in drinking to the point

We launch Visual Studio . NET .1.3.1

As I wrote in about the introduction , the VB is a part of a single program s , which is called s INDICATES **programming environment, the Visual Studio . NET** and allows programmers of possibility to develop user-friendly programs VB and some other languages. Therefore, Thu to be etc. to program the on the VB , you need to run the Visual Studio . NET . The English word " the Visual Studio " can be loosely translated as "Visual Studio", ie the place where the artist (programmer) can do their work .((program) visually (their e etsya since Stage 1 - assembly by hand

In what follows, instead of the words " Visual Studio .NET " I will often write **VS** for brevity . Don't confuse VS and VB . When I write the VB , I mean the language Programming and Niya, everything that is associated with the program, with the writing rules of software are to a hundred, and his meaning. When I write the VS , I mean Progra Wednesday and Niya, that is, everything that creates comfort and programming capabilities, loading, saving, etc. of grams, their view and configure the interaction. Analogies: VB is a steak , VS is a restaurant and waiters ; VB is the brain, VS is the body ; The VB - is a program, the VS - is to m pewter . One and to, I must admit that it is not always possible to clearly say where it ends VS and beg and . naetsya the VB

If VS on your computer e are not set as read 4.1 , and if the set and , : then run it as follows

Click on the taskbar Windows button "Start» (*Etpu Start*), the pop-up menu SEL e Rita "Program» (*the Programs*) , then in the pop-up menu - *the Microsoft the Visual Studio.NET* , and then in the pop-up menu - again *the Microsoft the Visual Studio.NET* . (In the future, instead of verbal description of mouse clicks on the menu I have to brief about the STI will write this: **Start ® Programs ® the Microsoft the Visual Studio . NET ® the Microsoft the Visual St u dio . NET** (.

When you first run VS , then the result of the start-up screen appears GLA in Noah box VS with **launches the second page it** (**Etpu Start Page**) . If you have Visual Stu dio installed . NET 2003, you have a window of . about tacos th view as Fig. 1 . 1

Figure: 1 . 1

If you have Visual Studio installed . NET , then the window looks like .Fig. 1 . 2 . The difference between on to us purely external

Figure: 1 . 2

If you are of a start up VS is not the first time, the main window can not be the start page , or it may have a few others in a goy appearance. It doesn't matter now , we don't need the start page yet . In any case, you can SMOs . and a thief of it in 4.3

The title of the window you see the words of *the Microsoft Development Environment* , which means "software development environment of firm the Microsoft ». The meaning of these words so as Coy . « same as that of the words « the Visual Studio

We create a project .1.3.2

Now you need to tell the computer that you want to create a new VB project . Your actions: **File ↓ New ↓ Project** . There in front of you a : (dialog about to but (see. Fig. 1.3

Figure: 1 . 3

In the left field you choose *Visual Basic Project* in a sign that you want from h to give the project is in the language of the VB . In the right field you choose *Windows the Application* (Applications e of Windows) in a sign that the going will build Vat normal program that dale train to run on your desktop computer or laptop running Windows in Oba h nom window. In this field, you could choose, for example, the creation of programs s to work on the Internet or something other in Goa.

A little lower down, in the *Name* the computer offers n s the name of your future project , and the same name of the folder where it is stored - *WindowsApplication1* , and even n and , in the field *Location* PC offers the folder in which bud from t store s camping folder of all your future 's prospect of CPC s - *E: \ the Documents and the Settings \ Sergey Lukin \ the My the Documents \ the Visual Studio the Projects* . If you are now Mr. and presses *OK* , then inside this folder then the folder will be created *WindowsApplication1* , and it will immediately appear all the necessary files, which will be your future prospect of the CPC.

Remember: each project does
not consist of a single file and from multiple files and folders , so it would be foolish to try to save the files and folders of different projects "ck about pom" in the same folder. They will just mix. The V the S ave e red understands this and therefore offers you to save the project in a separate folder. You just have to agree. If you agree also to name *WindowsApplication1* , then for the next new proe to that VS will suggest a folder already *WindowsApplication 2* and so on. All of these folders will be in the ranks of co com folder inside *the Visual Studio the Projects* , like shown in the picture depicting Fra g cop . Explorer Wi n dows

If you do not like the folder *the Visual Studio the Projects* , you can . (select a different button *the Browse* (neopy t nym not recommend

But we have to click on *OK* anyway until we, as we need to come up with more appropriate than *WindowsApplication1* , the name for the project. Because of its meaning , as I us e Teal, will bookmark th chatsya to " paint " some "form, and type in the *Name* Name *Cr and SIM form* , and then and . click *OK* . The project will now be stored in a folder with this name and it

N Before you came n a next picture (Fig. 1 . 4) , which says that you m of zhete start designing (the appearance of the image you may have little .(other at a goy

Figure: 1 . 4

You now see the words *Microsoft Visual Basic.NET* in the title bar of the window . This means that the development environment for Visual Studio . NET has taken one of its guises, and it stood among the ra s rabotki programs on the VB . If you were creating a C ++ project . NET , then VS .would wear a different mask and look a little different

Go to Windows and check the contents of the *Paint Form* folder that appears . You are kind and those that your project consists of several files and .folders

For the future, I want to warn that if up to you in the VS someone has worked, he could reconfigure the environment so that it is something will behave very differently than I wrote. In this case, if you run into hard about STI, preliminary study Chapter 4 . In any case, I do not recommend clicking on buttons and menu items, the meaning of which you do not understand. Try also not to be distracted by a large number of PRIS so acting per screen

buttons and windows, which I still hold back, to not sink in for Dr. detail .about styah

Place objects on the form .1.3.3

What's in Fig. 1.4 should you most interested in, it's gray, etc. I rectangle or square , located on a white background . It is covered with a grid of dots and IME is Header of wok *Form1* . This is the so-called **form** . If the form you for some prich and not see or subsequently lost sight of , in the . main menu window VS select **View Take ® Designer**

What is this - form? Let's remember our toy railroad. So, the form is an empty table, the very table on which we will collect this road. For a GDS you run the project, it will be the application the Windows , but the form will be . the window Wi n dows , which will work your e application

Pay attention to the three white squares - markers - in the lower right part of the form. Grasp them with the mouse, you can change the posted e ry .form. But you won't be able to drag the form across the screen here

Let's start working on the project from stage 1. To collect iron dor about gu, we need facilities: rails, wagons, etc. To create a project, we are also well w us objects - and VB us their offers. On the left side of the main VS window, we are in and dim the icon . Do not put on the first mouse , and maybe even will click . The screen will pop up on the left vertical s n th gray . (box, not like a box (Fig. 1 . 5

Figure: 1 . five

This window **Toolbox** («Box" with the **element s control**), that is the most popular set of standard objects, etc. and change when creating projects for the VB (it is our pairs of Carriers and trailers) . Each of the b control coperation element has its own name : a button, a text box, a timer, msgs e . skoe n of le and the like

If you remove the mouse from the window *Toolbox* , it is a drift away back, it is wise freeing scarce and valuable space on the screen . To see it . again , place the mouse over the icon again

Note : The concept of " Object " is central to VB . It has a much more wide of the cue point, than the form or control, but as others have GIH objects we do not know, I'll call it their objects. Post e foam I will expand and .refine the concept of the object

Well, let's start assembling. I'll explain, "until what " and will tol s to

explain the "what" and "how." N omestim to form so-called CCW n ku . To do this, click in the *Toolbox* on the **Snap e** (**a Button**) . The mouse cursor will become the Features p ny kind of cross button . W ATEM hold the mouse within the form small " Ramo h ku " . What does it mean to hold " district and the lobe ' ? This means to place the mouse cursor anywhere inside the form, click on the left mouse button and, without releasing the button, " Archpriest and schit " mouse a little diagonally. In this case, a " frame " will follow the cursor . Release the mouse. In place of the fuss frame , there is no button labeled *Button 1* . However, a frame can not be carried out quite .simply click anywhere VNU t ri form

Along the edges of the button, you will see 8 Br 's squares - **markers** edited e neniya dimensions (Fig. 1 . 5) . If you click on the form by the button, they prop and FLS, again if you click on the button - will appear. Just do not yet double clicks (if you still sdela ete FEB th Noah click on your site of the former will open another window. In this case, simply select the window bookmark *the Form1. Vb [design]* , and everything will return to the place . Per e drag mouse l th battle of the markers - the size of a button you can change to carry on the form button, if. x vatitsya mouse is not over the marker, and for any location within the button. You can move the button keys and E to move the cursor on the keyboard. to destroy a button click her right mouse button and the context menu will pop up in the s take the option *Delete* and whether press Delete key on the keyboard . Do it all. and now, ie .if you destroyed it, then with again of give

Similarly Put those of *Toolbox* to form another button and cl e blowing facility s: **label** (**Label**), several **flags** (**CheckBox**), several elements **switch** (**RadioButton**), **a horizontal scroll bar** (**HScrollBar**) , **ve p tikalnuyu scrollbar** (**VScrollBar**) and **those to Stow field** (**the TextBox**) . . Arrange them approximately as in the left half of Fig. 1.6

Figure: 1 . 6

Did you notice that the text box resists Zoom eniyu his pa s measure .vertically? In the future, (3.4.2) we will overcome this soproti in Lenie

Note the word [**design**] (design, engineering) to zag of cleverly main window the VS . It says that at the moment VB is in **p e benching design** , in which we may collect , designing our project (Phase 1) and write a program for him (Stage 2). B e Loe window with the form in which we perform Constr ation is called **the window to Mr. struktora (designer)** . (**forms** (*the Windows Forms Designer*

Please note that when we place in the form of several identical b ects n as an example of *a Button* , the VB inscribes them with numbers : *the ... Button1 , Button2 , the Button3*

Trial run .1.3.4

We launch . While you and I are creating that - "we ourselves do not know what . " After all, no to Mr. indiscrete task we have set ourselves are not set except misty wishes "item on the paint form" . However, to skip over the 2nd stage (drawing up programs s) and q and Dhu perform their Phase 3, ie run our baa with sages project performed ix. To do this, click on the button (**Etpu Start**) on the **panel of tools** (not to be confused yte toolbar with . *Toolbox* - «box" e e cops sound control in Lenia) , or press the **F 5**

Start tili . Historical moment. What do we see? Some time computer project and liruetsya, ie language VB translated into a language close to the car (for more details, which is **a compilation** , see Appendix 1). After a few seconds of Compiegne th tera appears on the screen , " another copy " of our form. This is our e adj of voltage the Windows , ie running , compiled and

running the project (see., Etc. and -hand half of Fig. 1.6) . What we see is no longer a form , but an application window. It differs from the form, like a daughter from a mother, that is, they are very similar. So in the future th shem I will often call our application running window also form. I'm over th

.yus, you will not confuse it with its parent s nitsey in the design window

We see that our working form is still a bit different from the form in the design phase , from which he and begat a camping : missing the point, transformed form of lost contact with the main window VS and behaves as independent st window , what fired it, in general, it is . For persuasiveness, minimize the main VS window to the taskbar. T e per form one prevails on the screen. It can be dragged around the screen by grabbing the title, and you can see its icon on the Windows taskbar . She breakdown e zhaschim responds to clicks on the buttons in the upper right corner. To sum os t Rie cursor to the edge or corner of the form, you can change its size. And these

.are the measures Knop c and other elements we can no longer m e nyat

But now the button and can be clicked with the mouse. Click. Nothing happens. Of course - we do not have written a program that explains Comp u ter at what to do when you press the button. In the text box t e per text can be entered, and how you want it to change it and wash . Mouse clicks, you can set and remove the flags (halo h s) in the elements of *the CheckBox* . Mouse clicks, you can set changeover and Tel in any desired position of the four. You can use the mouse to move the slider to the floor of the scroll. But all this is useless. The computer does not respond, that would in s there any d e l

.Ali. Etc. and rank the same

.So, we see that without a program, our project is not worth a penny

If the working form is on the background of a window (for example , and measures, principally of the window but the VS) and you accidentally click on a that the window in the mouse by the form, the shape of the prop and the children of a kind, as it will be blocked by this window. It does not matter - slit l choke icon form on the taskbar, Windows will return it to the re

.d Nij plan

Notice the word [**run**] in the VS main window title . It says that at the moment VB is in **operation** , that is, directed and IU of the project (Phase 3), during which we may build the project (Phase 1), or write a program for him (Stage 2) is not can , as with about communicate the word [**the Read . Only**] in the same title

For the top we eat . Finish the project by pressing the (*Stop Debugging*) button on the toolbar or by clicking on the cross in the upper right corner of the working form. VB exited [*run*] mode and returned to [*design*] mode . Now the project can be anything and change it again .ZAPU with tit. Etc

Mesh . We see in the Design view that forms and covered with a lo t Coy of the points. The goal of the net is twofold. On the one hand it helps us to visually assess and Vat pos e of the controls on the form, on the other hand, we can adjust the VS so thu about to control the outline Raspaud Laga strictly along the lines of the grid . It helped as a programmer to do the same controls on the size and location with a causeway them in the form of regular .rows

Writing a program .1.3.5

Let's come up with a task for our project. Let the screen when you press a button , say, or when moving the slider on the bands f scroll anything produ with walks. It's easy to say - let it be! To do this, you need to know what VB can do when pressing buttons or moving sliders . And even if we know it, then all the pa in but do not know how it is ordered, that is, how to write a program correctly. Well. Programming we will learn gradually and gradually learned everything m camping, and now I will choose something very simple . and tell you how it's done. Your computer is not on d go until I give the sign

Came up with an operator . Forget until the "paint forms' and n rikazhem Compute e p y to start pressing the button *B utton1* sdel amb width horizontal strips etc. of twist equal to 10 0 pixel I m [H] . Team (operator) :for this item and shetsya as follows

HScrollBar1 **.** *Width = 100*

.Let's figure out what is written here

HScrollBar1 - is *the name of* our first horizontal scroll bar (E th gave the VB . About how to learn the names of objects and modify them, I will tell . (later

. *Width* - " width " is translated

Meaning the operator is obvious: this is an order to make the width of .the object *HScrollBar1* pa in Noah 100

:Now that denotes th t point and equal sign . P eye claim ravilo is

To the left of the dot, we write the name of the object, to the right of the dot , the name of its *properties a* , and to the right of the equal sign , the value of this property. Point you need to write about me .necessarily

: Here are the taxes and

Kolya . *height = 140*

Brazil . *climate = hot*

Ning a . *surname = Rybakova*

In the future, we will see that the property itself can be an object, and :therefore, he so on can also be its properties. Here are the taxes and

Nina . *surname* . *number of letters = 8*

Pinocchio . *nose* . *length = 30*

Tank . *engine* . *gear* . *material = steel*

Said PROPERTIES *Width* object *HScr ollBar1* **assigned znach e of** .100

Code window . Now let's talk about that , as we do to a computer execute this statement , and it is when you click on *the Button1* , and not at any other time . With new with adites at the computer. Have you erased our project yet? Check nah about whether we are in a draft mode dimsya and tion [*design*] . Double-click *Button 1* . Before you there will n about howling .(window - a window of code or simply **the code window** (Fig. 1 . 7

Figure: 1 . 7

We will call the **code** any text programs. In the code window will zap and ik- entire code of our project (and larger projects can use and MULTI s .(to the code window

If for some reason you lost sight of the code window, select **View** ↓ . **Code** from the main menu of the VS window

Procurement procedure . We see that something has already been written in the code window. This nap and fats itself the VB . The situation is like in a restaurant: we have not ordered anything yet, but salt and pepper are already on the table. In addition, we do not even see much of what BASIC wrote , and we do not need to know this invisible yet. If you are an experienced person and you are curious to see this invisible code , click on the tiny plus sign next to the Windows Form D e signer generated code line , but not everyone will then be able to find the resulting minus sign and click on it to return everything to its previous form . If you are still caught and do not know how to get out of the situation , then either drop the project or item . to read 6.1.3

. Let us dwell on the meaning of the two lines visible in the figure

```
Private Sub Button1_Click ( ByVal sender As System.Object, ByVal e As
System.EventArgs) Ha n dles Button1 . Click
```

The word **Sub** stands for " procedure " . Word **Click** means " Click pull the mouse ." We do not dwell on the meaning of the rest of the words in the line. With the Troc can freely Perevi with five as the " What to do if you click your mouse button on *the Button1* » . And more exactly: ' Title n The procedure for s , in which the computer is written ins t ruktsiya what to do if « one arm n and presses the button on *the Button1* .

Below there is a free space and the cursor is already blinking there, inviting us to enter any operators we want there. They also will ful l Nena when the button is clicked.

In the lower part row *End Sub* , which means " end facials y p s " . All the space between these two lines, which we will fill operas and tori, we called a binding **procedure body** .

Thus, we see in front of you is not nothing but a blank procedures s cat of paradise should be executed when the button is clicked *the Button1* . Helpful VB realized with s to the workpiece, when we double the slots to zero by button *the Button1* .

You need to enter from the keyboard in the space between the two lines of our op e Rathore

HScrollBar1.Width = 100

If you have no experience of entering text into the computer , it will suspend and study Applications e of 2. You have no right to prevent the slightest error in the string. Even once, etc. on white can be oshi B Coy.

We enter the operator . Let's start slowly introducing . Enter the word *HScrollBar1* and then ch ku , and VB will forgive you if you enter a lowercase letter instead of a capital letter or vice versa . As soon as you enter a point, VB (so that you don't tire your fingers with the word *Width*) will unfold in front of you a list of all relevant words after the point. You about to Thanet just double click on the desired word (Fig. 1 . 8) , or highlight it and press on the keyboard at the D key *Tab* .

Figure: 1 . 8

If you do not want to scroll through the list to find the right words, you can just start typing a word from the keyboard immediately after the point , the computer then he will scroll the list and the right word finds and selects . Do not be lazy after all the n and press key *Tab* , even if the word and almost completely appended you, because it's about e cuts you from possible .grammatical err and side

But now the operator input is over. Remove now the text cursor from the line, the cat of Rui introduced. VB at the same time check the grammatically correct line and a bit d governs its external appearance, the basis of their perceptions of grace. If he Detect from INH error, the erroneous part will accentuate lines . If you see what the error is, fix it. If you haven't, . read 1.3.8

: Now the code window must be so as hell kind

```
Public Cl ass Form1
Inherits System.Windows.Forms.Form

Windows Form Designer generated code

Private Sub Button1_Click (ByVal sender As System.Object, ByVal e As
System.EventArgs) Handles Button1.Click
HScrollBar1.Width = 100
End Sub
End Class
```

Here is a ready-made procedure. Check if you have inadvertently entered anything above, below or to the right of these lines . If entered, then

.with about rub

We launch the project . It's time to start the project again. Let's assume that there is no error and bok in the code . Click the *Start* button . On the screen appears again the familiar form of a servant of the melting mold . Click *the Button1* - horizontal scroll bar is changed and la w his and Rina (. (Fig. 1 . 9

Figure: 1 . nine

Congratulations! Your first project is working. Click again. Nothing .changed and elk. Naturally. Let's complete the project

If at startup you did not get what you need , or VB gave some kind of . communication , read subsection 1.3.8 just below

Fast switching between code and designer windows . Above the code you see tabs (Fig. 1 . 7) . You can at any time slots to pull the on any of them and see, respectively, the start page or window of Form Designer ., or removal of the Island code window

Save , close , open, create a project , quit VS .1.3.6

Conservation . Now it's time for us to save the project. Actually, we should have survived a long time ago, before the first launch, but we didn't want to be distracted. Remember that n e saved project will be lost if the camera suddenly turns off electric e GUSTs or a computer hang. With Save the project are now , by clicking on the button (*the Save the All*) on the toolbar **.** Subsequently recommend retain s Xia thus performed at every time . before starting the project e of

You must understand one thing. Each time you press the button *the Save the All* , the VS , anything you without asking, erasing a disk ect your about EKT (in the form in which it was saved for the last time) and on e of place records with the same with his name and with the corrected and supplemented version by you , that is, the one that you see on the screen. Tutte with a temporary software. At the discretion of pr of programmers left resh and be, and not huzh ie whether a new version of the old and do not .mind if the old ve r these

Read the further material of the subsection without performing. Fulfillment of t lay down for later, when you really need to close, open or .create a prospect of the CPC

Closing . If you want to finish the work on the prospect of ektom, you need to save , and then just close it: **the File ® the Close Solution** . You should now be blank in the *Solution Explorer* window . If you have forgotten .to about store design, the VS before closing prompts you to do so

Exiting VS exactly the same way as most other applications the Windows - click the mouse on the X in the upper right corner of the home of the first window VS or **the File ® the Exit** . If you forgot to save the .project, VS will prompt you to do so before leaving the house

Opening . Save and close the project , you can safely get out of the VS and the m of zhete even shut down your computer . You can not go out of the VS . To continue after a while working on the project, again run the VS (ie with whether you come from a non e) and the so- load (open) a saved project : **the File ® the Open ® by Project ®** n Before you open a dialog box open the project ® Zayd and those in project folder (Fig. 1 . 10). If you do not know how to get to the desired folder proe to that, read the next .undermining of affairs

Figure: 1 . ten

Then click on one of the two files, namely, on the file of the so-called *solution* (it is highlighted in the figure). The other file (in the picture below) is the project file (by the way, you can click on it too). As you can see, these two files differ in their icons and extensions. The Posted in General on .(projects and solutions (*Solution*) will talk later (21.3

Then - **Open** . You will see and tsya your project. If the same does not appear, and even of a masonry windows Form Designer and the code does not appear lyayu tsya above the window, double-slit l choke icon *Form1.vb* in the window *Solution Explorer* (Fig. 1 . 5). The form designer window will appear . If the same is nowhere not seen windows *Solution Explorer* , sdela yte following : **View Take ® Sol u tion of Explorer** . Another joint venture of the GSS: **View Take ® Designer** (so that the window appeared .(Form Designer) and **View Take ® Code** (to display the window to a yes

Creation . If you are working on some prospect of ektom, and then want to throw it and start a new project, the first save and close th the projects you're working on. Make sure the box *Solution Explorer* you poo with something. Then follow **the File ® the New ® by Project** , and then - like . already on n and Sano in 1.3.1

When creating a new project, be sure to give it a suitable name. All your m project s invent , of course, different names. For each new project VS CREATE as a separate folder with the name you. After a while, you will accumulate a n e as folders Saving E project s . They all disintegrations of . lozheny side by side inside the folder *the Visual Studio the Projects*

Situations . If you are working in a project, and then, without saving, and not of a kryv it decided to open or create another project, the VS first prompts you with a store old and after saving will automatically remove it from the environment (not from disk, to about infinitely) to make room for .the newly on the second

If you by mistake instead of **the File ® the Open ® by Project** or **the File ® the New ® by Project** ful l Neath **the File ® the Add P roject** , the old project will be deleted and the SFA in the SG box VS you will have two instead of one project at once. You will immediately notice this, and those, looking at the *Solution Explorer* window . Sometimes be in an environment just two projects convenient, Institute of GDSs just req on Dimo, but as long as you do not need it . If this should happen, press while you are still inexperienced, **the File ® the Close Solution** , answering refused offers preserved and nitsya , as long as the window *Solution Explorer* .does not omit e a

Often a new project that we create is similar to an old one, and it is more convenient not to create a new one from scratch , but to redo the old one. This is my experience. I , without opening the old project , copy the whole of his folder. So now I have to drive two Odinak about O project. Then I open a copy of the project in the usual way and redo it into a new project. How to copy a folder, written in Appendix 2 "Actions on files and pa n kami" . You ask: Why copy when you can just redo the old project in its folder? Answer: in the process of remaking you fail, then you and the new project .will not be created and the old-used district chen

Working with files and pa n kami in the .1.3.7 window ah Open Project and the Open the File

In this section you'll learn how to travel in the nutri ok it *Open Project* and *Open File* folders and drives, to find in the district of the desired project , the folder, the file you want , but other than that - no s walking from the app it to create, copy, move, and delete files and folders , without which work

with VB can not do . All the mentioned work in the windows *Open Project* and *Open File* is very similar to a similar job in the windows of the Windows ., as described in the Applications e SRI 2

Open Project window . Run the VS . Then : **the File ® the Open ® by Project ®** n Before you will be shown a dialog box opening of the . (project (Fig. 1 . 11

Figure: 1 . eleven

Click on the *My Projects* button in it . You will see the contents of the . *Visual Studio Projects* folder

We travel through folders . In the large white box in the middle of the dialog box, you see a list of folders and files . This is the contents of the folder whose name you see in the district and Catching up just above the drop- down list is, *the Look in* . You can double-click to enter the l th buoy folder in this field . Then it appears in the list is, *the Look in* , and in a large field on the I vitsya its contents. Thus, you can move deeper into the folders, like inside a nesting doll. How to move outward? There is a button for this . Clicking on it you get out of the folder, as shown in the down list is, *the Look in* , a folder, inside which it is part. Another click is another step out. And so on until you find yourself on the Windows desktop (*Desktop*) . If you want

to again go inside and get to the prospect of OBJECTS sign in *my dock at* . *cops* (*the My the Documents*), and then - in *the Visual Studio the Projects*

If you now double-click on the *Paint the form* folder , you will find . yourself inside it, that is, you will see what is shown in Fig. 1 . 10

Clicking on a black triangular arrow in the drop-down list *the Look in* , you will pa a hidden folder list in the form of a tree similar to Windows Explorer tree the Windows (see. Fig. 1 . 12) . This will allow you to navigate .folders faster

Figure: 1 . 12

Practice. You should be sure that you can get to any folder on any disk .of your computer , and therefore open any project

Open File window . In the *the Open Proje ct* you see all the folders, but here are the files you can see , not all, but only those whose types are indicated in the list *Files of the type of* . That in and put all the files you need to call on the screen is not the dialog box you open a project, and di a log window to open a file, that is, do not perform **the File ® the Open ® by Project** and **the File ® the Open ® the File** . Do this and then go to the .(folder *Paint form* (see. Fig. 1 . 13

Figure: 1 . 13

In Fig. 1 . 10 we see only two files in the folder, and here - seven. In principle, the window *Open Project* and *Open File* are similar both in mind and in purpose , only the second show 's Vaeth all files and works with all fa . st Lamy, and the first - not so

The file open dialog box can open both projects and files. At the same time VS opens them in your environment, and if you find them too strange - . is Podhom of dyaschee from her point of view, the app the Windows

Further explanations will be given using the example of the *Open File* . window

When you perform some actions with folders and files in the dialog boxes do not advise you to avoid the unpleasant consequences of producing .them on folders and file and E open prospect of EKTA

We create folders . Inside the folder selected in the list is, *the Look in* , you can create new folders. The new folder will be created if you click e are on the button . Immediately you will need to enter her name from the .keyboard

We create files . Right-click on the white space of the large field and

select *New* from the context menu . The fuss to Shem submenu, you will see the types of files that you can create within the folder in s abusive in the list is, *the Look in* (see. Fig. 1 . 14). Click on the desired type and create . Among these types, and you will find *Folder* , which means the creation of .na n ki

Figure: 1 . fourteen

We copy files and folders . To **ck about feasting** file or folder to another location, select the shortcut menu of a file or pa n ki option *the Copy* (copy). Then Dober and Tes to the folder where you want the file or folder to copy, that is, dobe th Tes its presence in the list of *the Look in* , and right-click on the free space. In the konteks ton nom menu, select *the Paste* (insert). If you want to copy is and familiarize near Original and nalom, to get to the .other folders do not need to

Moving files and folders . To **move a** file or folder to another m e hundred, do everything the same as for copying, only instead of *Copy* from .(the context m e nu option *Cut* (cut

Rename files and folders . To **rename a** file or folder, select the

. context menu option *Rename's* (n e reimenovat) and enter a new name

We delete files and folders . To **prowess be** the file or folder, select konteks ton nom option menu, *the Delete* (remove) and the answer is . affirmative v but the question of the computer

As VB react uet on our Îøèáê and .1.3.8

In 1.1, using the example of going to football, I have already shown you the catastrophic consequences of errors in the program. What happens if you make a mistake in your VB program ? It depends on what kind of error and whether the VB her Timeliness e Menno about b out. Let's consider 3 types .of errors

Semantic errors - 1 . So you've saved your first project. Close it - **File ↓ Close Solution** . Create a new project. It is designed to ensure that we deliberately tolerance if it mistakes and teach were them with directs. Name the project something like "Oshi b ki". Put on the form a single button .((Fig. 1 . 15

Figure: 1 . 15

In design mode, double-click this button and in the resulting procedure : template in the code window, enter the following code

```
Private Sub Button1_Click ( ByVal sender As System.Object, ByVal e As
System.EventArgs) Ha n dles Button1.Click
    Dim a = 9 0
    Button1.Width = 100 - a
End Sub
```

. Let us explain the e nd

The operator *Dim a = 9 0* means "We declare that there is a variable *a* and assign it the value 9 0". In other words, "Let *a* be equal to 90". Oper a torus *Button1.Width = 100 - a* means "To make the width of the button *Button1* equal to *100 - a* », i.e. in this case *1 0* . If you have forgotten what a . variable, h and thaw 5.1

Run the project, click the button . The button has become too narrow , since 10 is quite a small value . We tend to think this is our fault, but we do

not require a Vat of the VB , that it such an error is detected. Indeed, how would he know m of Jette, we like to have such a narrow button. In this mod and Zoom, there are meaningful (from the perspective of human) errors in the code and the design of the project , which VB error does not think so does .not detect and basically detect should not

Semantic errors - 2 . Complete the ones performing the project . : (Facials change the code from ry (I only show the body

```
Dim a = 300
Button 1. Width = 100 - a
```

Sheer nonsense . It turns out that the width of the button should become negative (-200). It doesn't work that way. Run the project, click the button . The button disappears. Complete the ones performed e of the project . This error VB could detect and let us know. But not about the bodies. Or he couldn't. Just made the width of the button zero. And it seemed to us that she and from the chezla. MF and a thief if it is lack of the VB ? I do not know. In any case, we should know that VB is not so smart and helpful to discover all .our stupid even explicit about STI

Complete the ones performing the project . I recall that, before any change in the code or design project, the implementation of necessary .Covenants p shat

Errors performs eniya . While we VB only pyr and there. It's time :to praise. Let's change the code of the procedure

```
Dim a = 0
Button 1. Width = 100 / a
```

The " / " sign means division. What happens? From the beginning , we require that the Move n naya *a* st and la equal to 0, then we want to number 100 was divided into 0. divide by 0 - is not humanly possible. Interestingly, the computer may be able to something that is not d and but a man? Run the project, click the button. VB outputs from a communication error (Fig. 1 . 16 .(

Figure: 1 . sixteen

What happened? After pressing the button, VB began to honestly execute the procedure and the first statement *Dim a = 0* executed without problems. When performed ie the Research Institute of the second operator, he was faced with the need to divide by 0 , and being unable to do it, .interrupted the program and issued a corresponding coo b schenie

The message is intended to explain the man who knows England th sky, and having some programming experience, what is the error. If you have no such experience, remember that the word « *the overflow* » from the text .message is often the Criminal Code and zaniem to divide by 0

Also VB kindly highlight operator cat about ing failed to perform, and marked by an arrow-pointer to a string in which it is located. We immediately .know where to look oshi b ku

You can read more about how to deal with errors in Chapter 9. For now, click the *Break* button in the message box . The program was interrupted but not completed. It froze at the moment of error. Now, for a clue what went wrong, the field of but to know what is currently in s complements the program (that is, before performing labeled statement) are the values of the variables, properties, and other elements proe to that. To do this, simply place the mouse cursor over the item, without clicking, and vzgl I pull the tip to the supernatant. In Fig. 1.17 the cursor placed on the property of the *Width* and we and dim sum in a tooltip that the width of the buttons is 96. This is the

width of which was *to* perform e of the operator *Button1.Width = 100 / a* , that is, that the original width of the button , which we've seen before time . and n and zhatiya it

Figure: 1 . 17

Place the cursor on the variable *a* - and VB tells you that *a* really at this .point ra in on zero

Now, in our simplest case, all this information is not needed, but in the future, knowing the values of variables, properties and other project elements .at the time of the error, you will find it easier

Complete the ones the implementation of the project by pressing the button (*the Stop the Debugging*) on the toolbar and and to edit them a . mistake

So, the VB can not detect runtime errors in advance, even before the start proe to that . Indeed, how could he know that at the time of dividing the ! variable *a* will pa in at 0 ? Well it is necessary to look in another line .(Runtime errors also called **exceptions** (*the Exceptions*

However, there are a number of errors that VB detects beforehand. Here .they are

Compilation errors (build, build e r rors) . If you are talking to a foreigner, in order that he understood you, your proposal must be properly constructed grammatically, and have clear John about Stra n zu sense. Computer - the same alien, but in addition also a very "dumb" foreigner, so all that you claim and shete in the code window, there should be absolutely right in terms of VB and grammatich e Ski and meaning , otherwise he'll just will not understand . I'm not going to go into that so and . some "meaning" from the point of view of the VB

When you start a project for the execution, the VB primarily looks, whether in prospect of gram grammatical and semantic errors , and only if everything is correct, it allows the project to run. This search for errors occurs even before starting, even in the process of entering the program text into the

code window. Suppose you have incorrectly recorded seq about in *the Width* (cm. Fig. 1.18) and did not notice this . When you K and P and ete text cursor from the page of ki, which was administered, the VB is already beginning above the line work . He bit podprav lyae so it is w Nij view, leading to the standard. In particular, arithmetic signs and signs of equality b e rutsya in the gaps; capital letters are placed, as expected. But most importantly, the VB detect Vaeth some errors and handwriting ivaet their . wavy lin and her

Figure: 1 . 18

Place the e mouse over the underlined portion of the string without pressing, and look at a tip. You can guess the error from its text. In our case, those to ste tips we see the word *'WIDS'* . Nawa district Noah, with him that .something is wrong. Correct the error, the underline will disappear

Some errors VB detects only if , when you zapus Kae those project execution. It happens that you do not notice the underscore and running etc. :(of the CPC. In both cases, VB outputs a message (Fig. 1 . 19

Figure: 1 . nineteen

It translates it this way: "There were mistakes of building (compiling, *the build errors The*) . Prodo l harvesting work project? ". This is roughly the same as asking, "Torm on for no run. Are we rushing on? " But even if you answer *Yes* , VS will still run the previous version of the project, in which there were no errors yet. Answer *No* , and Ispra in cumstances should the .error

VB is wrong and self . VB is not always correctly finds place error,

and finding, often incorrectly determines what the error condition , and so on . Therefore, it happens that programs and stu have to turn into a detective , . searching for the perpetrator of St. misleading and detelskim indications

Council . I will not dwell on the complex process of finding errors here, since you are not yet ready for this (read Chapter 9.) . If VB throws an error, look again to see if your code looks like it should . Is there anything superfluous? Do all England first skie letters in the text actually English, or among them there are Russian? Person is very easy to confuse the English letters with the same mark of Russian (especially the letter "c") , the .computer did not confuse never Pryderi so camping

Complicating the project .1.3.9

Close the error project. With the copy of the project folder "Beauty of form," then, the series co com with internal and folder *the Visual Studio the Projects* . Rename the copied folder in some way, for example, "Paint Form 1". On the copy we will Prodo l harvesting work in . Open the project from this folder. This can be done on Windows by double-clicking the project or solution file. Note that by changing the name of the folder name for the . project and not of Maine and elk

Complicate and m project. Suppose that when you click on the button *Button1* happens anything else but change the width of the scroll bar , namely : three in the e soup

- **In those to Stow Field *TextBox 1* text appears *Bandwidth* = *100***
- **. (And scheza a second switch element (*RadioButton2***
- **.The second button (*Button2*) turns yellow**

To do this, it is enough to add 3 operators to the procedure . Let's start .by smacking Dr. Ku

Change the text in TextBox1 . In the text box *TextBox1* is a property of *the Text* , znach e Niemi which is the text in this field. So we add in operas procedure and the torus

" TextBox1 . Text = " Band width = 100

Remember: In the code value of the property *Text* to the text field .and other lens must but take FEB th nye quotes

Get used to the same Symbol e niyama in VB different things (we have it *TextBox1* and *the Text*) and be careful. Our statement can be translated as follows: The contents of the text field *TextBox1* do so - *width n of VOCs = 100* .

Don't assume that the meaning of the words *Stripe Width = 100* means anything to the computer. Firstly , it is in Russian does not understand, and secondly, no matter what was written inside double quotes, he is not to try to understand how the order and does not accept , but simply in a produced in the text box . To check, you can n and write in quotation marks any abracadabra and run, etc. of the CPC .

Here's what we should get :

```
Public Class Form1
    Inherits System.Windows.Forms.Form
    Windows Form Designer generated code
  Private Sub Button1_Click (ByVal sender As System.Obje ct, ByVal e As
                System.EventArgs) Handles Button1.Click
        HScrollBar1.Width = 100
    "TextBox1 . Text = "Shirin and band = 100
        End Sub
    End Class
```

To save vertical space, I've erased blank lines here. And you also do not vozbr and nyaetsya.

Let's start the project . Press the button *the Button1* . By clicking on the button *Button1* VB vypo l n um on the order of all the operator 's procedure between rows and mi *Private Sub Button1* **...** and *End Sub* . Let's make sure that both statements are executed. Because the computer is working very bys t ro, we would seem that both operators performed and were at the same time. But this is not the case: first changed eh width of the scroll bar , and only then the text in the paragraph about the le .

Manually edit the text in the text box. Click *Button1 again* . That the prospect of emanated? Complete the ones performing the project .

RadioButton2 disappears . For extinction element *RadioButton2* enough to finish in the procedure of such operas as a torus :

```
() RadioButton2 . Hide
```

Explanation : In addition to properties, objects also have so-called methods. **M e Tod** - that's what the object can do. Let us recall the example

of the games at the shech railway. There were points of locomotives, svetof on ry, scapegoats, etc. In the locomotive may be such methods: to go, to buzz, to let off steam ... At the traffic light - RTA and causeway red, s and Gigue green ... We switchman - translate arrow to sing a song ... To A team n :and rovozu and switchman might sound like this

Steam locomotive . goody

Switchman . move the arrow

Since the " *Hide Event* " in English means "hidden", the operator *RadioButton2.Hide ()* does not mean that other, as an order element *RadioButto n2* spr I tatsya, become invisible . They say that the following method **Hide** object *RadioButton2* . We ignore the brackets *( ) for* now. . About them on of the same

.Many methods are called **functions** . But , and this , too, later

.For writing operator, run the project, check his work

Paint Button2 . Now we need to color the *Button2* yellow. Etc. and am well on ogy. To color in yellow paper, we would have called and written :as follows

Newspaper . u vet = Yellow

However, it would be inaccurate , because the word "yellow" m about Jette have a figurative sense - "yellow press". Therefore, it would be better to :write like this

Newspaper . paper color = yellow

However, it would be inaccurate, because the yellow paint from different boxes with paint and E may have different shades. Therefore, it :would be better to write like this

Newspaper . paper color = Box # 5 . yellow

.Now you can't confuse

A computer all the more requires precision, so we write the operator we :need as follows

Button 2 . BackColor = Color . Yellow

Explanation : Here **BackColor** is a property of the *Button 2* object . It is on of the beginning of the color pover x Nosta button. (And there is also, for example, *the ForeColor* - color of the letters on the button.) Word *Yellow* n e

revoditsya "yellow". And the word **Color** , which stands in front of him, is nothing more than the name of a " box of paints". Do not confuse similar in spelling, but absolute, t but different in the sense of the word: *the Color* and *the BackColor* . And let them not deceive you boo to Shaft per e waters from English. Also don't ask me what a paint box means from VB 's point of view .

!Tomorrow, tomorrow, not a day

:This is what your procedure now looks like

```
Private Sub Button1_Click ( ByVal sender As System.Object, ByVal e As
System.EventArgs) Ha n dles Button1.Click
    HScrollBar1. Width = 100
    "TextBox1. Text = " Band width = 100
    () RadioButton2. Hide
    Button2. BackColor = Color.Yellow
End Sub
```

Run the project, click *Button1* . You should see a ka p Tink (Fig. 1 . 20 .(

Figure: 1 . 20

Further complication .1.3.10

Close the project and make another copy of its folder - "Paint Shape 2". Over to about Pia will continue to work at . About t Kreuth project from this .folder

Work for the second button . Let's write a program for the second : button (*Button2*) as well . Suppose when you click on it

- **The width of the horizontal scroll bar becomes 200**

- **In those to Stow Field *TextBox1* appears the text ""Bandwidth = 200**
- **The radio button element (*RadioButton2*) becomes visible again .**
- **.Button *Button2* painted in red color**

. As a result, the picture should look like in Fig. 1.21

Figure: 1 . 21

In the design (design) mode, double-click on *Button2* . Before us - the code window again, but it appeared as camping harvesting another procedure, that is, a new invitation - this time an invitation to enter the statements in response th boiling on a button click *Button2* . Let's introduce them. : Now the content of the code window is as follows

```
Public Class Form 1
    Inherits System . Windows . Forms . Form
        Windows Form Designer generated code
Private Sub Button1_Click ( ByVal sender As System.Obj ect, ByVal e As
    System.EventArgs) Handles Bu t ton1.Click
        HScrollBar1. Width = 100
    "TextBox1.Text = " Band width = 100
        () RadioButton2.Hide
    Button2.BackColor = Color.Yellow
        End Sub

Private Sub Button 2 _Click ( ByVal send er As System.Object, ByVal e
```

As System.EventArgs) Ha n dles Button 2 .Click
HScrollBar1. Width = 2 00
"TextBox1.Text = " Band width = 200
() RadioButton2. **Show**
Button2.BackColor = Color. **Red**
End Sub
End Class

Explanation : To element *RadioButton2* again became visible, we use . and whether its method of **Show** . The button is painted red

Run the project . Click on the buttons. The picture you have alternately . used in children have a form of the Fig. 1.20 , then Fig. 1.21

Visual Studio . NET 2003, for the convenience of the programmer, separates procedures in the code window with a horizontal line. In the book, I .will not do that, confining pass page of ki

Analogy . Let's draw an analogy between our program and the program we gave to the football lover in 1.1

That program consisted of a number of procedures, it also consists of n .is, how many (two) procedures

That program describes the behavior of a single person in this inventory . flushes Stand e denie several objects on the form

In that program is, procedures s composed and team recorded one after the other and s is satisfied in the order of recording. In this program - the same: the procedures s condition I t of the operators written one after the .other and running in n of the row entry

In that program we are, met the procedures s two types ("How to ..." and "What d f lat if ..."). In this program , both procedures are of the same type - "What to do if the button is pressed". About the prospect of the .procedure is like "How to ..." later

In that program is, events - is "Stuck Elevator", "got lost" in the **GSS s** .**ment** - a key press

We paint the form . But what about the promised painting of the form? Oh yes , I forgot ! Place Front and those in the form of tr e Tew button :and create a procedure for it

Private Sub Button 3 _C lick (ByVal sender As System.Object, ByVal e
As System.EventArgs) Handles Bu t ton 3 .Click

```
Me .BackColor = Color. White
End Sub
```

Me - in Russian "I". This is the name given to the form itself in the form code window , despite the fact that its official name is *Form1* . We paint it . white (*White*) . The result of pressing the button 3 in and den in Fig. 1.22

Figure: 1 . 22

How to work on a project in VB .1.4

Let's summarize our experience in creating a project . This must be done in order to n and write for myself immutable rules create all future projects . For example, I still follow them. And everyone follows. These rules are a little reminding th t description of Sisyphus life, with the ra of difference that is mainly rock on top , we still delivered it!

These are the rules:

. Open the proje

With a little work on the project in design mode , that is, place it on th form of multiple controls or write code in the window one or two lines , . and so forth of a gram

Save the projec

Run the project . If you are satisfied with the results, go to step 2

Correct the errors in the project . Go back to step 3

On my programming teaching practice, I have never met with those

who disregard paragraphs 1 and 4. However, very often the meeting were .those who neglect Poon to Tami 3 and 5 and properly executes paragraph 2

How did life punish them? Enough with the level. Those who neglected item 3 often lost their projects. Those who despise paragraph 5 and moved forward, putting correcting semantic errors then finally entangled in a large number of accumulated Osh and side. Those who tried to paragraph 2 to enter the entire code at once, and not a small part of it, 9 times out of 10 received the wrong results and could not figure out where to look for error, because .the code bol s Scheu

?What's next .1.5

So, the project is ready and working. What's next? Is not very clear, we are going to lock a program the something more complicated, such as a shooting game of introduction . I n about yasnit. You stretch the shape to fill the screen and give it not a color, but a photograph or a drawing of the city you draw. This is easy to do (see 3.6). It turns grams of race in all e to the wounds. Next, take in the *Toolbox* and arrange the form objects such as *the PictureBox* (image) - it's your future cars, passers-by, gangsters, etc. in there, etc. Then give each object the desired photo . Finally, write to each of these objects programs in behavior, including response to keystrokes keyboard and m s shinye clicks. The game is ready, you can start. The main difficulty here, of course, to Mr. and writing programs s, it will be quite difficult and you will have a lot to have ourselves to, to feel the strength of .her s to give

In VB a lot of interesting and useful controls. So, having inserted an object of the *Timer* type into the form , you will be able to control the work of the project " in seconds " . Many el e ments management *Toolbox* are not shown. But they are easy to premises with tit and enjoy them. For example, inserting a form of the type of object *the Windows Media Player Get* , you can were accompanied train to the game music, sound effects and video. Chapter 3 introduces you to many n of the useful to the pleasant and features . the VB

However, to understand this chapter, you will have to eat a little "pound .of salt." This is Chapter 2

Chapter 2. Objects and namespaces

In the previous chapter, you got acquainted with some objects (button, label, etc.), their properties (*Width* , *BackColor* , etc.) and methods (*Hide* , *Show*). All the power of your capabilities in VB is determined by how a number of objects, their properties, met on poisons and other item s you .know and know how to use properly

All the standard objects are provided by the .NET Framework Class Library , which I mentioned in the Introduction . For starting it , this library seems to lab and Rinta. In this chapter we shall learn to move through the maze in search of the desired n s object , and at the same time and get . acquainted with a certain of torymi of objects

The concept of an object is central to VB and quite complex for a .beginner. We will deepen this concept throughout the book

Deeper into the namespace .2.1

Analogy . So far, everything we've achieved in VB has been easy for us. We wanted to shorten the scroll bar - once! - and her short line of code is sd e Lali. We wanted to hide the switch element - once! - and made it page on Coy , consisting generally of two words. You had enough to know the name of a simple "tool", which you forth on the scourged and were purpose - *the .Width, Hide Event* , etc

.But not everything in VB is made that simple

Remember the fairy tale, where Prince Ivan had finished with Kashcheev Bessmer t nym? To do this he called hell of it was to break the needle. But to get to this needle was d about the quite difficult , because the needle is something - in the egg, and the egg - in a duck, and duck - in a hare and a rabbit - in a casket. If the box is in your hands and you know the structure of this "matryoshka", then it is another matter. Rather, half the battle. Half the battle, but not the whole point . Because the hare will jump from the casket into y - you need to catch up, the duck will jump out of the hare , and into the sky - you need to shoot it, etc. In a used present - darkness. So it's not in vain e ! Through this Moroku we much understand in programs . vp of Vania on the VB

If you want to VB something with overshi be some catchy "needle", is to find it, you will need about doing all the way deep into the "casket". We have already partially faced with this situation it when searching for "needles» *Yellow* instead prospect of grained

~~*Button2.BackColor = Yellow*~~

we had to write more complicated

.*Button2.BackColor* = Color .*Yellow*

Here the additional word *Color* is the "egg" in which the "needle" *Yellow was* . True , yes, VB forgave us the "duck" and so on , did not demand. But he is not always so kind. We need to know exactly how to get to the right "needle" . And also, when we forgive, and to a GDS no . And this is .our conversation

Deep into the casket . When programming for the .NET platform, the " casket " is the just mentioned .NET Framework Class Library . About Opens the lid of the casket, we will see in two other caskets. They is called th tsya **spaces s name** . (n e will yet go into the literal meaning of the name , for ck and Zano: " Though a pot name, only in the furnace do not put !") . Each namespace has its own name . One of the two mentioned is called **System** ! and the other is **Microsoft** . Each of them - a real treasure th poss w Nosta .Create a new project. Let 's go all the way into our caskets now

To find out what is inside from the pros transtv Name *System* and *the Microsoft* , Ven and accurately write their name with a dot. Create a button, double click on it. As soon as the procedure is prepared, we will try to write . and write something

Beep .2.1.1

Let's take an excursion inside one of these two "chests". And to tour b u la not aimless, pose the problem : And zvlech of the computer and pressing .(the button koro t cue sound (ringing peak

I know that this sound is inside *Microsoft* 's box . And if you didn't know? - I would sit down to read a textbook or reference book, or the *Help* system . This chapter is not in any way will not tell you , Where is I tsya objects . There are too many objects for this. She just pa to tell you how to .get to the desired object, the location of which you already previously known

N Make a written word *Microsoft* and set point. N Before you Expand a

camping sleep juice , pre d stavlyayuschy contents namespace *Microsoft* . It 3 .(page of a and (Fig. 2 . 1

Figure: 2 . 1

Each of these 3 lines is nothing more than another namespace. It turns out that inside the *Microsoft* “box” there are 3 other smaller “ *boxes* ”: *CSharp, Visual Basic, Win32* . In a sign that this is the "l and rec", that is, the namespace, and not something else, in a line item on the IT icon of two . **{ }** braces

Let's take a look inside the *Visual Basic box* . (Why is it inside of it - because I know what it takes us to the sound of sitting there.) To do this, make a double click on sootvets t vuyusche m spaces ie names and again press the point. Before n s will develop a list of what is included inside *the* .(*Visual Basic* . P rokrutim n e a lot of this list (Fig. 2 . 2

Figure: 2 . 2

We see that the inside space of the names of *Visual Basic* includes another space consisting of tons of names - *the Compile rServices* , inside which is also probably something there. But we don't need to go there. Because to rum it we see in the list , and many other lines , the notation n GOVERNMENTAL other icons. We will not understand the meaning of these icons for now , but double-click the *Interaction* line and press point again . Before you unfold the joint venture and the juice that Rin on dit inside *the Interaction* . Let's select the line *Beep* . This is our "needle". This is what

makes the computer emit a short sound. Here's how to look is now your : facials at RA

```
Private Sub Button1_Click ( ByVal sender As System.Object, ByVal e As
System.EventArgs) Handles Bu t ton1.Click
() Microsoft . VisualBasic. Interaction .Beep
End Sub
```

Run the project, press the button, and if your computer sound system is okay , you will hear a short beep. We say that the load m e Tod *Beep* object of *the Interaction* (n on analogy with the way in 1.3.9 , we said that the load . ((method *Hide Event* (he pr i tal switch elements

So, a needle, an egg, a duck and a hare are present in the code . You will say: Is it necessary to spare on Mina are so many names of these "animals"? Not at all. You need to know only the name of the "needle", and the help system (4.4.9) or *the Object Browser* (4.2.4) Tell me have so you .where she tries to ditsya

If you were attentive, then you probably have some more questions. I hope to answer them , and for this the following example rassm atrivayu . more on Dr. detail

Debug. WriteLine .2.1.2

Suppose you need a tool to display numbers and text in the special window of the VS , to about Thoroe called *the Output* . Why such a .conclusion is necessary, you will learn on of the same

To find this tool of aglyanem in "casket» *the System* . Type this word and period at the beginning of a new line within the procedure . You will be presented with a list of what is included in the *System* namespace . We see that to include other space consisting of tons in a name , for example *Diagnostics* . Besides Mr. them we see and many other elements , n of IU chennyh vari mi icons. We will call them *objects* . In consequence Detect you have zhite that most of his they do not like what we called objects earlier - the .form and elements of sound control in Lenia

: Objects are of different types . Specify it nek about tori e types

kla s s s

structures

modules

n enumerations

These are not “needles” yet, but something like “eggs”. All of these sites - important Instr at cops programming, containers useful properties, .methods and other relevant in e soup

But our road leads past the - ext s namespace *Diagnostics* . Sel e Rita said it in the list and enter the point. Before you unfold the joint venture and . juice facilities , which entrance I inwardly namespace *Diagnostics*

Select iterated here Class *Debug* and again enter the point . Before you except p netsya list of what class has *the Debug* . Let's call it **the components** (**members**) class. Note that the icons are already different: no stem with owls or modules or other objects. And then there's what these objects have , that is to m nents . In the case of the *Debug* class, these are properties and methods. Here are the icons most widespread on roubleshooting components :

properties

m practice methods (many of the methods

(referred to as **fe to tions**

events

These are already real "needles". We choose method *WriteLine* , in brackets after the n and write *3 + 2* . By the way, after entering the opening bracket VB helpfully offer podska of ku on the subject of what should be in .brackets. Over time you will learn how this podska of Ku understand

: This is what our procedure looks like now

```
Private Sub Button1_Click ( ByVal sender As System.Object, ByVal e As
System.EventArgs) Handles Bu t ton1.Click
() Microsoft.Vis ualBasic. Interaction .Beep
( System.Diagnostics. Debug .WriteLine ( 3 + 2
End Sub
```

If you do not see the *Output* window on the screen now , do **View ↓ O ther Windows ↓ Output** . Run the project, in the window *Output* you SW and wee emerging lines overhead associated with running and compiling your project. Then, on the screen , I vitsya form. Press the button, you will hear a sound and the number 5 will appear in the *Output* window . That's right. End the project. In the *Output* of I vitsya line of proprietary information .related to the shutdown of your prospect about EKTA

conclusions .2.1.3

What are the main points you need to esti consideration of these two :note e moat? There are two of them

1. Libraries and .NET Framework classes including e r an enormous number of objects, their properties, methods, and other elements. To all the .district can be reached pa from looking above manner

2. P azberemsya in the structure of the two written statements. Each of : them consists of three parts, separated by dots. Come right cash ie in

- **The far right is a *property or method,* with or without .parentheses**
- **To the left, through the dot, there is *an object* - the ."owner" of this method or property**
- **And the left-most part of - one or more namespaces, see n .GOVERNMENTAL points**

Remember this simplest architecture u. Soon we will shorten it and and .are different for generators

Namespace m You can called s Vat so "namespace *Diagnostics* , Nachod I scheesya in the namespace *the System* ». A can be called s Vat shorter "space consisting of m in the names *System.Diagnostics* ». They say even this: "Triggering method *WriteLine* object *Debug* of pr of space names .« *System.Diagnostics*

More points . That in front of us turned the list, we need to start recording with the words *System* , and whether *Microsoft* or we know the name of the object that is included in the above I mentioned namespace. And why else? In 1.3 we started s and pis to the name of the control that is already on your form, for example, *the Button1* . And with the "pronoun e Niya" forms of *Me* . You can start with the name of the project, but we don't need .that yet

Let's get "by all the rules" to yellow. Add the procedures as follows th :conductive line

```
Button 1. BackColor = System . Drawing . Color . Yellow
```

During its introduction, you learned that *the Color* - it turns out, the object namespace *the System.Drawing* , and the *Yellow* - its its first GUSTs.

And before we wrote shorter. When you can write shorter, see the next .section

Not all class library .NET Framework is currently connected to in and Shem project. This was done to save money. If you already have the experience and you know exactly what you are missing, you can quickly connect additional space consisting of t in and the names of the library (see . (para. 21.3.1

In the future you will see that the properties and methods themselves can be "objects", that is, on the GDS after them, too, can put an end, and so to .is infinite hours Nost

Save ink .2.2

Still, our operators are very long. Is it possible to record their claim on .the shorter? Can. There are two ways

The first way is to omit the namespace names .2.2.1

In most cases, VB allows you to throw out the operators name space : consisting of Mr. STV and names . Try

```
Private Sub Button1_Click ( ByVal sender As System.Object, ByVal e As
System.EventArgs) Handles Bu t ton1.Click
() Interaction.Beep
(Debug . WriteLine (3 + 2
End Sub
```

.Everything went fine. The result is the same - 5

The *Microsoft.V isual Basic* namespace allows you to drop even the : name of an object. P on the sample first ones

```
Private Sub Button1_Click ( ByVal sender As System.Object, ByVal e As
System.EventArgs) Handles Bu t ton1.Click
() Beep
( Debug.WriteLine ( 3 + 2
End Sub
```

. It worked

But sometimes the name spaces and names can not be thrown away : the VB objects. This means that there is an object with the same name in

other namespaces, and VB objects because it doesn't know which of these objects you mean. If the two companies Kolka, then to avoid confusion of all .the name Kolka with Polyanka, and another - with Kolka Basman

Getting to know some objects .2.2.2

D To train and useful contacts with some of the functions you must burn in the process a few more lines . But from the beginning you will be examined in the notes , etc. and administered by below , and so then : recording the first ones . This is how the procedure should look like

```
Private Sub Button 1_ Click ( ByVal sender As System . Object , ByVal e As
System . EventArgs ) Handles Bu t ton 1. Click
        (Debug.WriteLine (10 - 2 * 2 + 30/3
        Dim a = 4 - 3
        (Debug.WriteLine (a
        (Debug.WriteLine (a + 1000
        ("!Debug.WriteLine ("Hello everyone

        (Dim b = System. Math.Abs (-20
        (Debug.WriteLine (b
        ((Debug.WriteLine (System.Math.Abs  (-20
        ((Debug.WriteLine (Math.Abs  (-20

        (("Debug.Write (Microsoft.VisualBasic. Strings.Len ("World
        (("Debug.Write (Strings.Len ("World
        (("Debug.Write (Len ("World

        ( Debug.WriteLine (Len ("War") + Math.Abs  (-80) + a + 1
End Sub
```

When you s Run the project and on zhme those button in the window :*About utput* you will see the following result

```
sixteen
1
1001
!Hello to all
20
20
20
33387
```

Before you go to read the explanation, read **about chen useful with**
: **a vet to work on the program s**

When you check how the program works from the book (for example, just, etc. , and management), or how the program works, created by you, never enter all prospect of the gram as a whole, then to .run it and see how it runs entirely. Enter and run little by little

The reason is this. Every programmer , even the most experienced, as you type, etc. about grammnogo text allows oshi b ki and simply slips. And he does not consider it a big sin. For 100 lines of the program, errors 5-10 are the norm. Not all of these errors will be detected by VB . P of this, whether you like it or not, run to the execution of the program you have to err , and non-timber. When you get the results of the wrong program, they, too, horseback hours but will be wrong. And you, looking at these results, have to guess what the mistakes are . It is difficult to look for errors in a long !program, and even when there are not many of them

Do this. We entered the first two or three lines of the program, launched the project, looked at the results, checked whether they were correct or at least plausible. If everything is in order, you can be sure: in these lines ERROR ok probably not. Added to them cl e blowing two or three lines, again launched a project checked out. Etc. Each time, the nut 's repenting on the erroneous result, you can be almost sure that the cause of it - in item of .Latter 2-3 lines of code

Another reason: it is easier to study the material and of books. In order not to break your teeth on the granite of science , you need to bite off a little . from it

With all this in mind, proceed to reading the explanations, .simultaneously entering one line of code and launching the project

: Explanations for the first 5 lines

```
(Debug . WriteLine (10 - 2 * 2 + 3 0/3
```

This operator prints the first number (16). After the word **WriteLine** you have to post a twist brackets and parentheses to indicate exactly what you want to display. Here, the asterisk * for a significant multiplication and slash / (**slash**) - division. Thus, in parenthesis, one hundred I boiling the .method name *the WriteLine* , you can write any ari £ meticheskoe expression

```
Dim a = 4 - 3
```

.Here we declare (**Dim**) a variable *a* and set it to 4-3 = 1

```
( Debug . WriteLine ( a
```

The *a* value of 1 is displayed in the *Output* window . Thus, in the parentheses behind the name of the WriteLine method , you can write variables and ranks

```
(Debug . WriteLine ( a + 1000
```

, and any arithmetic expressions containing variables and us

```
("!Debug . WriteLine ("Hello everyone
```

.as well as any character string enclosed in double quotes

I explain mathematically e function and

```
(Dim b = System . Math . Abs (-20
```

Here we declare a variable *b* and piles Vai s module [†] numbers - 20. How do we do it? The module is a mathematical function to to m pewter we calculate it, we need to VB find the appropriate tool. Write after the equal sign called a namespace title of *the System* , then point and the RA of the bounce class list, select **the Math** . He is to be endowed with five of us different Math and ical functions. Then again, write and point in the ongoing list of selectable m m e Tod **Abs** of this class. It is he who calculates the module. In parentheses after the NSP eat - 20 [‡] . I remind you that many of the methods are called functions, as wo b schayut to us as something a .number or other information

```
( Debug . WriteLine ( b
```

. The *b* value of 20 is displayed in the *Output* window

```
((Debug.WriteLine (System.Math.Abs  (-20
```

This operator is also displayed in the window *Output* number 20, that is, doing what before s duschie two operators together. I wrote the e nd to dormancy and to show that it was possible to manage and without a variable th *b* , because in brackets behind it is, it method *the WriteLine* , you can write .and appeal to mathematical Fung to tsiyam

```
((Debug . WriteLine ( Math . Abs (-20
```

Here I threw out the name of the *System* namespace and everything . worked out - another number 20 was displayed

Let me explain the work with textual information : Often we need to know something about some text. H as an example, even if, as in the .letter

```
(("Debug . Write ( Microsoft . Visu alBasic . Strings . Len ("World
```

Function **Len** Class **Strings** Namespace *Microsoft.VisualBasic* finds many characters (letters, digits and other spaces) Prissy t exists in the line, which is its and p argument of. The word “World” has 3 letters, so the number 3 is displayed in the *Output* window . Note that here I am not using the *WriteLine* method , but the *Write* method . Ra s maths is that for our next troika information in the window *Output* will be printed on the same line as .the 3 , but not in cl e blowing

:M ozhem writing shorter

```
(("Debug . Write ( Strings . Len ("World
(("Debug . Write ( Len ("World
```

:The last line will calculate and print 5 + 80 + 1 + 1 = 87

```
( Debug.WriteLine (Len ("War") + Math.Abs  (-80) + a + 1
```

Second way - Imports .2.2.3

And yet I would like to be even more laconic. Operator **Imports** item on allows one to omit the names of not only the namespace, but also objects , they included : classes, modules (not to be confused with the abs of lute value) , structures, enumerations . Let's set the task to omit the names of the classes *Debug* and *Math* in our program . For this very ver 's her line in the : box, the code you need to write operas and torus *an Imports*

Imports *System.Diagnostics.Debug* **,** *System.Math*

After the word *Imports* through the names appearing in the comma objects (in our case classes *Debug* and *the Math*) with the obligatory indication of the namespace in which they Rin of DYT. Here's how our code window will now look (for the sake of brevity, I left in facials at 're only two :(lines

```
Imports System.Diagnostics.Debug, System.Math
Public Class Form1
    Inherits System.Windo ws.Forms.Form
    Windows Form Designer generated code
Private Sub Button1_Click ( ByVal sender As System.Object, ByVal e As
```

```
System.EventArgs) Handles Button1.Click
        (WriteLine (3 + 2
    ((WriteLine (Abs (-20
End Sub
End Class
```

As you can see, now instead of *Debug.WriteLine* you can write . *WriteLine* , and instead of *Math.Abs* - just *Abs*

But what about now with ambiguities, when the objects with the same name Rin about DYT in different namespaces? In this case, refuse *Imports* or favor th Tes its complementary and -inflammatory abilities, which I do not .stop

In the future, I will write fragments of programs, where omitting and where not omitting object names. Where the names are omitted, it is assumed . that the used sootvets t vuyuschy operator of *an Imports*

Chapter 3. Fireworks of Opportunities

In this chapter, by creating two projects - By alkulyatora and P lifelines - I want to introduce you to a simple and pleasant features the VB . Mr. lava necessarily has n e as a promotional tone. A self-assembled tablecloth will open before you, on which you will see basically what is tastier and does not require chewing. However, among the UWC with Nogo everything useful, and almost everything is completely necessary . Despite the lightness of the material, e is the head serves as the foundation for the rest , so read it " by . diag about Nali " does not, without it will not be clear further

" Task for the project " Calculator .3.1

In Chapter 1. we have not created anything worthwhile, but we went . from beginning to end all the way to create a project in VB

Now we have set ourselves the task of doing the real thing - cal s kulyator. Roughly the same as in Windows . If you forget it, in the Windows (and a ton by no means a in the VS), follow these dis t hold: **Start ↓ Programs ↓ Accessories ↓ Calculator** . Count something in it. How do you - a lot of effort spent company Microsoft to create it? The calculator that we will create, in the sense of mathematics , would be simpler for children, but it will be beautiful, equipped with music, a password and various other . jokes

Let's start with the fact that we come up with an appearance (still . modest) for our calculator - see Fig. 3 . 1

Figure: 3 . 1

The idea is - you type in the top two text fields, the first and the second number, then click on one of the buttons in the lower text box floor at chaete .result

We design .3.2

Create a new project and place in the form of three text boxes, and h e . tyre button. You will get as in Fig. 3.2

Figure: 3 . 2

To proceed further, we need to become better acquainted with the .properties and E forms and controls

Properties window .3.2.1

A little earlier we already got acquainted with such object properties as

BackColor , *Width* , *Text* . Each object has a fairly large number of properties . Many of them we can change in operation [*the run*] with pom on the program soup, as we have already done (for example, *HScrollBar1.Width = 100*). It turns out that many of the properties shapes s and ale n comrade control can be changed in the design mode [*design*] . Affairs and etsya it manually without any software and tion (for and intended, .(!that the hand easier

Are you in [*design*] mode now ? Good. Suppose you want to view Propert t wa buttons *the Button1* . One click on this button . In this case, markers appear on the button. They say that the object is **selected** , became **active** . In this case, the properties you can and put in **the Properties** .(**window** (Window *Properties* in the right part of Fig. 3.3

Figure: 3 . 3

If the window is not visible, do the **View Take ® Properties has the** . **Window** , and then again the slots to Nita button *the Button1*

You may have this window in a slightly different position. If it seems to you enough, try the races expanse be it, grabbing the l e vy or the lower edge of the I . For more details windowing environments s VS time on the brane in . 4.2

In the property - two columns: the name of the property and its value. For your udo b -OPERATION properties and may be ordered alphabetically .((tab *A lphabetic*) or grouped into categories (tab *C ategorized*

Select alphabetical order. Note that some properties are marked with a "+" sign. These are complex properties that are themselves composed of several properties. In Fig. 3 . 3 e then, for example, properties of the *Font* (a font). If you click ie those on the icon "+", it turns into a "-", and the properties from which the *Font* with a cost, will be seen individually, each - in his line. In Fig. 3 . 3, you can see what it consists of properties and *Location* (location) and *Size* (Size). For example, in property *Size* includes .(two Propert t wa: *the Width* (w and Rina) and *the Height* (height

Let's try to change the value of the *Width* property . Find the properties window it including Mr significance. Enter instead of the existing therein of a different number, n as an example, 30 . You have see that the width of the button on the form has changed. Exactly the same result you add and were be in operation [*the run*], running the operator *a Button 1.Width = 3 0* . And now let's do the opposite about the mouth - drag one of the button markers a little to the left or right. We will see that amended accordingly e nilos and the . number in the properties window

Select by clicking some other object on the form . Now in the properties window - its properties. Select the shape by clicking on any free spot on its surface - now we can see and change its properties. Etc. Poeksperime n .tiruyte

I will acquaint you with the name and gradual sense of the different .properties of n but

W The values of many (but not Telegram s) properties from the properties easily change a progra m IU operas and torus s type *Button* . *1.Width = 30*

Tags. With voystva: and on me , so No lyrics , .3.2.2 w rift

Name and text . Each object has a property **name** (**the Name**). Many have a **Text** property . Let's try to change them and find out of and what they need. In a previous project, we were blithely s and did not care about names and those to ste . For this, we will not be punished , because the VB in the case of our bespo h Nost (as they say - **the default**) himself gives the value of the property, including names and those to stu. P Rich, without thinking twice, the name and the text VB does the same. without wasps on - . singular reasons , just so convenient

Let's check. Spd e Lim form. Look at the properties window and find the name (*the Name*) - it is above all in the alphabet. We can see that the name of our form is *Form1* . It is the same word we and dim sum in the form header. Change the name of , say, the word *Kal s kulyator* . Changed. And .what? Nothing has changed in the form header

Attention! In ernite form of her name *the Form1* , otherwise the project will not be started (reason - about the Kommersant clear is . (Busy in 21.1.2

We find are now in the properties form the property text (*the Text*). Of course, the text on our form is also *Form1* . And of we apply e th at *My personal calculator* . Aha! These words appeared , and were in the form .header

Now let's take the first button. Let's select *Button 1* . Let's look at the properties window. We can see that the name and text of this button are the . same - *Button 1* . D and dim her name *Kl_slozheniya*

<u>Attention!</u> The name must contain only letters, numbers and signs subordinated p nodding, and the name must begin with a letter .or underscore sign and Niya . It turns out that shyness is prohibited

Since spaces are not allowed in names, I used the underscore and . underscore instead

The above rule applies not only to the names of the objects, but also other e e cops VB - properties, methods, variables, namespaces, etc. For . more information on them , see 5.3.4

Our button text should consist of a single + symbol . We find the + on
. the cells and viature and introduce its property value of *the Text*
I think you can already guess why the name and the text are needed and what is the difference between them. On the surface of the object, we see not the name, but the text . Text view en all (Stirlitz) , and and on me remains so Xia hidden (Col. Isaev) . Another analogy: Mal s chick walking down the street, on his shirt - text " E mmanuil". And his name was Nick, but it's Ruba
.w ke not written
Why do I need a name? Then, that the program we refer to the object it
.is for them ie no, not in the text
Why we changed the text is clear to us . That all could see comfortable seq of the Islands or symbols. Why did you have to change your name? Well,
at least because the op e Rathore

Cl_addition . Width = 60

is somehow clearer than

Button 1 . Width = 60

Tags . A Now for explanatory inscriptions on the left side cal s kulyatora (*first number* , *second number* , *Result*). For this we need a new control element - **the Label** (**tag**), which is mainly explained for e Nij and applied. Find *Label* in the *Toolbox* window and place three labels on the
. shape. P of them ka have the form: *Label 1* , *Label 2* , *Label 3*
Please set the following properties *Name* and *Text for* all project objects
:

Text	Name	An object
My personal calculator I'm tor	Form1	*Form1*
+	Cl_addition	*Button1*
-	Subtraction	*Button2*
(**x** *(it's just the letter X*	Cl_multiplication	*Button3*
:	Cl_sections	*Button4*
	Number1	*TextBox1*
	Number2	*TextBox2*

	P e result	*TextBox3*
First number	Label1	*Label 1*
Second number	Label2	*Label 2*
P e result	Label3	*Label 3*

Explanation: Because we want to begin using a calculator in the text of O fields were empty, erase in the properties value of the property *Text* for all three claim to lei. M 's do not become obstructed and change the label name because the program their names did uch as update themselves will not . When you run into ash project , it should look et well as in Fig. 3.4

Figure: 3 . 4

Sh rift . Suppose we want the word "result" was maybe more s Shae and fatter. Click on *Label 3* . H aydem its Propert m in a font (*Font*) and click plus. A dozen properties will jump out of the property (font size, weight, etc.), and you can change each of them. But personally I think more Udo b nym other cn of sob : Click it on the property of *the Font* . A button with an ellipsis has appeared in the value field - this is always an invitation to further conversation (you can see this button in Fig. 3.3). Click on it - there was a so-called front of us **the dialogue of howling window** (Fig. 3 . 5), offering to customize the font of the text, that is, choose a font size (*Size*), Draw and of (*the Font s tyle*), name (a headset , that is, drawing letters) the font (*the font*) and something others in Goa. Inscription may be CONVENTIONAL (*Regular*), *ITALICS* (*Italic*), **BOLD** (*Bold*) and ***BOLD ITALIC*** (*Bold Italic*). Among the headsets, there are curious ones. Choose a size and . stuff to taste , then n and press *OK*

Figure: 3 . five

The arithmetic symbols on the buttons are too small. Increase their size .and make them fatter

All! The design of the first version of the calculator is complete! The calculator now looks almost as shown in Fig. 3.1 . You can proceed to .Programming and NIJ

We program . The project is ready .3.3

Let's start the project. Let's enter the number 3 in the upper text field, and enter 2. in the middle one. Click on the add button. Nothing happened. Naturally. After all , we did not write any foolish procedure for the button. Complete work proe to that. Double-click on the add button to create a template for the procedure in the code window . Since any knowledge we have, n about trying to invent what is written there. We will reason like this: 3 and 2 are nothing more than the values of the *Text* property of the *Number1* and *Number2* text fields . In other *words* , it is *Chi with lo1.Text* and *Chi with lo2.Text* . We need to *Rezultat.Text* equal to their sum. What if you write such

:operas as a torus

Result.Text = Number1.Text + Number2.Text

: It is said - done . We get

```
Private Sub Cl _ addition _Click ( ByVal se nder As System.Object, ByVal e
As System.EventArgs ) Handles Cl _ addition .Click
    Result.Text = Number1.Text + Number2.Text
End Sub
```

We launch the project. Enter 3 and 2. Click on the add button. The results s tat there. But not quite what we expected. Instead of 5 has turned 32. What prich and on? The fact that VB used to consider the sign **+** in relation to the contents of the text boxes are not familiar complicated e Niya, and the sign " Connection ", lining up the chain. Check it out. Instead of 3 and 2, enter *Koro* and a *barrel* , the result is a *Box* . It is not for nothing that text fields are called text, not numeric. What we called them *Number1* and *number2* , the case does not claim to could , because VB does not pay attention to the meaning of names, names for him - just meaningless .combinations of si m oxen

What to do? It is necessary to order the VB apply in this case, the contents of the text of O fields , not as text, but as with numbers. For this - record enough it is not so and anyone form

Number1.Text Number2.Text

- and t and com

(Val (Number 1. Text) Val (Number 2. Text

Here the function **Val** - reduction of *the Value* - the value, the :numerical VALUES e set. Now, our operator will u look like this

```
(Result.Text = Val (Number1.Text) + Val (Number2.Text
```

We launch the project. We introduce any two integers and make sure .that all the warehouse 's INDICATES correctly

We program the other three buttons in the same way. Remember my advice about not entering more than 2-3 lines into the code window at a time. Entered subtraction procedure, ZAPU with Titus the project is to test how it works. Now we introduced the multiplication procedure, launched it, checked .it. Etc

: That's as a bud from t Vaglen I do after that 4 received treatments

```
Private Sub Cl _ addition _Click ( ByVal sender As System.Object, ByVal e
As System.EventArgs ) Handles Cl _ addition .Click
(Result .Text = Val ( Number 1.Text) + Va l ( Number 2.Text
End Sub

Private Sub Cl _ subtraction _Click ( ByVal sender As System.Object, ByVal
e As System.EventArgs ) Handles Cl _ subtraction .Click
(Result .Text = Val ( Number 1.Text) - Val ( Number 2.Text
End Sub

Private Sub Cl _ Multiplication _Click ( ByVal sender As System.Object,
ByVal e As System.EventArgs ) Handles Cl _ Multiplication .Click
(Result .Text = Val ( Number 1.Text) * Val ( Number 2.Text
End Sub

Private Sub Cl _ dividing _Click ( ByVal sender As System.Object, ByVal e
As System.EventArgs ) Ha n dles Cl _ dividing .Click
(Result .Text = Val ( Number 1.Text) / Val ( Number 2.Text
End Sub
```

So, the calculator is ready! Test it properly . Enter Negative and :fractional s nye chi with la. Wherein

When you enter in text fields instead of decimal comma s those . points . The result will be displayed with a comma

Cautions : Our calculator is not yet protected from the input instead of numbers any nonsense (eg, text, *KU-KU*), from entering too large or If u com small numbers, dividing by zero. In such cases, VB but with so we wrong or neudoboch and Ty result. You will find protection and . improvements in 5.4 and 7.7

Complete the following two tasks. They allow you to enrich your calculator with new features . You will find answers to the tasks at the end of . the book

Create button squaring the number of the top ones in Stow field. Hint : .To square is to multiply by itself

Our calculator lacks a *RESET* button that would empty all three text fields. Create it. Note : For this you need oper about ry type *Chislo1.Text = ""* . In the code window, the text content of text boxes must be enclosed in double quotes. In this case we have inside the quotation marks empty, as .required and elk

From properties from the properties .3.4 window

Let's start in design mode to improve and decorate our cal s kulyator.
For this n s need and tsya to change some other properties of shapes, buttons, text on the O fields, labels - facilities , of which calculator is designed . Rassm t Rome these properties . Many of them are available and in most other controls , nekot of rye - only one or two. After reading the material on the next property you breakdown e INH then experiment to see how its different values affect the appearance and dressing f denie project objects in p . [ie bench [*the run*

Good advice : After several experiments, your project will look like n of admixture parrot with a porcupine . Besides, the objects are many , and others from GIH properties that I have not explained yayu and n rirodnaya curiosity pushes you " popered Batko " razobratsya s camping, what do they mean, giving them to this random any value. No problem - rack your brains and your health project . The problem is that someone tries *to about store* design with the changed values of unknown properties. And then he can't break off the pop at the guy's horn . The way out is this. Copy the project folder before starting experimenting. And exp e rimentiruyte on a copy. " A sense of . " ! security about the STI

General properties .3.4.1

Now I will list popular and lightweight properties of familiar objects and IME n but: forms, buttons, text fields, labels . Others you understood and dobyatsya not soon. I repeat that e ti properties are and have mnogi x drugi x .element s controls e Nia

We have already met in the properties window with the properties

.**Name** , **Text** , **Wi dth** , **Font** . Let's get to know others

Appearance

The BackColor - the color of the surface of the object - the SA and komoe property. Select the shape. Let's click on this properties . The field value is displayed and familiarize a button with a black triangular arrow. There are such buttons in many properties for convenient selection of their values. Click on it - the window appeared in front of us with tre on me of a . (stem d kami (Fig. 3 . 6

Figure: 3 . 6

Select a color tab ah *Custom* , or *Web* and click on it. Forms, buttons, and labels (but not text boxes) all took on this color. Highlight button and similarly the mouth and novite e nd some other color. Do the same with the .label and the text vym n of Lemma. We 'll talk about the *System* tab later

If the proposed colors you are missing, right-click on a e Nome of white squares in the lower part of the tab *the Custom* . You will have a color .(definition window (see. Fig. 3 . 7

Figure: 3 . 7

By clicking on the large multicolored square you select a color, and moving the slider up and down on the right side of the window select the brightness of the color. In the *Color* field | *Solid* you can see a sample of the resulting color. Click *Add Color* .

ForeColor - the color of text and lines drawn on the object . Configurable similar to *BackColor* . Try to make the text field as l fifth with blue ones to the ostomy.

BorderStyle (for text boxes and labels) - border style. About n redelyaet border character and volume of the object type.

The FlatStyle (for buttons) - roughly the same as the *BorderStyle* for text fields and th e current.

FormBorderStyle (for forms) - on the one hand, this prospect and approximately the same as the *BorderStyle* for text fields and labels. But on the other hand h ere 7 options, and the head is, blow them not only the style of the border, but also the number of buttons in the upper right corner of the form s , and the ability to resize the form in p e bench [*the run* .

N Assumption and size

Size (size). Consists of properties **Width** (width) and **Height** (height). The height of the form and controls is configurable in exactly the same way .as wide and not. Try it

. Specify a location using coordinates, we learn a bit n and , in 3.5

Locked (locked). Typically the value of this property is *False* (which means False) . If you set it to *True The* (what it means - true), then the element of control in p e bench designs of Bani will not be able to carry on the form and change its size (and shape - only resize) . This is useful if you are afraid of sl in the tea mouse movement to displace carefully installed . [§] element control e Nia

Anchor (anchor, anchor). When we resize the form in [*run*] mode , the controls remain in place . This means that they retain me by their distance to the left and top edges of the form , instead of to the right and bottom . We'll change this situation for this control by changing its *Anchor* property . And of menite, for example , and measures the property for the button. Clicks on each of the strips 4 (Fig. 3 . 8) you order s INDICATES maintain or not to maintain a constant distance from the button to the corresponding edge of the form. Make all 4 stripes dark for interest . W and empty project. Resize .the form and watch on Poveda e Niemi button

Figure: 3 . 8

Dock (dock, stick). Typically, the value of this Propert t wa - *to None* (which means - None) . If you opt for one of the control element pre forth false values of this property, it will be "glued" to one of Cr and s shape as we called and changed its size. This is how strips of twist are usually glued to the . edges of the window

Other properties

Cursor (cursor, mouse icon, mouse pointer). Floor of the properties of OAPC of A e t you change the appearance of the mouse cursor when it is

. over the object

You Affairs ite button by going to the Properties window, select the ones to its properties *Cursor* any value from the list (for example, the cursor in the shape of a cross). Run the project. Premises with Titus mouse cursor .over the object. Cursor gained appearance cr e a hundred

If you specify the form of the mouse cursor for the form , it will have a predetermined view over the entire surface of the mold and those objects on . (the form, for which he has not, and of Menen (except text fields

Played with mouse cursor icons? And now return everything as b s lo.

You shouldn't get out of the habit of the standard interface[**] - it scatters . attention. Exotic x o rosh in Africa

Visible . Usually the value of this property is *True* (which means True). If you set it to *False The* (meaning - Lies), the element management per e .will be visible during operation. But will obey the programs of

Enabled (operational). Usually the value of this property is *True* . If you set it to *False The* («false») , the object is established in nerab of Others state that there shall be seen, but it will not work and they will not be able Custom s camping . Thus, it will not be possible to enter text in the text field , the button cannot be pressed, the text of objects will acquire a pale gray tint. But the programs have the object will listen. In a thanes inoperable form, you .are set outside with a distance all the elements of sound control in Lenia on it

TextAlign (alignment of text on the surface of an object) . CREATE th those high and w and rokuyu button and select it in the properties of one of .(the 9 variants of values of Propert t Islands (Fig. 3 . 9

Figure: 3 . nine

Depending on the value of the text on the subject is, or in his left ver 's it right .or at the bottom right, or elsewhere

Simultaneous setting properties in several control elements

e Nia . Obviously, all 4 calculator buttons must have the same pa of measures, font and color. Setting them up separately for each button is boring. VS makes it easy to set properties for multiple controls at once. All .you need to do first is to select these few elements

To select multiple elements , simply frame them. Or by clicking on the . first of them, the slit L kite at rest while pressing *Ctrl*

After elements are selected, the properties window only shows those properties that are common to all selected elements. A property values show s vayutsya only where they are the same for all selected items. For example, its minutes GUSTs *Location* (location) shows no values since the location in the .book about the dormancy on the form is different

Now change the *Ba ckColor* property in the properties window . It will change all apportionment of n elements. Write in the field of values of the *Size* property two numbers separated by semicolons: ***100; 70*** - all selected .elements etc. and will find the specified resolution

Group selected elements allows to drag yourself in shape and to change the posted e ry elements by moving the mouse. The selection is deselected . with the *Esc key*

Text box properties .3.4.2

Multiline (multiline). When you create a text field, znach e of this Propert t wa - *False The* . E it means that the text field can not enter more than one line of text , and you're cell field can not do more than one line. If a thanes it in *True The* , then all this becomes possible . Install. In velichte height field. Try to work. The mode [*the run*] is the text field behaves like a document window, etc. of the simplest text Reda to the torus. Enter several lines of free text into it. The right mouse button allows you to perform basic .operations Edit of Bani

WordWrap (folding). This property is currently set to *True* . This means that when you type a line in a multiline text box, the text reaches the right edge of the field, entering , as we are accustomed to, will automatically continue from the beginning of cl e blowing line. If set to *False* , then the .input will continue on the same line, scrolling to one hundred

ScrollBars (scroll bars). To make it easier to scroll through the text in the BUILD h nom text field, you can provide it with a vertical scroll bar, and .if the property *WordWrap* is set to *False The* , the horizontal and b Noah

ReadOnly (read-only) . When you create a text box, the value of this property is *False* . If you set it to *True* , then you cannot enter information there manually , but only programmatically. I think it makes sense to SET on the twist is its first GUSTs in *True* text box , intended to result , in your To

alkulyatore. True, it will change the color, but you can back it Restore on vit, . asking its second GUSTs *the BackColor*

CharacterCasing (case). You can make all the letters in the text box to be uppercase or lowercase , despite the fact that enter you into it (with . keyboard or software) , and those and others have Gia

Form properties .3.4.3

WindowState (window state) . About predelyaet in what form in s arises on the screen fo r ma startup project: normal, in a minimized (that is, you will find it on the taskbar, the Windows) or maximized (full e to the .(wounds

StartPosition (start position). About predelyaet, in any place of the screen there is a form when the project starts . So, you can catching and be it will appear s camping in the center of the screen (*CenterScreen*) or give yourself the opportunity to hand (*the Manual*) n and the tunable place of occurrence forms using properties and *the Location* (see. The next pa of cases .(

MaximizeBox (the maximize button is a square in the upper right corner of the form). This property is currently set to *True* . If your tired of the twist in his *False The* (False), the box will be unavailable. Property .*FormBorderStyle* can completely remove this quasi d ratik

MinimizeBox (m button yn imizatsii - horizontal dash in the upper right corner of the form). Now the value of its second -OPERATION - *T rue* (True). If a thanes it in *False The* (False), the dash will be inaccessible on . . Property *FormBorderStyle* m about Jette completely remove the dash

MinimumSize (minimum size). Sets min and mal width and height form, which form can not be less than the decrease in p th bench [*run*] , and . in the design mode

MaximumSize (maximum size). Sets the max and mal width and height of the form, which is more than a form can not be increased in Dir and IU [. *the run*] , and in design mode

Icon (icon, pictogram, icon). In the upper left corner of the form you and wee icon. You can see it on the taskbar when the project is running. If you don't like it, you can replace it with another one. Click on the *Icon* property . A button with three dots appeared in the value field. Click on it - the usual window for opening a file will appear in front of you . M nozhestvo

icons you can on the first tee it at

Program Files \ Microsoft Visual Studio . NET \ Common 7 \ Graphics \ icons

ShowInTaskbar (show the icon on the taskbar). You m of zhete prohibit icon shapes appear on the taskbar by setting this property to *False* . *The*

Opacity (opacity). Very picturesque property, which was not in sight s duschih versions Visua l Basic . Now it is set to 100% and the form after launching the project is completely and habitually opaque. Set it to, for example, 50 % and run the project. Form of Art and there is a translucent (.(see. Fig. 3 . 10

Figure: 3 . ten

With a value of 0 %, the shape will be completely invisible. In 12.9 we set the task of m and terializatsii ghosts , and now you can do quite a !translucent our Cal s kulyator. And what is very convenient

The position of the object on the screen. .3.5 With YSTEM coordinates

To draw shapes to place objects in the right spot on the screen, or fo r we to animate moving objects in the right direction, you should be able to about b yasnit computer y, where , in what place the screen or form you want Naris about Vat figure or place an object. Learn it , you can quickly , having an idea of the coordinate system. But e If you do not know what it is , then . here's the whole point without the use of Mon i ment

In Design view, in the Highlight any object on the form , click in the properties window of a plus sign in its properties **the Location** (location) and . Zagli I Nita in components it its first -OPERATION **X** and **the Y**

X **is the distance of the left edge of the object from the left edge of .the shape**

Y **is the distance of the top edge of the object from the top edge of . the shape**

Drag any object around the shape, keeping track of the *X* and *Y* values .
:Dobe th Tes

- zero in one property
- zero in another property
- zero in both properties
- maximum in one property
- maximum in another property
- maximum in both properties
- minus in one property
- minus in another property
- minus in both properties

Now increase the size of the form properly , place a button in an arbitrary place on the form and try to guess what its *X* and *Y* properties are equal to . A t e per check in the window e properties - guessed or not. Do this several times. If wrong aete smiling each time no more than 2 00, then all . smacking e ke you have mastered and were

The unit of distance on the screen in VB is the **pixel b** . At the same time the pixel f m is called the smallest possible point of light on the scr and not. Any image on the screen is made up of a plurality of pixels it . About pixel conveniently thought of as Malen com squares. E to the wound consists of pixel s as school notebook into the cell from the cells (see. Fig. 3 . 11). . You can see that the width of the letter *A* in the figure - 7 pix e l s

Figure: 3 . eleven

Columns of pixels it numbered from left to right, and the line - from the top down. Pixel s - a very small quantity. Its size depends on the resolution of VideoRen e benching. In the figure, the mode is 640x480. On modern .computers with large screens modes - 1280x960 and s Shae

.Read more about the screen image in Appendix 1

In pixel I x are expressed in values of properties *X* , *Y* , and *Width* . (width of the object) , *Height* (in s cell object) and other x

In design mode, we cannot drag the form around the screen by the heading, and this is not necessary. How to control where it appears on the screen after launching the project is described in 3.4.3 . If the *StartPosition* property *is* set to *Manual* , then we do this by changing the *X* and *Y* properties . In this case, the measurement is carried out relative to the left and upper . edges of the e to the wound

: **In the [*run*] mode,** we control the position of objects as follows

Button 1 . **Left** = 200
Button 1 . **Top** = 100
: Here

. Left and Top are the equivalents of the *X* and *Y* properties

Coordinate system

. At school, you are used to such a coordinate system as in Fig. 3 . 12

Figure: 3 . 12

. On the screen, computer a , you guessed it, is applied as in Fig. 3 . 13

Figure: 3 . 13

As you can see, the *y*- axis is directed downward. This is not very common. If you are that irritated and is then in 12.8 , you can program in your project a return to the usual system is, IU. However, hardly anyone takes this opportunity, therefore, you will be helpful, etc. , and become used . to the conventional computer system coord and nut

A la objects on the form of the start of the computer system of coordinates is in ver 's left corner of the form. To form the beginning of the . computer system of coordinates in the left ver 's it right corner of the screen

Commendable thing supply our calculator various tricks I bring to the :job to an independent district and bots

Nervous Key ". Suppose that when you press the subtraction and Niya, " this key jumps somewhere in another place and calculator on it instead of mines in sa text type appears, " I'm tired deduct " or " Do not touch me - I'm nervous! ". There's also a receptacle and considerably increase the w and widths buttons and the text on it will not fit. When you get it so camping, make sure that when you press the RESET button subtracting the modest returns to the scene and took its former appearance. Do not try to progra m .ized change the font size: you do not yet know how

Place photos on objects .3.6

Let m s want mold surface was covered like them someday Graphical e skim image, e.g., a photograph. For this qs about necessary that the file of the photo already stored on some drive of your computer. If there is no photo, then a drawing created by you in some graph and iCal editor, for example, in Paint, will do . In the extreme case, Neid is, those that - some of the address

Program Files \ Microsoft Visual Studio .NET \ Common7 \ Graphics \ bitmaps

.or in the Windows folder you will find the desktop wallpaper

Make a copy of your project folder and experiment already on it. Select the shape. Find the **BackgroundImage** property (the image in the background). Then click on the ellipsis. In the window that opens, you need to get to about the second you an image file. Click on *the Open* - as a result of .the photo app and zhetsya on the form

Figure: 3 . fourteen

If the form is larger photo graphy , the photo grafi I by stacking th tsya .on the surface of the mosaic

Grafi cal image can be placed on the button, the label and some others in Gia objects . To do this, you are in a similar manner using the properties of their wa **Image** (picture) or *the BackgroundImage* . In Fig. 3 . 14 you see graphics Picture and zheniya on the form on the left of the CCW n ke and on .the label

Property **ImageAlign** aligns graphics on surface of b EKTA just as the .property *TextAlign* (see. 3.4.1) Compensating and a text

The same, but in the code . Photos on objects can also be placed in the code. For example, the operator

```
Cl_sections . Image = Image . FromFile (" D : \ Photos \ Earth . JPG
("
```

will place on the division key a picture whose file address is indicated in quotes. You m of zhete put this statement into the ranking procedure and then by pressing the keypad and shu division it will acquire the image. See 12.3.1 . for details

Transparent " color . Highlight Snap in or label at. Go to the **"** *BackColor* property , there - in the *Web* tab and set the value to *Transparent* (" transparent " color). In Fig. 3 . 15 you see a " transparent " color set for all buttons and labels. If in control has a picture, the picture this, eating e - governmental, not transparent, that we see on the label and the addition of .key results

Figure: 3 . 15

I put the word "transparent" in quotes, because transparency only works against the background of the shape's background. So, for example, other controls are not visible through the "transparent" color. In Fig. 3 . 16 you see stretched "transparent" subtraction button through which the owl p Chennault not see other buttons and text th floor e . P oluchaetsya that rather than to provide true transparency control , the VB simply carries on his background .shape

Figure: 3 . sixteen

To accommodate the schedules in VB is provided as a special element . of control is, Niya *the PictureBox* . We'll talk about it in 12.3.1 . For more information about using images RASSC and Zano in 12.3

Displaying messages - MsgBox .3.7

You can force your computer at any time during the CONCLUSIONS program on dit us any message. Put into some facials at the pair (let it be the : addition procedure) so new to us operas and torus

*("!**MsgBox** ("Hello everyone*

Now every time you press on the adding button, you will see on the :screen is a window messages e Nia

After reading the message, click *OK* . H e-clicking, you can not continue the servant of a thief with a project . A message box is specially made so annoying to prospect for programmers had to bring it to the Watch

yo and set and read what is written there. To do this, you need to write valid .and Tel'nov important things

We'll give the message box the real deal shortly. And now we had only .within a komilis

Main menu .3.8

As th same application Windows without its menu ? ! Our calculator is, like, for anything, but still, it is interesting and useful to make at least a .simple s something

Put breech chu .3.8.1

. Suppose we want to create a menu that looks um so as to Fig. 3 . 17

Figure: 3 . 17

M ethod which is located directly below the application header is called . s INDICATES **g Main Menu** . So , we will create the main m e nu Task menu put this: By clicking on the item " Appearance " , we should see a drop down menu of two items : " Transparent " and " Color fields " . By clicking on the item " Clear " of the drop-down menu, we do cal s kulyator translucent . Hover your mouse on the item " Color fields " , we should see another drop-down menu of two items : " Yellow " and " White " . By clicking on the item " Yellow " , we cr and sim all three text fields in yellow by clicking on the item " White " - in white. Peep to UHC under item " About

.the program " , we call message with a brief description, etc. of a gram
.We have to create the menu first and then make it work

Create a menu .3.8.2

.Copy the Calculator project folder and work on the copy
To create a menu , you need to in the *Toolbox* to find a new element for us **the Main the Menu** (main menu) and place it on the form. But , strange to say, it is a Normal display s in aetsya not on the form, and under it (the icon *MainMenu1* on Fig. 3 . 18). This area, where the dis of false icon, called the **designer component** (*COMPON the ent Designer*) . VB on mesch and is on component designer those element s , which during operation project to .anything be seen s mi
However, at the same time in the same figure we see the "germ" of our future e th menu in the upper left part of the form, where you would normally and posted e tsya main menu, for I lift axle white field with the words of *the .Type Here* , that means " Print here. " That is what we are and we do

Figure: 3 . 18

Click on this field and enter the text "Appearance" in it. Once you're on h HETE enter the text, there will be even more fields of the words *the Type Here* (see. Fig. 3 . 19). They appear where on the opinion of VB you wish CREATE wa be other menu items. Comparing the location of these fields at the specified location Poon to the menu option in the Figure. 3 . 17 , sdela it concluded that in place the right margin should be the text "About", and the ".place of any train him - "Transparent

Figure: 3 . 1 9

Enter the text "Appearance" , click on the right margin and type the text "On the prospect of the gram" . At the same time there will be new field with those in ostomy *the Type Here* (see. Fig. 3 . 20). Looking at Fig. 3 . 17 , we understand that we do not need tons of camping.

Figure: 3 . 20

After entering the text "About" , click again on the field labeled "Appearance" . In this case, lower n it again The appearance and tsya floor ie text *the Type Here* , where we introduce the text "Ave on transparently" (see. Fig. 3 . 21).

Figure: 3 . 21

Looking at Fig. 3 . 17 , we can see that below this field, we have to cc e STI text "Color fields". When we will introduce, the right of the "Color Fields" box appears with seq about you *the Type Here* , in which we, following Fig. 3 . 17 , enter the text "Yellow" (see. Fig. 3 . 22) , and below the text "Yellow" - the text "White" , the stop and Viv ignore other fields .

Figure: 3 . 22

Everything is in order - the appearance of the menu is ready. Run the project. You can see that the intended menu has appeared on the calculator. Click on the points. Submenus drop, as expected, n of more e stestvenno, nothing happens. For this to happen, you need to make the menu work, and . for this you need to write your own procedures for each item

Making the menu work .3.8.3

Each menu item is an object. To verify this, click in the direct mode of ektirovaniya under the item "Clear" and look in the properties window. You see that the menu item has properties, like other of Z OBJECTS. Replace its name (*N ame*) with *Menu_Transparent* . This is to make it easier for them to .use when programming and rooting

Now make a point of "Transparent" is not a single and a double slit l choke mouse. In the code there will preform the procedure to be performed :when selecting a floor on the first menu item

```
Private Sub Menu _ Transparent _ Click ( ByVal sender As System . Object ,
    ByVal e As Sy s tem . EventArgs ) Handles Menu _ Transparent . Click

End Sub
```

Words *Menu_Prozrachnyy.Click* the right side of the header means that we are talking about is clicked (*the Click*) click on the item *Menu_Prozrachny* . It remains to write the code for the Performing ie direct : action

```
Private Sub Menu _ Transparent _ Click ( ByVal sender As System . Obje ct ,
    ByVal e As Sy s tem . EventArgs ) Handles Menu _ Transparent . Click
        Me .Opacity = 0.7
End Sub
```

Here the word *Me* designated form, *the Opacity* - the opacity of its

property of STI (see para. 3.4.3) . W The number 0.7 represents the .magnitude of 70%

.Run the project, check the operation of this menu item

We organize the work of the remaining 3 menu items in the same way.
First, let the first ones them names: *Menu_Zhelty* , *Menu_Bely* , :*Menu_O_programme* . Then program their dis t Wii in the code window

```
Private Sub Menu _ Yellow _ Click ( ByVal s ender As System . Object ,
ByVal e As System . EventArgs ) Handles Menu _ Yellow . Click
Number 1.BackColor = Color.Yellow
Number 2.BackColor = Color.Yellow
Result .BackColor = Color.Yellow
End Sub

Private Sub Menu_ White _Click ( ByVal sender As System.Object, ByVal e
As System.EventArgs ) Handles Menu_ White .Click
Number 1.BackColor = Color.White
Number 2.BackColor = Color.White
Result .BackColor = Color.White
End Sub

Of Private Sub Menu_ About _ program _Click ( ByVal sender of As the
System. The Object, ByVal an e of As Sy s tem.EventArgs ) the Handles
Menu_ About _ program .Click
("MsgBox ("Calculator created in 2003
End Sub
```

In the first two procedures, we are talking about the color of three text fields: *Chi with lo1* , *Number2* , *Result* . In operators, everything is familiar (.(see 1.3.9) . In the last cent e fool you create a message box (see para. 3.7

Entering the next procedure of APUS kai the projects and checks yay .those jobs corresponding th conductive item menu

If you do not want to click on the menu with the mouse, you can select a keyboard shortcut with the same action for each menu item. This combination is called a t Xia **Shortcut** and selected by setting properties of the same name points m e nu in the direct mode of ektirovaniya in the . properties window

Using VB tools, you can create, transform and destroy menus , not only in design mode, but also in run mode. N on on e that I will not stop. Besides the usual menu VB is able to create and context menus (how to do it , .(discussed in 18.9

How we will further improve our .3.9 calculator

- **Provide sensible error message when divided by zero and .indirect and Villeneuve entering numbers into the text fields**
- **Endow it with a password so that all and sundry could not .(enjoy its possibilities of styami (both see. To 7.7**
- **.(Provide the familiar eye view of the results (see . 5.4.7**

.But for this you need to know the variables

"Sound and video . Project "Player .3.10

Since your computer and have a display adapter (video card) and surely there sounds the second adapter (sound card) , you can add to your project view video and listen to star in ka . For example, you can do so by pressing the button or right at the start of the project was distributed to music or on a screen placed on the form, to play video on a clip . To do this, it is necessary that the video file or sound file with music or your voice is already on the .computer disk

You can record your voice into a sound file if you have a microphone. To do so camping is the standard applied and eat Windows «Sound recording" to which get this: In an environment Windows , click **Start ↓ . Programs ↓ Accessories ↓ Activities ↓ Sv in kozapis**

You can perfectly listen and view by using VB fi ly wide variety fo r .mat s

Sound and video in Windows . You need a special program to play audio and video files. P Started in the Windows , you will surely to this enjoy th e diapleerom . Perhaps you simply click on the icon of audio or

video file, then listened and watched, not paying attention, that you lose those files is pr of grams " **Media Player** » (**the Windows Media Player Get**). The media player is standard in the latest Windows versions . If you have never used it, you now useful little work out in it and to use . Open it : **Start ® Pr of gram ® Standard ® Entertainment ® the Windows Media Player Get** . Approximate appearance media player you WMS ie those seen on Fig. 3 . 23 . However, in recent versions of Windows , he readjusted and . can be extremely bizarre and vym

Figure: 3 . 23

Your further actions are intuitively clear . P ri aid **the File ® the Open** (then in s possible, **Bro a wse**) you are looking for and open to the hard drive or a CD or on a network sound of howling or video . W ATEM play it using the usual buttons and controller I tori at the bottom of the media . player : "Play", "Pa in favor", "Stop", "Volume" and so on

If you can not find a single file that will prompt that a few sounds and chalk on diy are on the hard disk of your computer at *the Windows \ Media* **,** and MULTI s to just a very short video clips - at *Program Files \ Microsoft Vi s ual Studio . NET \ Common 7 \ Graphics \ videos*

If you can't hear sound , check your sound card settings in the **Mixer** app . To do this, in Windows, click **Start ↓ Programs ↓ Accessories ↓ Entertainment ↓ Volume Control** . In the resulting window, uncheck (tick) in s Turning the channel and set the maximum sea level of Wen in the channels *the Wave* (for WAV -files and MP 3 -files) and *the MIDI* (for MID -files and RMI -files). If you first want to record an audio file from a

microphone, go to the "Properties", then in the tab "Record", the Charter of vite box switching channel "Microphone" and set a large enough level to and ."Nala "Microphone

Sound and video in VB . Copper and the player can be inserted into your VB about EKT , where you have the opportunity to free them to use, manage, and mood and Vat , and both in design mode and in operation. To use the on vatsya media player in the VB , you need to place in the form of a joint venture e -exponential element control *the Windows Media Player Get* , which is a "copy" of the media player in your project . But it is not included in the standard set on the *Toolbox* . First, it needs to be to claim a place. To do this: **Tools ↓ Customiz e Too l box ↓ COM Components ↓** you find *Windows Media Player* in a long list and check the box **↓ OK** . He appeared l camping in the *Toolbox* . Now it can be placed on the form in the usual way. More forth about process of adding a new ale n is the . management *Toolbox* considered in 4.2.3

Statement of the problem . Let based media player will create . their own n e a great player that looks like this, as in Fig. 3 . 24

Figure: 3 . 24

We see it at the time of the video file playback. The center is located el e ment control *Windows Media Player* as a screen Instr in control cops

immediately underneath. In addition, we see 12 buttons on the form: 6 white buttons on the sides and 6 gray ones at the bottom. On the left - 3 white buttons with the names of sound clips. Right - 3 white buttons with titles and niyami videokl and dressings. I made only 3 each , but there may be, you know, at least a hundred of them. In the future, not to the connected s vatsya with buttons, we will learn how to use the list (18.5), or simply Views .(tonnes regarded disc (20.2

Use our player only. First, by clicking on the white button, select *what* we will play. Then start the playback click the CCW n ke "Start". By clicking on the "Stop" button, we stop. We can enjoy and with b -governmental management tools media player , which immediately below the screen visible. ... But the 4 buttons in the center at the bottom convince us that by means of VB you can control the media player thinner and more conveniently . than in Wi n dows

We are designing a player . Let's create a new project. Let's put *Windows Media Player* in the *Toolbox* and from there on the form. There he will partake p but this view as Fig. 3 . 23 , only without a menu . In the properties window, we see a large number of its properties. No need to change the values of the properties that you are not appreciated and ripped .apart

:Do the following

- **.(Yes yte it a name (eg, *player***
- **Set the property *autoRewind* (winder) in *True The* , to after .in a product player is installed at the beginning of the clip**
- **Set the property *AutoStart* in *False The* , so that the player did not start Playback on dit file just after opening it by clicking on the white button, and waited for pressing the Play button on the . (product of (although you can, and not to do so**

Right-click on the player and then click on the shortcut m e nu .*Properties has* . Here you can also customize some of its properties

Place 2 labels and 12 buttons on the shape. Let's name the buttons .(appropriately (it is best if they look like the text of the buttons

We are writing a program . In order for a specific file to start

playing, you need to perform only two operators. The first shall indicate ple e ru address of the file on the disk to the player knew " *that* play." The second is to directly give ple e ru command to start playback. Here is an example of :a pisi these two statements

Player . ***FileName*** *= "E: \ WINNT \ Media \ Windows Logon "Sound.wav*

() Player. ***Play***

: Here

- ***. The player* is our player object**
- ***FileName* is its property (name, file address). The quotes .indicated by l ny address file**
- ***The Play* - it is a method object *Pl e ep* - the order to begin .playback**

By double clicks on the buttons, go to the code window and fill in the blanks of procedures. Do not forget that after entering the next procedure, the project should be launched and the results of the procedure should be : (checked. Here's what we should get (explanations and the same

```
Private Sub Sound _ Boot _ Windows _ Click ( ByVal sender As System .
_ ( Object , ByVal e As System . EventArgs
Handles Sound _ downloads _Windows.Click
Player .FileName = "D: \ WINNT \ Media \ Windows Logon
"Sound.wav
End Sub

Private Sub Canyon _Click ( ByVal sender As System.Object, ByVal e As
System.EventArgs) Handles Canyon .Click
"Player .FileName = "D: \ WINNT \ Media \ canyon.mid
End Sub

Private Sub In _ cave _ of the mountain _ king _Click ( ByVal sender As
_ (System.Object, ByVal e As System.EventArgs
Handles In the _ cave of the _
mountain _ king . Click
```

```
Player .FileName = "D: \ WINNT \ Media \ In the Hall of the Mountain
                                                    "King.RMI
                                                    End Sub

Private Sub Corkscrew _Click ( ByVal sender As System.Object, ByVal e As
                                                    System.EventArgs) Handles Corkscrew .Click
Player .FileName = "D: \ Program Files \ Microsoft Visual Studio .NET
                                                    "\ Common7 \ Graphics \ videos \ DRILLRLE.AVI
                                                    End Sub

Private Sub Counter _Click ( ByVal sender As System.Object, ByVal e As
                                                    System.EventArgs) Handles Counter .Click
Player .FileName = "D: \ Program Files \ Microsoft Visual Studio .NET
                                                    "\ Common7 \ Graphics \ videos \ COUNT24.AVI
                                                    End Sub

Private Sub Notre_Dame_Click ( ByVal sender As System.Object, ByVal e
                                                    As System.EventArgs ) Handles Notre_Dame.Click
Player .FileName = "D: \ Documents and Settings \ d \ My Documents \
                                                    "Visual Studio Projects \ Miscellaneous \ Notre2.avi
                                                    End Sub

Private Sub Start _Click ( ByVal sender As System.Object, ByVal e As
                                                    System.EventArgs) Handles Start .Click
                                                    () Player . Play
                                                    End Sub

Private Sub Stop _Click ( ByVal sender As System.Object, ByVal e As
                                                    System.EventArgs) Handles Stop .Clic k
                                                    () Player . Stop
                                                    End Sub

Private Sub Play _ Continuously _Click ( ByVal sender As System.Object,
                                                    _ (ByVal e As System.EventArgs
                                                    Handles And Mr. Men _ continuously .Click
                                                    Player . PlayCount = 0
                                                    End Sub
```

```
Of Private Sub Play _1_ times _Click ( ByVal sender of As System.Ob Ject,
ByVal an e of As System.EventArgs ) the Handles And Mr. Men _1_ time
.Click
Player . PlayCount = 1
End Sub

Private Sub Faster _Click ( ByVal sender As System.Object, ByVal e As
System.EventArgs) Handles Faster .Click
Player . Rate = 2
End Sub

Private Sub At _ usual _ pace _Click ( ByVal sender As System.Object,
_ (ByVal e As System.EventArgs
Handles At _ normal _ pace . Click
Player . Rate = 1
End Sub
```

Explanations . First of all, some operators because of their length, I moved on the next e blowing line using a "hyphen", consisting of the space and underscore sign and Bani. According to Dr. detail about this - in 3.12.2

The first 6 treatments correspond to 6 white buttons. You can easily distinguish them by the names of the buttons in the titles of the procedures. They all do the same thing - they prepare a specific file for later playback. Of course, the addresses of the files on to a shem computer will be a completely .different and mi

The meaning of the remaining 6 procedures is clear after the following :explanations

- ***Stop* is a method of the *Player* object - an order to stop playback. The next time you press the "Start" button, playback will resume from the IU with the one on the cat of rum it was .stopped**
- ***Playcount* - it is an object property *Player* (how many times produ g undermines file). If you set it to 0, then the playback will .be repeated inf of the finite**
- ***Rate* is a property of the *Player* object (playback tempo). If**

your tired of twist it , for example, to 2 , playback will be carried out in 2 times faster than normal, then there are people on the screen will be running around like crazy, and p azgovarivat voice th Boer and Tino. And if in 0.5 - will crawl, as boiled , and .snarling

Run the project. Check how all buttons work . Shabby is e in-right .pointer of the current playback position, move the knob gro m bones

Additional notes . If you set the property *Visible* player in *False The* , the player with their management tools will not be , and den on the form, but you will still be able to them to manage , because the shape of the stop and nutsya buttons Creating e you. Thus , you can listen to the sound of the stems clips even when the "absence" of a player on the form. However, when .you try to view a video clip, it is still on I vitsya

Copy the project folder and experiment with it. Here Prospect of stye :methods and properties with which you can experiment

AboutBox	The method shows a window with a .summary of copper and player
Duration	Property - the total duration of the selected . clip in the secular n dah
CurrentPosition	Property - on a second is currently Playback .e denie cells and pa
Volume	The property is loudness. A negative number -10000 ozn and chaet the smallest volume - complete silence. The number 0 (zero) is the loudest. intermediate numbers - .intermediate gro m bone
... *Show*	Several properties, beginning with *Show* , control which of the m e diapleera show on .the screen

Many of the properties of objects, such as *Rate* or *the FileName* , allow the program from a tanavlivat and change their meaning , as we have just been and done (say *Pleer.Rate = 1*) . These include i tsya as *Volume* and *currentPosition* . But there are properties that do not allow this. These are called **read-only properties** (**R ead O nly**). These include and tsya *the*

Duration . What is the use of them then? The benefit of them at least is that we m of us show to see which equals their value , even if we do not ask it . This can be done, for example, by placing in the form of a text box and apply :operas and torus like this

TextBox1.Text = Player.Duration

It means that the value of the *Text* textbox *TextBox1* becomes pa in nym . property value *Duration* object *Player*

A host of other media player properties and methods let you take advantage of its wealth of capabilities , including selecting clips from a list, playing DVDs and audio discs . But to them I ostanavlivatsya s camping will .not

Microsoft Multimedia Control 6.0 . In the Visual Basic 6.0 has its own Media player and Tel audio and video files - *the Microsoft Multimedia the Control 6.0* . E with whether you have installed on your computer, not only the VB , but the Visual Basic 6.0, you can use it in the VB . But the methods and properties are different there. Op and dignity they .(are in my book (see References

:Upgrade your turntable

- **Add a button for slow motion playback. In the future, in the s can instead of buttons p ganiz acce smooth playback tempo .control**
- **Add multiple buttons to set different volumes. In the future, you can instead of buttons to organize a smooth volume control, .comfortably hundred n -standard**
- **Add a large label . When you click on the white button , the label must- s of I in and tsya few words or sentences that describe . your impression of the shaven Mr. Nogo cells and pa**
- **Add a button and a text box to define the duration of the cl and pa. When you click the button , the value for the total . duration of the selected clip should appear in the field**
- **Add a button and two text 's floor I for *determining* the current position of Play e Denia. When you click on the button , in**

one field must appear the number of seconds since the beginning of the play , the other - chi with lo seconds remaining before the end of play . In the future you will be able sdela be so that the .numbers in the floor of the floor s m e nyalis Automatic e lift

- **Add a button and a text box to *specify* the current position of Play e Denia. Clicking the cur th positions I play a must and We establish and vatsya in the value in seconds that you enter in the . text of the numeric field**

Music in the calculator buttons ". Suppose that when you click on the " calculator each of the four buttons arithmetic distributed some koro t cue melodic sound, and for each button a. T and FIR sounds a lot on the disc at *the Windows \ Media* . There they are written in files with the *wav* extension . . Note : Set a novite Propert tons of *AutoStart* in a *True The*

Events and their handling .3.11

Various events . Recall that objects have properties, methods and GSS s Voith. We've already seen several properties and methods. But little was said about the events. In fact, we only know one sort of events - schelch ki we w Coy button and menu item . But the objects of the VB (h t Nosta - the forms and controls) has a large number of other events. So , at the buttons have the following th conductive events: click the mouse button, mouse over the appearance of the button, pressing the mouse button over the button, release the mouse button over the button, change the button color and of Menen size buttons and a few dozen others. In the text box have the following th conductive events: double click on the text box, change the text in the text of the PTO field and dozens of others. The form has the following with a life: movement of the mouse over the form, the size change of the form, the form load (for example, the appearance at the start of the project), closing the mold (for example, by clicking on the cross in the vertex x corner .of it) and a few dozen others at GIH

Event handling . Every time in the previous projects, we Click and whether the mouse button *the Button1* (ie upcoming events - click we w Coy

button *the Button1*), the computer does something that we he was ordered : painting, showing the text, playing m e lodiyu etc. .P. He did it just because we are in the code window in advance written facials at py, which : comprehended with Neely computer that it in this case to do. For example

```
Private Sub Button1_Click ( ByVal sender As System.Object, ByVal e As
System.EventArgs) Handles Bu t ton1.Click
        HScrollBar1.Width = 100
        () RadioButton2.Hide
End Sub
```

Let's look at the title of the procedure. Where it is said that this procedure should be in s fills it when you click the button and it is for *the Button1* ? Answer: in the right-hand side with a minute header after Staples :to

Handles Button1 **.** *Click*

Here **Click** is a mouse click (event), *Button1* is an object with which this event occurs, **Handles** - in Russian "processes". Full of Stu procedure header can be translated as : A procedure that **handles** a click th s sew on the button *the Button1* , or otherwise - specifies what to do with slots h ke mouse . button *the Button1*

:Something similar is written on the left side of the procedure header

Button1_Click

But that's just the name of the procedure. You can change it as you want, .facials, and have pa will still work

On the meaning of what is written in the nutri staples ok , we 'll talk . later

How to get a stub procedure for handling any event? To the floor at the chit procurement procedures to handle clicking on a button or menu item, we are in design mode just did *a double* slit l choke button or : menu item. And in the code window they received a blank

```
Private Sub Button1_Click ( ByVal sender As System.Object, ByVal e As
System.EventArgs) Handles Bu t ton1.Click

End Sub
```

Why did the template appear just for the mouse click and not for some other event? P otomu it so convenient it programmer at , because a mouse click - the most popular event, when it comes to the button or menu item.

How do we get a package to handle some other event? Let us to direct and least .

. (Create a project with two text fields and a button (Fig. 3 . 25

Figure: 3 . 25

We pose this challenge: P ust when a mouse over a button *Button1* the text of the PTO field *TextBox1* text appears " Mouse appeared on the CCW n hell ."

Go to the code window. At the top, you will see two white boxes. Click on the black triangular arrow in the left margin. The cn and juice will open . In it you will see the names of all the controls on a form and as my form (. (Fig. 3 . 26

Figure: 3 . 26

Since we are interested button *the Button1* , choose from the list was her. Now our way is to the right field. Click on the black arrow in the right

margin. A list will open. In it you will see the names of the events associated .(with the button *the Button1* (Fig . 3 . 27

Figure : 3 . 27

Select the **MouseEnter** event , which in this case can be translated as: " Mouse over". In the box, the code singing in the wish to set up the workpiece

```
Private Sub Button1_MouseEnter ( ByVal sender As Object , ByVal e As
System.EventArgs ) Handles Bu t ton1.MouseEnter

End Sub
```

The final fragment header (*the Handles Button1.MouseEnter*) says , that this procedure handles (*the Handles*) the appearance of the mouse (*the . MouseEnter*) over the button *the Button1*

: Let's write the required operator inside the procedure. We get

```
Private Sub Button1_MouseEnter ( ByVal sender As Object , ByVal e As
System.EventArgs ) Handles Bu t ton1.MouseEnter
    " TextBox 1. Text = "The mouse appeared above the button
End Sub
```

Important note on registration programs Foot text books e

: Zag of agile procedures in the blanks that occur in the code window can be very long s E and being transferred to the book list, often do not fit on whether the width with that. At the same time, they contain names of namespaces that VB is fine to throw away. For example , in the proposed of a head

```
Private Sub Button 1_ MouseUp ( ByVal sender As System . Object , ByVal _ e
As Sy s tem . Windows.Forms. MouseEventArgs) Handles Button1.MouseUp
```

you can throw away the namespaces that I have in bold with impunity . P : about radiates much more compact

```
Private Sub Button 1_ MouseUp ( ByVal sender As Object , ByVal e As MouseEventArgs) Handles Button1.MouseUp
```

.In the given s Nation I often and I will act

Examples of event handling

. Below is the program of a number, etc. of the procedure is to process in our project of several forms of popular events and ale n Control comrade. Enter the program into the code window. D To do this, Dr. obeytes the manner described above , in the window to the code The appearance ilis Zagot ki for each event procedures. To handle form events, select if you are working in Visual Studio . NET 2003 , at l e tion field *the Form1 Events Read* , and if in the Visual Studio . NET - *Base Class Events* . Not forgetting the first ones after you enter the next draft procedure is recommended to run and check the results of the procedure. Explanations . are given e us below

```
Private Sub Button 1_ MouseEnter ( ByVal sender As Object , ByVal e As System . EventArgs ) Handles Bu t ton 1. MouseEnter
" TextBox 1. Text = "The mouse appeared above the button
End Sub

Private Sub Button1_MouseDown ( ByVal sender As Object , ByVal e As _ (System.Windows.Forms.MouseEventArgs
Handles Button1.MouseDown
" TextBox1.Text = " pressed key click on the button
End Sub
```

```
Private Sub Button1_MouseUp ( ByVal sender As Object , ByVal e As
MouseEventArgs) Handles Button1.MouseUp
" TextBox1.Text = " Released button mouse over the button
End Sub

Private Sub TextBox2_TextChanged ( ByVal sender As Object , ByVal e As
_ (System.EventArgs
Handles Tex t Box2.TextChanged
" TextBox1.Text = " The text in the lower text box has changed
() Beep
End Sub

Private Sub Form1_Load ( ByVal sender As Object , ByVal e As
System.EventArgs) Handles MyBase .Load
MsgBox ("The Form 1_ Load procedure is being executed . The form is
(". not visible yet
" TextBox 1. Text = " Form loaded
End Sub

Private Sub Form1_Closed ( ByVal sender As Object , ByVal e As
System.EventArgs) Handles MyBase .Closed
(" MsgBox (" Form is closing
End Sub

Private Sub Form1_Resize ( ByVal sender As Object , ByVal e As
System.EventArgs) Handles MyBase .Resize
"TextBox1.Text = "Form resized
End Sub

Private Sub Form1_MouseMove ( ByVal sender As Object , ByVal e As
_ (System.Windows.Forms.MouseEventArgs
Handles MyBase .MouseMove
" TextBox1.Text = " Mouse moved over the form
End Sub
```

. Explanations

- **TextChanged event . Text changes are tracked in the *TextBox2* , rather than *the TextBox1* , that we only need to issue**

messages e Nij. Don't be confused. Checking if the event is simple - try the first ones to change the text in the lower field - immediately appears and you will hear the sound in the upper field. By the way, you will hear it when you start the project, because when you start the project, VB writes their text to the fields. But why, then, we do not see when you start in the upper field of the message "The text in the lower ones to Stow the Edit l Xia" , is a mystery. Which a little .later try to guess for yourself

- **Form *Form1* in the right side of the header and the . handler indicated MyBase**
- **Event Load form - download form - occurs when you run proe to that. This is a very important and widely used event. While the procedure *Form1_Load* in s is satisfied, forms in the screen you it still can not see. Therefore, all the actions that you programmed in this procedure are performed before the appearance of the form on the e to the wound. However, as the form itself, and controls to this point already invisibly etc. , and present to a project and they can be run. Some events had to occur and be arr and nerds to the procedures s *the Form1_Load* , as evidenced by the sound before the appearance of messages "is carried out of the Form 1_ the Load . . " The form has not ve d on**
- **Event Closed form - closed form - occurs when you are of a kryvaete fo r th, for example, a cross in the upper right corner. Once the form is closed, it disappeared from view along with its text boxes . Therefore th message of this fact I have not entrusted the . text field, and on to the well, *the MsgBox***
- **When you drive the mouse over the form, the event .*MouseMove* there many times with e Kunda**

Event List object with the notes you can find in the syst e aid IU (see para. 4.4.9), placing the text cursor on the name of the object and pressing *the F1* . In the window n on the power, you will find events Title (*Events*

.*Read*) and lightning icons

We will get acquainted with the rest of the popular events as we study .the book

Various necessary things .3.12

The first three subsections concern the design of the program in the .code window. Computer they are not well train us, and you need them

Comments .3.12.1

Comments - this clarification in the text of e program. Why are they ?well Well us

When a person looks at the part of your program, such as this (from cal (s kulyatora

```
Private Sub Cl _ subtraction _Click ( ByVal sender As System.Object, ByVal
e As Sy s tem.EventArgs ) Handles Cl _ subtraction .Click
(Result .Text = Val ( Number 1.Text) - Val ( Number 2.Text
End Sub
```

he may not understand what is the meaning of, and for what purpose the program napis and on. Why, for example, *Val* ? And everything else ... Well, okay, if not first met it , it's still not so bad, and the trouble will come then , . when you are after some time forget what is what

Any professional programmer knows two things. Prim th - any old program after 12 months is forgotten completely. Deut th - any old program h e res 13 months is suddenly desperately needed (for example, as a raw material for the n of the howling of the program) and therefore in urgent need to understand. Taught by bitter experience, the programmer provides many lines of code with his own comments. Of prev in boiling n three lines :on the beam this

```
' Procedure for handling pressing the subtract button:
Private Sub Cl _ subtraction _ Click ( ByVal sender As System . Object ,
ByVal e As Sy s tem . EventArgs ) Handles Cl _ subtraction . Click
Rezultat.Text = Val (Chislo1.Text) - Val (Chislo2.Text)     ' Val function is
"needed for "turned e Nia text in a number of
End Sub
```

The computer does not need a comment, it will not understand it, and if it starts to understand, then nothing good will come of it. So the computer should not look into it . And Thu about would clearly show the computer from where it is not necessary to look into the programmer in each page of Ke code before the comment raises a single quote. The VB , compiling the next line of code, it scans from left to right, and as soon stumble on to a quotation marks, to the right is not looking. Therefore, etc. and BEE .comments write the code does not make sense

.VS highlights the comments in green for our convenience

Transfer of Long on track th conductive line .3.12.2

Sometimes the operator is obtained so long that does not fit in w and Rina code window, and to see it as a whole, have to scroll the Code window to the left or right. This n e nice, because the scroll does not want to , and I want to see the whole of the operator on the screen entirely . Part of the Emperor of the well can be transferred to another line combination of space, and underscores. For example and measures, instead of

```
( Result. Text = Val (Number1. Text ) + Val (Number2. Text
```

can be written

```
_ ( Result. Text = Val (Number1. Text
( Val (Number2. Text +
```

or

```
_ = Result. Text
_  (Val ( Number 1.Text
(Val ( Number 2.Text +
```

As you can see, there are a couple of characters at the end of the lines as .a hyphen : a space followed by an underscore

Personally, I use one more tool for the visibility of the program - for the code window I choose the *Arial Narrow* font . It helps a lot, since the font is narrow and many letters fit the width of the window. How to do it, napis well . but 4.4.7

VS offers another tool: **Edit ↓ Advanced ↓ Word Wrap** . In this case, the code wraps to another line automatically. But I do not like this tool because it violates the usual eye etc. of programmers and padding system on .the left edge of the window

Writing multiple statements on one line .3.12.3

VB allows writing on a single line several operators, separated by colons , I mi . For example , instead of

Button 1. Width = 200
Button 2. Width = 100
Button 3. Width = 140

:can be written like this

```
Button 1 .Width = 2 00 : Button2.Width = 100  : Button 3 .Width = 1 4 0
```

.This nicely saves vertical screen space

We run our and programs without VS .3.12.4

In Windows, you are used to launching games and other programs in two ways: by clicking on the program icon on the Windows desktop (or in a folder or in Explorer), or by selecting a program from the start menu on the taskbar. We have created in VB application of a start-up th tsya so far only from the environment the VS , that perhaps offends your pride. Let's try to run and x is independent of s e program s , that is, without ZAPU with ka the . VS

Shut down the VS . While in Windows , go (using Explorer or from the *My Computer* icon) to your project folder, for example *Calculator* . There you will see , and those few files and folders on your computer. Go to the *BIN* folder . There you are and the file *Calculator.exe* . This is the program of your calculator, which m about Jette run without launching the VS . Start it by clicking (or double-clicking). She works as a. You can copy it wherever . you go , or create a shortcut for it anywhere in to m pewter

How did the *calculator.exe* launch file get into the *BIN* folder ? When we started our project in the VS , the project is automatically compiled and the compiled file *Kalkulyator.exe* of a pisyv and I into this folder. You can . achieve the same and without running: **the Build ® the Build Solution**

However, if you try to copy your *Kalkulyator.exe* on another computer w ter and there to run it, then you can expect a failure. There may need more .(sophisticated de second consequence, which I have described (see. 4.6

Chapter 4. Work in the environment Visual Studio .NET

Starting with the next chapter, we will take programming seriously. But going into battle, you need to get used to the armor. Will not we spot s katsya in our armor? Our armor is Visual Studio . NET . We can not afford to puzzle out of the windows, toolbars, menus, and buttons media VS has become for us ka m him stumbling. And vice versa - the knowledge of VS will give our programming req on sary Udo b GUSTs and power. Therefore, now we will study exactly how to work in the VS environment , that is, we will not .program, but will learn to press buttons

Not all, h t on here described, is useful to you immediately, but even if you are in a hurry, carefully about the look are all the material chapters to , when there req about gence, to know that the solution occurred means the problem is, and you about him where read something . In any case, at some point in the future, stumbled again, you br on the sieve everything and read .this chapter systematically

If you decide to study the book , starting with this chapter, know that the very basics of working in the VS environment are already outlined in 1.3 and further on with examples of creating projects. The hour t NOSTA in .1.3.3 briefly described as working with *Toolbox* in 3.2.1 - a box m properties

In addition, some of the material relating to the environment, the VS , it seemed appropriate to present later, in those parts of the book , where they originated n a neo b divergence . Here are the locations : Chapter 9, " Debugging Program », 11.5.9 " the IntelliSense », 21.3 " Structure of the project and solutions . Solution Explorer window . ", 24.4 " Working in VS ." with databases without a project using Server Explorer

. Everything else is outlined here

In the same chapter, I briefly outline the installation process and Visual Studio . NET on your computer and how to install your finished project on other computers. If you are not going to do either one or the other, it is still closely proglyadite this material, as it contains the necessary information . about the VS

Installing Visual Studio . NET .4.1

To work with the VB , on your computer is, to be installed as well (is . installed on the van and) the Visual Studio . NET

If you do not know whether the installed VS on your computer e SOCA about Buyten run it from the Start menu, where it should be in a program called *Microsoft Visual Studio .NET* or *Microsoft Visual Studio .NET 2003* . If the start of the menu vom not, then almost certainly it is not installed. There is another way to check - look in the Windows Control Panel , in the *Add or Remove Programs* item . The best way is to get to the startup file of the program. That's it and e res : *Program Files \ the Microsoft the Visual St u dio . NET 2003 \ Common 7 \ IDE \ devenv . exe* . Double click on it - if VS .starts up, everything is fine

If VS is not installed, you need to install it. If you've nothing We establish and wali, at the beginning of 4.6 , you can read, h is a software installation on Compute e re at all. In truth , installing VS is not for a beginner. But if there is no one around who could help you ... ? In general, .boldly forward. Vero t Nosta that you get enough to eat ka

VS makes quite high demands on the computer and the operating system is, IU. These requirements change rapidly from year to year. In 2004 Treb on was considered as a minimum Pentium II - 450 MHz with 192 Mb of memory, and the recommended Pentium III - 600 MHz with 256 Mb of memory. As for the operating system, then n olnuyu Programmable functionality of Bani could only provide Wi n dows 2000, the Windows XP . Professional and the Windows Server 2003

To install the VS , you need an installation package CD-ROM s . Installation is slightly different on different computers and for different . versions of Windows

: As of consistency in ashi 's action first installation the VS

- **Insert disc # 1 from the package into your CD - ROM drive . To do this, first district and to press the drive button and wait until it pops out from the disc tray. And to kuratno , without distortions, place the disc on the tray shiny nenadpisannoy down. Art and Wright did not touch the shiny surface with your fingers. Press the drive button again. The tray will slide into the drive,**

.taking the disc with it

- **Further , one of two things is possible. If you automatically start the wizard setup, then re jump iterated through two points (to the " start work installation wizard ") . If nothing happens . after a few minutes, then read ff e blowing point**
- **You need to get to the *Setup.exe* file on the CD. To do this, go to the receptacle and choke *My Computer* on the desktop, the Windows (or in the start menu) and look in the window icon drive the CD - ROM drive . Right-click on it and select " Open " in the context menu that appears . Before you OPEN on window etsya .with a list of folders and files to m pact drive and**
- **Find in this window the file *Setup.exe* and double click run .it on the executed e set**
- **The installation wizard starts. He leads the entire installation . In ashi steps of a given master. Your job - to answer his questions and choose from the proposed variation n comrade. From time to time you have to s full yat request masters insert another disk drive. Sometimes the computer will ask you to reboot, sometimes - as per a sitsya on licensing conditions (to indicate that you agree to these terms , select the point *I agree* or *I of the accept*) . Now I look at the key in order m on cops mouth .and Novki**
- **Before you install on your computer program itself VS master mouth and Novki must SET it a number of pro gram, the approximate list of which you can see in Fig. 4 . 3 and without which VS can not work . Since on most computers these x programs there , the installation package VS they prudently soda p zhatsya. Therefore, after some time you will see the setup window (Fig. 4 . 1) with vyd e lennym step 1. This means that the computer has detected the absence of some necessary programs .and offers their mouths on the Add button**

Figure: 4 . 1

- **Click step 1. After a while, you will see (Fig. 4 . 2) the message of and interested in those who are going to develop and make available on its computer Web -applications (see. Chapter 23 .), and use your computer as a Web - server**

Figure: 4 . 2

- **If you are not going, then click *Skip* and go to the next item .**

If the GSS and raetes, click *the Setup Instructions* . The guidance will you read, you'll have to immediately supplement the installation of the operating system the Windows , etc. about addin g in and setting the so-called *Internet Information Services* and *the FrontPage Server the Extensions* . It also details how to do this. Keep this instruction manual, otherwise you won't get to it later. Do what it says. To do this, you m about Jette will need the system drive the Windows . After the instruction executed in s revolve to have a hold is reached VS . Reinsert 1 disc and click on step 1 of . the installation

- **Now the computer shows you (Fig. 4 . 3) , which is software : for it lacks the beginning, with SETTING the VS**

Figure: 4 . 3

- **Chief among these programs are *Internet Explorer 6* and the *.NET Framework* . U of Latter is not even the program, and , as I said, the whole platform on to a Torah based work is not only the VS , but also other programs. It is so important that Microsoft is going to include it in the next version of Windows . Push the first ones *Install Now* and follow all that tells the wizard until you have set all needs and mye programs s . Finally you will see the same picture as in Fig. 4 . 3 , but against each program bud e t put a check mark and a sign that the program and set and . Press and hold *Done* . Installation step 1 is complete.**
- **Soon the master will show you the picture (Fig. 4 . 4) in a sign that you can proceed to step 2 of installation, that is already n . e mediocre to install the VS**

Figure: 4 . 4

- **Click step 2.**

- **After the conclusion of the license agreement you will be given perturbation Well Nosta select components the VS , you want to install (Fig. 4 . 5 .(**

Figure: 4 . five

- **Check the box next to *Visual Basic .NET* . Remove the check boxes in *the Visual the C ++ .NET* , from the *Visual C #* and *the Visual J #* . If you want to use the full capabilities of the VS , SET and those all the other check boxes as shown in the figure. If you want a bit of a ehkon about mit s disk space, so to leave them as the wizard . If you really don't have enough space , you can even take them off . In this case, you will have to come to terms with the fact that some features of VS will not be available. Table top right dormancy and binds you to the volume of the selected component , in the middle of the field - its meaning , and diagrams m ma Lower**

.Right - Free Di with ke

- **Click *Install Now* . Computer long enough copies files from the installation n GOVERNMENTAL disc to your hard drive , and finally gives the image with the message that the Step 2 of the . (mouth and Novki and completed (Fig. 4 . 6**

Figure: 4 . 6

- **. Click *Done***
- **The computer shows the same picture as in Fig. 4 . 4 , but with a highlighted 3 step. You are invited to establish a reference library of the MSDN Library then , which is expected . simultaneous e Menno is the basis of the Help**
- **Click step 3. MSDN Library will be installed on your computer . In the process of the mouth and Novki you will see a row of windows. You can see the names of these windows in the**

order of their appearance and the pointer to the current window in Fig. 4 . 7 in the left column. Including you will see the license agreement (select the item *I agree* or *I of the accept*) and windows, wo b -rotating you certain information (press *the Next* , which .("means "Gave s Chez

In one window, you will need to select the type of mouth and Novki (.(Fig. 4 . 7

Figure: 4 . 7

:Choose one of three

.(*Full* - full installation (requires 1.8 Gigabytes of disk space

Minimum is the minimum setting. Requires 620 megabytes of disk space. For early and huge capacity that is enough. If necessary, you can .always get more info p mation with co m pact disc

The Custom - Custom installation - for experienced programmers who . know how mustache t roena the MSDN Library then

In the *Ready to Install* , press *the Install* - master begin prolonged to about feasting files on your computer , after which the window *Finish* will . complete the installation of the MSDN of L i brary

- **After installing MSDN Library, Step 4 offers to upgrade VS via the Internet or from a dedicated disk. E with Do not want to . press *the Exit***
- **VS installation is complete. After the installation is complete, you can remove the CD and launch Visual Studio . NET .**

Windows environment Visual Studio .4.2 .NET

Until now, I have not systematically looked at VS windows , toolbars and menus . And you have probably already encountered in this regard, with some prospect of Bloem. For further progress is necessary to consider them used to Leah detail. Let's start with the windows. They are among the VS pretty much dazzled and a feeling stack with e on where you want to find a needle Therefore, in terms of different windows, I will focus on of the same, but first we have to learn to cope with them , regardless of their meaning : .open, close, , move, Mr. and tunable

Moving and configuring windows .4.2.1

Open a project . Let it be called “Hierarchs I ” . P check on Fig. 4 . 8 . It does not matter if the picture on your screen is not very similar to the district .and sunok

Figure: 4 . 8

Window groups . How many windows do we see in the picture? . Let's try to count and count

First of all, there is a large white window on which the form is located. This is the form designer window . We see two bookmarks above the window. Let's click on of a clutch of *Form1.vb* , and found that the code window appeared on the site of the designer window. Let's click on of a clutch of *the Form1.vb [Design]* and found that in place of the window code again there was a window designer. Thus, the two windows are combined in **a group** of windows with tabs for easy switching th cheniya between . windows

On the right we see another two of these groups, only the size of their small and of a clutch arranged not *over* , and *under the* windows. The upper band consists of two windows (of a masonry *Solution Explorer* and *Class View*) . The lower one also *consists* of two windows (the *Properties* and *D y .namic Help tabs*). Clicked on and thumbnail Dr. Kam

Below we see an oblong *Output* window . No bookmarks. But this is .also the group, Tol s to one of the windows , etc. otomu and without tabs Place your mouse over the *Toolbox* icon on the left side of the VS window . Our usual gray "toolbox" *Toolbox will* appear on the screen . This is .also a window, despite the fact that others have Gia window does not Notice the *Server Explorer* icon next to the *Toolbox* icon . P ohm with Titus on his mouse. About to but the *Server Explorer* will pop up on the .screen

In fact, there is much more about con in VS than we see in the figure, . only they are not open

We close and open the windows . To get rid of the feeling of tightness in the e to the wound, let's start with the fact that close all the windows , IC key (not to spare at tatsya) *Toolbox* and *Server Explorer* . Each window is in the upper right corner of the icon cr e stick « ○ ». Let's click on the crosses as long as the main media window does not OPU with teet (.(Fig. 4 . 9

Figure: 4 . nine

Now we can move in our empty apartment. Do not think that our project is closed, it is open, only not visible , the windows are closed , not the project .

The commands for showing many VS windows are located in the **View** item of the VS main menu . You need to start by opening the *Solution Explorer* window . Our actions : **View ↓ Solution E x plorer** . In Fig. 4 . 10 you see the environment VS after this action , as well as open the menu *View Take* , which is open submanifold e nu *Other the Windows* («Other . ("windows

Figure: 4 . ten

Click on the *Form1* icon in the *Solution Explorer* window . Now choose m e nu *View* items *Code* (the Code window to appear) , *Designer* (so that the window appeared Form Designer) and *Properties has the Window* (to appear and elk properties window) , and in the submenu *Other the Windows* - window of *the Output* . In your environment, it looks like the one in Fig. 4 . 11 . It differs from Fig. 4 . 8 with fewer bookmarks (as the number of .(windows has decreased

Figure: 4 . eleven

Moving the windows . Now let's learn how to move windows. To bring the exp e ments to the reality in the Select menu *View* window *Class View* , and in the menu *the Help* - box *D y NAMIC the Help* . Wednesday gained the previous form (I - as in Fig. 4 . 8 , and you may have another) . H adeyus that the right hand side you can find at least one group of windows and thumbnail Dr. Kami. If it is not, then read on, you'll soon understand how such a group sex in the chit.

A group of windows can be dragged around the screen by grabbing the title with the mouse . Drop e of - a central band with a window E *Code* and *Designer* . In Fig. 4 . 12 I grabbed his arm for h and heads one of the groups (tab *Solution Explorer* and *Class View Take*) and re tasch silt it in the center of the screen , without releasing the mouse button . Visible outline of the situation of the window , when I let go of the CCW n ku mouse.

Figure: 4 . 12

.In Fig. 4 . On 13 I released the mouse button

Figure: 4 . 13

You can see that a group of windows (tab *Solution Explorer* and *Class View Take*) was in a e Redin screen, and a group of windows (tabs *Properties* and *the Dynamic the Help*) smartly seized m th century in ECHA lice group. This is - a distinctive feature of the VS . When you run the project, use and SRI care system and in the many other cases, some of the windows themselves emerge, disappear and are interchanging This shustrost movement, appearance and disappearance of the windows at first leads novice confused, and then you get used to and even beg and naeshsya feel comfortable with this behavior. Behavior is intended to ensure that at the right moment on the screen were just the right window, and the rest should be

.spryat and us

Start a risky business, namely, repeat after me all manipulations with the window and E, which I've done. Successful manipulation requires experience, and if you don't have it, then after a while your environment will look like a heap of garbage . However, do not despair and White, persistence will voznagr and REPRESENTATIONS, and in a few paragraphs, you camp .ete calm and confident master of his cf. e dy

First of all, repeat after me the movement of the group just described . Goth o woo ? And now once again grab the title of the same group, and a wide variety of mod and Zom we walked it across the screen , without releasing the mouse button . At the same time approaching yte her close to the edge of the main window the VS , wire ite it differently over other groups and windows, with a sandwiching mouse with Zago agile, bookmarks and boundaries of other groups and windows. You will see that depending on the current position of the header with respect to n th rechislennyh objects contour windows abruptly changes shape and position, showing those n Dentsu " stick " to the cr and the main window wells VS and other groups, and the windows , "push", shifting the group and windows vertically or horizontally. If you align the mouse cursor with a title or other tabs GRU n nN, the path will take such a form as to Fig. 4 . 14 . You can see that here the contour of the whole group took *the Output* , and from there to the circuit 's stupas at the bottom (the future tab) . This means that when you release the mouse, your group is attached so Xia this. The result at this point is formed .from a single group 3 with three windows bookmark and mi

Figure: 4 . fourteen

.Release the mouse

Similarly, we can move most of the groups and windows around the !screen. Feel free to move with all his might, let n and set foot chaos

Setting up windows . Now let's see how to restore order. Zakro first ones all the windows , including the *Toolbox* and *Server Explorer* . About tkroyte window *Solution Explorer* and moor nd those using the right edge of the window the VS , how it is moored at the Fig. 4 . 10 . Right cells and vishey click on the window title *Solution Explorer* . You will see the context .(menu for the SEL about Dr. condition I Niya window (see. Fig. 4 . 15

Figure: 4 . 15

.Let's talk about the items on this menu

In the picture you can see that the menu is selected on the condition **Dockable** («moored nd u s Xia"). This means that the window (or a group) when you move the screen will seek to tie up, to stick wherever you got, as I have already described the little wounds s Shae . Try it. Etc. and Lipsheim . window called **tied**

If you have selected a state **the Floating** , that window will cease to sum and came down, and the behavior will move through the environment familiar and calm, not mutual for acting with other windows, the behavior of . the normal windows the Windows . Try it. Such a window is called **floating**

When the window is docked to any of the window edges of the VS , the menu pojavl I etsya click **the Auto Hide Event** («avtopryatki") . If you select this status , the window entirely , I each time when you are not using it, will float away from the window edge the VS , on a taviv at the edge of his badge with the inscription as it makes *Toolbox* . When the window again you need , you just become and those mouse on the receptacle and chok - swim out the window and a. This is very saving on screen space. Try it. Please note, until you take off releasing the window is now impossible to steal from . the edge e of the point *the Auto Hide Event*

The window group behaves the same way. Each window from the group .has its own icon at the edge

Item **Auto Hide** is so popular that there spetsial title for him a window s ny icon, which you can quickly flip we w Coy , installing and removing from a distance *Auto Hide* . When the icon looks like , the state of *Auto Hide* is set . but when - not in a tanovleno

:Here are some useful windowing techniques

- **.To "pull" a window out of the group, drag it by the tab**
- **To attach the box to the group, drag ite its title in the . group header or e Well of bookmarks**
- **To prospect and connect the box to the central group (which includes the code window and the Form Designer) Uber and those at the window selection as a point *Dockable* , and point *the . Floating***
- **In order to "pull" is attached window of the central group,**

isolated , and those at the window click *Dockable* or click *the*
. *Floating*

- **The code window and designer window do not move. And they have a different context menu. To see the expected simultaneous e Menno both window, select from the context menu *the New the Horizontal Tab Group is* or *the New Vertic al Tab Group is* . To go back to the previous pos e of, the context menu of *the Move . to the Next Tab Group is* or *the Move to the Previous Tab Group is***

The **Hide** item simply hides the window. Open it again - via the *View* . menu

When a window or group adjacent to one another (like the window *Solution Explorer* and *Properties* on Fig. 4 . 11), you can drag your mouse in the usual way to shift their border . To do this, slowly moving the mouse across the border to catch the moment when the mouse pomenyatsya I is a .kind of a double arrow , and then so on and shield mouse

A minimum of windows for work . What are the minimum :windows that you can get by with to get started ? Here is cn and juice

- ***Toolbox***
- ***Solution Explorer***
- ***Properties Window***
- ***Designer***
- ***Code***
- ***Output***

Now that you know all this, close all windows one by one and exposing the first ones desired. W ATEM tow them to the desired location , moored to the vending edge, choose a comfortable condition and connect to the .appropriate group. I wish you from the infantry

Solution Explorer .4.2.2

. Now let's talk about the meaning of some VS windows

The window *Solution Explorer* (it is evident, for example , to Fig. 1 . 5) we can see the structure of our e of the project. So far from the window little use , since our project structure, etc. of the flock - one form and associated

code. Complex projects can be with a stand of many forms, and other component s , the relationship which is just and convenient to see this on to .no

You can simultaneously work on multiple projects, Combine and nennymi in the so-called *solution* (*Solution*). You will learn later about why this is necessary . All solution projects are visible in the same *Solution Explorer* window . Until now, we have worked with one and night projects, but it was still believed that we were working not just with a project, but with a solution consisting of one project. We will think so in the future. So often I . do not bud in Dr. h Leach project and the solution

B of Leah details of the decision and the *Solution Explorer* is written in . 21.3

Toolbox .4.2.3

Take a look at Fig. 1 . 5 . The *Toolbox* window contains *Data* , *Components* , *Windows Forms* tabs , and others at the bottom of the window. The controls are located in the tabs. Most of them are located in the *Windows Forms* tab . Bookmarks are not very similar to regular bookmarks and behave in an unusual way : they travel up and down. But nothing, and you can throw .it out

Usually, all the controls in the tab of the *Windows Forms* do not fit in on the s cell window *Toolbox* . To get to those who are not located, we twist the window with black triangular arrows in the upper and lower parts of the .window

You can drag the controls up and down the *Toolbox* window in a way that suits you. Right-click on the window surface and in at I curled context menu, select *the Sort the Items Alphabetical l y* . The controls are arranged alphabetically. Do not select the *Cut* and *the Delete* , not the *Toolbox* lose ale .n Control comrade

If the *Toolbox* on your computer is too narrow and if a wide , you can .Peret and schit we w Coy its border

The standard set of controls in a window *Toolbox* lacks many n of Leznov and nice controls. To put on the *Toolbox* desired e e control cop, on the context menu *Toolbox* click *the Add / the Remove the Items* . Singing in .(the wish to set up window of *the Customize Toolbox* (see. Fig. 4 . 16

Figure: 4 . sixteen

The tab .*NET Framework Components* (new components) or *the COM Components* (old components) you in the long put a tick list there, where it is not yet available e on , in front of the desired item . Then - *OK* . Corresponding control app and is called in the window *Toolbox* . Now it can .be placed on the form Oba hours manner

Object Browser .4.2.4

The Object Browser - to review the tools and search facilities and amounted to -governing their properties, methods, and so forth ochih component , as well as the namespace in which they dis about lozheny. That is, to browse and find the entire wealth of the .NET Framework class library . And not only her. Thus, the *Object Browser* is like a reference system from .the topic

Let's click **View ↓ O ther Windows ↓ Object Browser** . Before you open the t smiling window *the Object Browser* , which is in the left panel .(*Objects* you will see n ie how many rows (Fig. 4.17

Figure: 4 . 17

Assembly and . Each line here is the so-called **assembly of the second** . An assembly is a file (sometimes several files) that stores one or more namespaces. Shown here are the assembly, the space where the names .in the district and worth the time at the disposal of your project

Please note that among other assemblies, the assembly of your project is .also indicated. We'll talk about it later , but now you can ignore it

The rest of the assembly, to about torye you see in the picture, provide a repository namespace a library, e ki classes .NET Framework . H e entire class library .NET Framework is currently connected to your project, but only its most widely used e mye part. This was done to save money. Accordingly, in the figure you can see not all assemblies of the .NET Framework class library . But you can if you want to rapidly d Enable complemented and tion .(assembly (see para. 25.2.2

The assembly file usually has the extension *dll* or *exe* . By appointment assembly and a near and to the well-known libraries, *dll* , but it's not the same .thing

The same assembly may find tsya more namespaces and vice versa - one namespace can be located in several assemblies. Often the assembly . contains a namespace with the same name called her

N Example n uteshestvi I for Browser the Object . A avayte d Aubert in *Object Browser* to method *Write* stem with ca *Debug* . Click on the plus in the assembly line *System* . We see (Fig. 4 . 18), that the inside of the assembly *System* includes several namespaces . Each of them is marked with

. **{}** a pair of curly ck on the side

Figure: 4 . 18

The peculiarity of displaying in this window is that all namespaces in the assembly are immediately visible , including spaces nested in other spaces. So, you see, and the namespace *the System* , and its member space consisting of n names GUSTs *System. ComponentModel* , and namespaces . included in *Sy s tem. ComponentModel*

Click on the positive side in the namespace *System.Diagnostics* , which gives us different Oring s tools, including and debugging etc. of grams. We see (Fig. 4 . 19) within the space of a large number of names and honors

.objects

Figure: 4 . nineteen

Click the class of *the Debug* (for him personally and not on the positive side) to vyd e pour it. We see (Fig. 4.20) in the right pane *Object Browser* assembles n you class *the Debug* : properties and methods. At the bottom of the window you can see a brief description in the s divided element. Over .time, we will learn to understand these descriptions

Figure: 4 . 20

Note that not all of the components of this class are shown in the right pane. Many inherited "inherited" from the parent class (see para. 22.7) of the

. Criminal Code does not and are connected

Another example : A Aubert to the color *Yellow* object *the Color* .
Click on min u sikam to collapse all lines in the left pane of the *Object Browser* so that it returns to its original form. Then click on the positive side, go to the assembly of *the System.Drawing* , there go to the namespace *the System.Drawing* , it locate the object (structure) *the Color* and s divide it click. In the right pane, you will see many properties of this structure, whose . names are the names of the colors. Cf. e di them you will see and *Yellow*

As I said before, some namespaces can appear in parts at once in several assemblies . So, we see parts of the *Microsoft.VisualBasic* namespace both in the *Microsoft Visual Basic* assembly (in other *words, Microsoft Visual Basic.NET Runtime*) and in the *System* assembly , and we see parts of the . *System* namespace in the *System* assembly in the *mscorlib* assembly

Remark . Please note that the *Object Browser* has many properties, the cat of ryh not in the Properties window, as shown here, and the properties that make sense only in the mode of [*the run*]. For example, a property *SelectionLength* object of *the TextBox* , which s gives the length of the .fragment of a text which is emphasized in our text about the PTO field

Search . Very useful ability to search the *Object Browser* we need

assembles n t s . The search button is used for this . Let you forget where tries to ditsya *Beep* . You click on the search button, then the pop-up search window (Fig . 4.21) ff dial on in *Beep* , click *Find* , and in the resulting :window, you can see below p e result

Mi crosoft.VisualBasic.Interaction

Figure : 4 . 21

start page .4.3

When you first launch the VS , the first thing you see is the front page (Fig. 1 . 1 or Fig. 1 . 2). Open it on the tab *the My Profile User* («User») . If you have a well Kie some preferences and programming experience , here you can set up your profile on Wed e de the VS , if not - leave everything as it is. Only recommend to install a filter *Help Filter* on *the Visual Basic* , to assistance for the system is not used ivala screen nonzero Well Noi you information on other lang s stone. If you have previously programmed in Visual Basic 6.0 , then you can select the appropriate profile here, then VS .will provide you with the usual settings

The next time you start, the start page will open to the *Project s* tab (in Visual Studio . NET 2003) or the *Get Started* tab (in Visual Studio . NET) . This tab offers you to choose to open one of the last projects with which you .are a slave of the hoist, or create a new project, which is very Udo used yet

.The rest of the bookmarks are related to the Internet

If you get tired of that every time you run VS appears the home page for the first wee tab *My Profile* in the list of *the At the Startup* (Fig. 4 . 22) and :select one of the variation n comrade. Here they are in order

- **.Show start page**
- **.Open the solution (project) that you worked with last**
- **.Show project opening window**
- **.Show the window for creating a new project**
- **.Show nothing**

Figure: 4 . 22

... . If you miss the start page, then ***Help ↓ Show Start Page***

Visual Studio .NET Main Menu .4.4

And now briefly consider all the necessary n s at the beginning of the main menu items cf. e dy the VS . We have already talked about many of them or will talk later, and therefore they are only mentioned here .

Note that the m ethod VS and its individual items significantly me u t its content depending on what we are doing now. So, point *Format* is not visible (as unnecessary), when we work and eat in the code window.

File .4.4.1

New ↓ Project . Removes from among all open projects and creates n of vy Prospect of the CPC.

Open ↓ Project . Removes from among all open projects and open for one of the projects you saved district and her.

Add Project ↓ New Project . If you are in an environment already open s project s , the Doba in the wish to set up for them a new blank project .

Add Project ↓ Existing Project . If you are in an environment already open project, the Doba in the wish to set up for them one of the projects that preserve nennyh you earlier.

Close Solution . Removes from the environment solution that is all proe to t s .

Save All . Saves the solution, that is, the entire contents of all open prospect of CPC s .

.Print . Prints the program or other project information to a printer .

. P age Setup . Setting up printing .

Recent Project s . A small list of projects with which you have been working recently . Clicking on a project opens it. It bys so yardarm than *the . Open ® by Project*

. Exit . Yield of VS .

Edit .4.4.2

Undo . Undo recent actions. Works not only in the code window, but also in the design window.

Redo . Revert undone actions. Works not only in the code window, but also in the design window.

Cut , Copy , Paste , Delete . Oba h nye and familiar for you to move commands, copy, delete words and fra g ments of your program text. It

applies not only to the text in the code window , but also to the e-mail e cops control on the form in the designer window . How these actions are carried out with the text is described in Appendix 2 . And with the elements of control e Nia's how : Often, when we need to have in the form of several absolutely identical on during configured controls, it is more convenient not to take them one by one in the *Toolbox* and adjust each po0otdelnosti and blur with TIV one on form and carefully configuring it necessary way, :then copy the n th number of times . Copy two ways and E

- **Click on the copied object ↓ Edit ↓ Copy ↓ click on the ... form ↓ Edit ↓ Paste ↓ Edit ↓ Paste**
- **Click on the copied objects, right-click ↓ item Copy in the popup context menu ↓ click on the form on the right mouse button ↓ click Paste to in s a fallen Conte to communities are high menu ↓ several times point Paste ... so quickly.**

Similarly, **Cut** (cut to transfer to another place) and **D e lete** (destroy) .are used

Find and Replace . Search and replace commands in the text of your .program for individual words and fragments

Advanced ↓ Word Wrap . The program text in the code Auto and .cally endure so Xia from line to line

Advanced ↓ Make Uppercase . In the selection makes all letters .Zaghlah in GOVERNMENTAL

Advanced ↓ Make Lowercase . In the selection makes all letters tup h .GOVERNMENTAL

Advanced ↓ Comment Selection (***Uncomment Selection***). These menu items are removed and put quotes comments and riev once in each row selected code snippet you. Why is this needed? In prospect of programmers when debugging is often n e necessity to get a few lines of code do not ful l nyatsya. But it's a pity to tamper with them, since they may still be needed. Then programmers put a single quote at the beginning of each of these lines . VB doom and is that the floor and the page of ki - a comment and does not execute them, as required. When these lines of code will be needed again, the programmer quotes beats and wish to set up. When these lines a lot to put on .and take off the quotes in each line individually ut of mitelno

View .4.4.3

This menu is necessary mainly in order to show the window and panel tools n comrade, and consists mainly of the titles of these windows and pan e lei. Clicking on the desired item - and a window or panel appears on the .screen

Project .4.4.4

This menu is necessary mainly in order to add to the project a variety of computer on -components : forms, classes , modules and others. P Processes . (described in the appropriate places in the book and gi (eg, 21.1.2

Build, Debug, Data .4.4.5

Build is used to compile the project , **Debug** is used when debugging the project. Their servant to the one described in the relevant parts of the .book. **Data** used when working with databases and n GOVERNMENTAL

Format .4.4.6

This menu IME e t deal with the size, shape and location of the item s sound control in Lenia on the form and relative to each other, that is so, with what you already do an excellent job without any menu. However, when the e-mail e cops manage to form a lot and they need to position itself as a pack .of of ordering , this menu helps to save time

The *Format* menu is not available when you are working in a code .window or running a project

The meaning of the menu items is shown in Table 4 . 1 . On the left side of the table you see a ton indoor menu *the Format* . In the central part of the table for convenience, I put sootvets so proper tools panel VS to perform the . same actions. About panels Instr at cops h and thaw in the next section

Create a project. Position on the form several buttons, labels, and other Elem n comrade control of different size and of elongation . Select one or more of the Z OBJECTS on the form . To select multiple objects, click them while holding cells and Viseu *Ctrl* or trace their outline. N Try to use these items . P check that win t smiling. This is the case when it is easier to .understand by experience than to understand the explanations

The meaning of tools	Instr panel at	Format menu

	cops La y out	
The upper left corner of the selected element and control aligned and etsya with plaque and a nearest grid node		
Horizontal alignment of selected controls		
Vertical alignment of selected controls		
Align selected e-Ctrl-I to the width, height, posted e ru		
Alignment of the horizontal gap between the selected el-mi control, increase, .decrease, destruction of it		
Aligning the vertical gap between the selected el-mi control, increasing, .decreasing, destroying it		
Centering the selected e-controls on the form horizontally and vertically		
Bring selected e-mails to the front or back		
Locking" and "unlocking" the allocated" e-mail in the control		

Table 4 . 1

: Explanations

- **Paragraphs Align and Make Same Size alignment performed include and Tel'nykh sound control element in Lenia .allocated last**
- **Order . It happens that as a result of the design of your form nekot on rye element s management overlap others. For example, premises with Titus on the project two large buttons so that one partially or completely covers another. Then the question becomes important - which of the objects is closer to you, and which is further. Manage in lyat that you can select for Bring to the Front (push to re d Nij plan) or of Post Send to the Back (zadv and pull . (the background**
- **Lock Controls . Sometimes, with an awkward mouse movement, you can move or resize an object in your carefully**

constructed project. To prevent this from happening, this point is needed. Objects example of chickpeas out of place. When you want .to unfreeze them again SEL e Rita this item

Tools .4.4.7

Here we are interested in the **Options** item , which allows **you** to **customize the VS environment** , that is, to adjust the appearance and behavior of windows, toolbars and other elements of the environment. Be careful not to change the district and buildings, which you do not understand, otherwise VS will behave in an unusual or indirect t nym way. Some of the . settings take effect immediately, but only after Perez and GRU of ki the VS . (H and you look at the item *the Options* (Fig. 4 . 23

Figure: 4 . 23

On the left you see a list of main *pages* , each of which is dedicated to one side or another of the environment setup. Pages are grouped into *folders* . .Consider those that may be of interest to beginners

The General page of the Environment folder . In Fig. 4 . 23 you can see the page **of General** (General) from the folder **Environment** (Wednesday). Upper switching th ents allows you to choose for a core group

of windows is one of the two ways to represent the windows : with bookmarks and E, to which you have become accustomed to the VS (*Tabbed the Documents*) , and without tabs - one to which of th you are accustomed to in other applications Windows (*MDI environment*) where windows can move independently of each other . You will recall that in the other groups of windows you d e barks choice between these two ways of using them konteks . t Foot menu - *Dockable - the Floating*

On the right (**At startup**) you see a list of what VS greets you with at startup. You have already seen it in Fig. 4 . 22 , in the same place he is op and . dignity

Checkbox *Show status bar* manages the appearance of the bottom line of the screen state I Niya. Remove it if you are short of IU with the one on .the screen

Next check with the regulator allows you to adjust the speed of " vypolzanija " because of the edge of the screen window *Toolbox* and other . windows with the inclusion n nym *the Auto Hide Event*

The Fonts and Colors page of the Environment folder . In Fig . 4 . 24 you see the page as nitsu **Fonts and the Colors** (fonts and colors) of the folder *Environment* . It allows you to select the fonts and colors of .windows intended for text information

Figure: 4 . 24

First, in the *Show settings for* field, we select a window or type of windows for which we will have to configure fonts and colors. In the figure, the *Text Editor is* selected , that is, simply the code window (and, for example , to configure the *Ou t put* window, we need the *Windows Text Output Tool*) . In the *Font* and *Size* fields, we select the font and font size for .this window. Setting the vial and Joc *Bold* , we make the font bold

Now let's do the color. In the *Display items* we s Biran window element for a cat about cerned will customize the color. Thus, *the Text* meaning as a basic text in the code window, *the Selected the Text* - the selected text and *the Keyword* - key (reserved , service) words the VB (you're used to and put their blue) . In the list x *Item foreground* (*foreground* color) and *Item background* (*background* color), we adjust the color of the text itself and the background color for this one to a hundred, respectively . *Custom* buttons .allow you to fine-tune the color

.The *Sample* field allows us to see the result of our settings

The Projects and Solutions page of the Environment folder

. In Fig. 4 . 25, you see a page *the Projects and Solutions* (Projects and . Solutions) in the folder *Environment*

Figure: 4 . 25

In the *Visual Studio Projects location* field, we manually write or use the *Browse* button to select the folder in which VS will by default save new projects. **According to EMA L Chania** it means that a person does not .intervene and does not choose another option

If va meters do not need that to each time it runs, etc. of EKTA to perform appearing elk window of *the Output* , and if an error occurs during start appearing elk window of *the Task a List* , then remove ff e blowing two .FLA Well ka

Switch *the Build and the Run the Options* defines , will be whether the VS before any ZAPU with whom the project on the implementation of automatically save the project (*the Save changes to the open doc all u ments of*), offers small and causeway to save the project (*the Prompt to the Save Changes to the open documents*) or not save (*Don ' t save changes to open .documents*). I recommend the first to the careless

Project folder s . Here we enable or disable *Option Explicit* (see .5.2.4) and two other options

The General page of the Windows Forms Designer folder .

Here we set the horizontal and vertical distance between the grid lines on the

form (**Grid Size**) and have the ability to hide or show this grid (**Show Grid**). Setting the *True* option **SnapToGrid** , we orders and eat in order to outline . the controls only passed along the lines ce t ki

Window .4.4.8

Here we are interested in the **Split** item . It allows you to look at your code h e res two windows rather than through one that is handy when a lot of code. Let your code contain several dozen procedures . All of them, of course, does not fit into the field of view, and to see , for example, the lower procedure, you have to box code , etc. on kruchivat. But then the upper procedures go out of sight. Suppose you need to keep in sight the most faiths x nyuyu and the lowest procedures simultaneously . For this n The items *Split* divides the code window into two windows positioned one above the other (Fig. 4 . 27) . In them you see the same code , just like through two windows in your apartment you see the same courtyard. If you change the code text in one window, you will be able to HC and put these changes in the other. To ach window you can scroll independently and n of the field of view of Mr .and directs any piece of code

To turn the two windows of code into one again, click on this item .(again (which this time will be called *Remove Split*

There is a quicker way to halve the window and reconnect. P on the .place the mouse in the area marked on Fig. 4 . 26 oval

Figure: 4 . 26

The mouse will change its shape. Drag her down. We received two windows .((Fig. 4 . 27

Figure: 4 . 27

.To merge the windows again, move the mouse up in the same motion

Help .4.4.9

When you write a statement in VB , you must, first, understand what it means and what each part means, and second, you must know the rules for writing it. Also used iblioteka the .NET Framework class is extremely rich and , in the not e includes thousands of items - . Properties, events, etc. It is impossible to remember the meaning and ka recording rules Well dogo element. This is where the need and *the Help* - help system, which allows you .to bys t ro find information on any item. Of course, if you know an d liysky

:There are many ways to use VB help . The most common way is this

F1 . If you want to know more about a word from your program , called a sample, *the Width* , you click the mouse put on it the text cursor and click on the cells and viature key **F 1** . Window appears with explanations or instructions (usually it is straight and is connected to the central group of windows). You will learn which class belongs to an inter e popping your property or, say, method. If you are asking about an object, you will find out

which namespace its class belongs to . P on the link e **Members Active** (components) of the help window will learn all the properties, methods, and other items that belong to this class . By the way, you will find more properties here than in the properties window, since here are also those properties that are meaningful only in [run] mode . For example, Propert tonnes in *SelectedText* object of *the TextBox* , which produces a piece of text, .selected by us in the text of the PTO field

The link **Overview** read common words and core material on so called .Nogo stem with meat, etc

The following tools help you find out in paragraph *Help* GLA in Nogo . menu the VS

Dynamic Help . This is an even more convenient way. Select this item. Before you appear on the CCW *the Dynamic the Help* (Dynamic Assistance) . N OMe art and those it into one of the windows groups. When you enter program text, I recommend keeping this window open . When you enter text VB sees , in a few words and which operator you enter, and without your request displays in the window *Dynamic Help* headers references on this th word in and at this mu op e r a torus in , and by drugi m words am in the operator, as well as headers references on concomitant m the am . You just have to select the desired title if necessary . Anal about tech information appears when you simply click e those mouse on any word etc. of a gram. Similar is the useful information regarding the longer words of the VB , and your activity will appear in this window , when you're running in a window to .the n struktora and others about to tries

Contents provides you with information about VB to systematize of Vannes in and de. This is the best place to learn what VB has and what .doesn't. But remember that this is not a textbook, and spra in ka

Index is convenient when you know in advance the name of the element to a tory you are interested in (for example, *Bed and utton*). You enter this word into the field and see the Headers in reference ki this cl of .Wu

Search . This item is useful when you isch e those information not only on the specific words of the VB , but also by a set of words, for .« example, « wi n dow types

Show Start Page . This item is not related to help, but it is useful

.because it brings up the start page on the screen

Panels and tools .4.5

Toolbars are designed for quick and easy access to the tools n there VS of a particular specialization and . In Fig. 4 . 28 you see two toolbars. The upper one is standard, it is always on the screen, because it is needed very often. Its focus is the main operations with projects. We have already used some of its buttons. Hold your mouse over any icon bar - and he will call .you

Figure: 4 . 28

Often the same and n strument that the panels are available in the main menu, but the panels provide a b s Striy access to the tools. Get to know all the buttons on the standard panel. You may find that a certain use of torymi .(are more convenient than the menu items (one click faster of the two

VS has a habit without permission to display panels, which it believes you may come in handy at the moment, and then remove them from the screen when its point sp e Nia they are no longer needed. Hand toolbar brings up so: **View Take ® the Toolbars ®** click on one of a couple of dozen titles Pan e lei. The panels that are already on the screen are marked with a check mark. Remove the panel from the screen can be the same FPIC on a .bong as the cause

Display the *Layout* panel . Her specialization - work with the elements of sound control in Lenia on the form. In 4.4.6 we have already met it indirectly. Most likely she dormancy and zhetsya in the upper part of the window VS next to other panels. In Fig. 4 . 28 you can see it under a standard panel ins t ments. If you want to move it out of there, grab one of the several gray vertical lines separating the tools on this panel with the mouse and draw .the shield

Then you can change the shape of the panel . You do this in much the same way as you change the shape of windows by grabbing their edges with the mouse. In Fig. 4 . 29 you see the same *Layout* panel , which we have already moved and which we changed the shape. Now her mo w but close, .and the cross and the ball

Figure: 4 . 29

While on the form are not allocated controls, buttons on the *Layout* will pale in E and inactive in E , because they have nothing to do. Select multiple controls, the buttons come to life, they can be the servant of a thief. And how - you are already on nyali of 4.4.6

By clicking on the triangle arrow beside the cross in the top right corner of the pan is, whether you can add or remove buttons from the panel. There you can create a prop so vennuyu panel ol from the cops. But you will figure .this out for yourself , if need be

The benefits of the rest of the toolbars will become clear as you further . explore VB

Transferring your project to other .4.6 computers

When your project, for example, *Calculator* , works on your computer , you will want it to work on your friends's computers . In ancient times to

solve this problem it was enough to create the executable file of your project (eg a call it *Kalkulyator.exe*) and copy it to the computer your friend (Mr. de find the executable file, written in 3.12.4) . Now the situation has changed . In modern programming is much more solid and heavy approach to migration, etc. of grams with the computer and on the computer . *Do not copy, but install - that's the motto!* The reasons for rejection of an easy life, I'm not particularly go, but basically the - Prospect of grams written in the VB , too head and screens from Windows and other software programs , and on different computer s operating system is Windows and system software are different , n about copying the launch file is not enough for this, you need to somehow reconfigure Wi n dows and possibly install some system programs. All this is included in the concept of insta l lyatsii your program on .a foreign computer

This does not mean that you can not try to install without any just ospreys and Rowan executable unpretentious project to another computer .with the same system m nym software . It is quite possible that he and run

What is installation . If you have ever installed any and g py on the computer , then you know that this installation. So you bought a CD with ponra and curled a game or program (let it be the Microsoft of Office). In many cases ku n Retained program just will not start, you need **the installation** , or , in another way , **the installation** (from the English *the install* - install). D on voryat that the CD drive is not the program itself is the Microsoft of Office , and **installs I insulative package** program the Microsoft of Office , ie Mr. and boron files in which to go to the compressed files and the Microsoft of Office . You will find in and n stallyatsionnom package installer (it runs the file often nazyv and etsya **the Setup.exe** , although it may be other names) and run it. All your further actions are dictated by the program , which has called and is called **the Setup Wizard** . In fact, you only have to answer her questions. For example, you are answering the question to which folder on your hard drive you want to install Microsoft Office . The setup program expands compressed files Microsoft Office , and copy them to the railway e stky drive of your computer as well. In addition, it configures Windows to work with Microsoft Office . After installation you can run Microsoft Office from the start menu " Start " on the . taskbar Wi n dows

During the installation of the games you might , come to mind thoughts

of superprogra m Urban Areas, which is not enough that create the game, they also create an installation program and do so that the installer correctly installed the game on your computer . Xie second hour, I invite you to become such a " superprogrammistom " . Our goal - in half an hour to get to the hard drive folder - Installation first package your program , Goth to you .first -to-eat. Then you can copy it to a CD and let rit drusen s pits

As an example, create an installation package for our calculator , created the first in 3.3 . If you want to create an installation package for other adj about zheniya - it does not matter, my explanation does not depend on the .installed App e Nia

First stage - project preparation .4.6.1

In the code of your project , operators like

```
(" Cl_sections . Image = Image . FromFile (" D : \ Photos \ Earth . JPG
```

or

```
"Player .FileName = "D: \ WINNT \ Media \ canyon.mid
```

These operators are united by the presence of file addresses in them . This means that in the operation of your project can obraschatsya s Xia to graphic, audio and other files, tries to dyaschimsya somewhere else on the disk. You never have occurred to me, what would happen if someone, not knowing that the files need to rather necks program, erase them? Nothing good will come. And if you also intend your project to be installed on .Chuzhoe second computer , danger and confusion doubles second not

Hence the advice : All files are pre-copy the folder *BIN* folder inside your e th Ave about EKTA, to where you are and tsya executable file of your project . According to the principle “ I carry everything with me ” . Since after the start of the project, this folder is the current (see. 02/19/12), then : also be simplified and addresses in the above op e Rhatore

```
(" Cl _ division . Image = Image . FromFile (" Earth . JPG
```

```
" Player . FileName = " canyon . Mid
```

Pictures that you have uploaded to the objects on the stage of designing , I do not need all of these efforts, since they are already implicitly stored in .one of the files in the folder, etc. of EKTA

I recommend also to the inception of the project to keep it beyond the

pre Dr. Laga name *WindowsApplication 1* , but with a unique name, such as *a . calculator , I torr*

The second stage - creating an installation .4.6.2 package

So, we have a project *calculator I Torr* , ready to ensure that from it you can create an installation package. In the VS environment, the installation package is created as follows. First Sun is, of a proof create a new project of a special type, which is a tool for the project *calculator* and Mr. stallyatsionnogo package. Then the programmer performs certain actions in this new project to set up the future and n stlation package. This new project is then compiled in the usual way. P e result of compilation and is so Xia .installation package

.Let's go through these steps in order

We create an installation project . Run the VS . Then **the File ® the New ® by Project ®** in the left field of the project to create a window (. Figure 4 . 30) , select *the Setup and the Deployment the Projects* , and the right - *the Setup by Project* ® come up with a name for the installation etc. . of EKTA, for example, *My to Falk in trimmer* . **↓ OK**

Figure: 4 . thirty

The installation project and the decision set s . Tool with h given (Fig. .(4 . 31

Figure: 4 . 31

If you are going to install the Web - application, you would need to
. select *the Web the Setup by Project*

Add a new project . Now the tools you need to explain something to it should tr y ditsya to get the installation package. This "subject of labor" will be just our *Calculator* project . H uzhno simply add to a project *status* in the decision. This is done in the usual way: **File ↓ Add Project ↓ ... Existing P r o ject ↓**

Thus, our solution is now consists of two projects: the installation of the project *My to alkulyator* and the project *calculator* (Fig. 4 . 32) . Obviously, the prospect of JECTS may not have the same name , otherwise VB confused. For more information about MULTI s FIR projects in the decision
. and of the *Solution Explorer* is written in 21.3

Figure: 4 . 32

Determining the content of the future installation . Next step. Select the *My calculator* project in *Solution Explorer* . Next : **Project** ↓ **Add** ↓ **P roject Output** . P e ed you arise window (Fig. 4 . 33 .

Figure: 4 . 33

Make sure that all information in the fields of the window is the same as in the figure. In this window, we point out that the installation package should be created specifically for proe to that *calculator* (top field) and define the Table of Contents of the work of the installation package (the list in the center of the window) . Dedicated e lement *the Primary o utput* means that the results s Tat installation will be the main thing that is required, namely an executable file, etc. , and Proposition (*the exe*) or library (*dll*). However, p ezulta that installation mo gu m be others in Gia , helper objects , e.g., files with the source program . When n e necessity, you can highlight the list of all . the elements , but right now it suffices as exactly *the Primary o utput* Click OK. In the *Solution Explorer* project *My to alkulyator* began to . (fill with content (Fig. 4 . 34

Figure: 4 . 34

File System window . Let's talk about the structure of the window *File System* our installation n Foot project *My to alkulyator* that you see in the picture. It allows you to customize the installation. In his left panel you can see schematic e skoe view of some (related to the installation) file elements in B with the theme of the computer on which you will install the :application. That is the meaning of these e e ments

File system of the computer on which the application will be installed and installed	*File System on Target Machine*

Application folder	*Application Folder*
User desktop	*User's Desktop*
Programs in the user start menu	*User's Programs Menu*

In the right pane, you see the contents of the selected item in the left .pane. While the inside of the last two elements is empty

We remove the manufacturer . Let's start setting up the application folder. Select the *Application Folder* element and go to the properties window . Property value *DefaultLocation* (default location) is a : kind from a Karlovna scheme

[ProgramFilesFolder] [Manufacturer] \ [ProductName]

He rather means that files To install the CSO application I located by default so camping on a computer in a folder with the name of the application (*ProductName*) , which is sprayed on the Laga in the folder with the name of the application manufacturer (*Manufacturer*) , which pa to rely folder inside *Program Files* . You can leave everything as it is, or you can change something if you know the mechanics of this scheme. Let's take, for example, :produces discard e la. P about is what shines

[ProgramFilesFolder] \ [ProductName]

Start menu shortcut . Now we take care of that in the start menu floor s zovatelya proved to label our application, otherwise the user is not able t run our App e of from the start menu. To do this, select the *Application Folder* element in the left pane , right-click on the *Primary o utput element* in the right pane and select *Create Shortcut to Primary o utput* in the context menu . Label with length n nym name is added in the forward folder and . (Proposition (Fig. 4 . 35

Figure: 4 . 35

There's nothing to do, drag it to the folder *the User's the Programs the Menu* , and then rename the first ones into something used to Leah digestible and a

. short , for example, *Calculator*

Add the file 's in your application folder . If your project code *Calculator in trimmer* has links to any external files, to which I referred in the previous by e Section e le, these files must be included in the installation :package . Let the code Project *Calculator* is such op e Rathore

```
(" Cl _ division . Image = Image . FromFile (" Earth . JPG
```

This means that in the folder *BIN* inside the project folder *Calculator* has msgs e sky file *Earth.JPG* , which the project uses during operation. This file must be included in the installation package . For this isolated Yai m in *Solution Explorer* project *My to alkulyator* , followed **by Project ® the Add ® the File ®** n and go on a disk file, and do it on a double-click. The file appears in the folder, etc. , and Proposition (Fig. 4 . 36) . After installation .on the user's computer, it will be there, too, in the App folder e Nia

Figure: 4 . 36

Add files to other folders . If you want, you can make when you install any other files you need not add to the folder, etc. and Proposition and in other folders on your computer. Right-click m s shek element *the File the System on the Target Machine* , and then click one of the suggested folder, for example, *the Windows Folder* . It will appear in the window *the File the* .(*System* next to the other (Fig. 4 . 37

Figure: 4 . 37

Now add the application folder multiple content you want, and then per e drag them out to the folder *the Windows Folder* . After installation, they will appear on the user 's computer in the Windows folder (which I generally .(do not recommend doing

Application icon . If you want to run the file to your App e Nia had an icon different from the boring default icon triggering F nd fishing, proceed as follows. Right-click in the window *Solution Explorer* on the project *calculator* and shortcut m e nu select *Properties has* . In the resulting Properties window (Fig. 4 . 38), select *the Build* . By clicking on the button .with three dots, select the appropriate icon. Then OK

Figure: 4 . 38

This does not mean that your application will have this icon on the taskbar. To do this, as written in 3.4.3 , set the appropriate property *Icon* fo r .us

Setting up the configuration . In ashi steps : **the Build ® Co n**

.(**figuration Manager** . A window appears (Fig. 4 . 39

Figure: 4 . 39

Projects can have configuration and tion *Debug* , convenient for debugging, and may have a configuration *Release* , necessary, when all debugged application and issued (*Release*) in high light. We debug already do not need anything, so in the top list Choosing and eating *the Release* . . Everything should be as in the picture. Click *Close*

Compilation . We have made the basic necessary settings. The final and to the cord - a compilation of the installation package. For this - **Build ↓ Build Solution** . Neck of the given period of time the computer compiles (and compile both projects) , in the window *Output* n about are messages that finally in the status bar there is a message *the Build Succeeded* . This means .that creating and Mr. stallyatsionnogo package is successfully completed

Where is it - our installation package? Exit VS to Windows . Look at na n ku project *My to alkulyator* . The *Release* folder appeared there . In it you can see the *My Calculator* ***.msi*** file among other files . This file is our .installation n and kW

Stage three - installation .4.6.3

I have already briefly described the installation at the beginning of the section . Let's start with the fact that as the experiment was carried out it install on the same Compute e p e , where you created the installation . package

Double click the *My calculator.msi* file . The installation has started. . **1**
The installation wizard window (Fig. 4 . 40) warns that those who install it without your permission, go to jail. After this, le g kim heart can press *the* . *Next*

Master is a program that allows the user d f lat complex in e soup with a minimum of mental powers. Typically, the wizard consists of several steps, each of which sets the master user Issues on sy. The transition to the next step is carried out with the *Next* button . Back you m of zhete click *the Back* , . abort the installation - click *the Cancel*

Figure: 4 . 40

In the next window (Fig. 4 . 41), you are invited to address the . **2**
folder in which will have to tanavlivatsya your program. If you do not like, click *Browse* and s take the desired folder or enter the address in the text box. Use the switch below to set who will use this program: all computer . users (*Everyone*) or only you (*Just me*). *Next*

Figure: 4 . 41

The next window (*Confirm The Confirm the Installation*) offers . **3**
just in case ACK p dit that you want to start the installation (after all, it is related to writing files on your computer u ter and b og receptacle and a can
. end). *Next*
.(There is a setup process (Fig. 4 . 42 . **4**

Figure: 4 . 42

The last window says that the installation is complete (*Installation* . **5** . *Complete*). Mo Well but to press *the Close*

The program has been installed. Now you can run it from the Start .menu "Start" , there appears l Xia desired label

You will be useful to look into the folder *My to alkulyator* in the folder *Program Files* , where to install your program, and to discover there the . launch th conductive file *To alkulyator* . *exe* and graphic file *Earth.JPG*

. Installation on another computer

To install your software on another computer ie , copy any FPIC about bong installation file *My kalkulyator.msi* on a drive to the computer and and run it there (copy you can, for example , through a network or floppy disk, or . (the CD - the R the W

The problem to date (beginning of 200 5) is that not on ka w home to m computer's installation can be successfully realized. However, this problem should be a thing of the past once the .NET Framework becomes a standard . part of Wi n dows

Part II . Programming and of at V] ne p vy level

Our sweet life is over. Rather, there has been a big break in it. Deiss t pheno-, our life was carefree . After all, the calculator and the player took vivo Recording us without labor . Little work - lots of fun. I wish that it was always ... But life is harsh and offers a traditional choice - or roll up your sleeves or over e be easily th running shoes. Indeed, we created a calculator and a player, but do you have a feeling that you can do something else .yourself? Unlikely, because you were led by the handle

And yet ... e nce there is still room almost without programming quick n of the beam and be anything " ed as something " ? There are, and there are many of them. In popular magazines devoted to programming, you will find a lot of programmok of two or three lines of code, Copy of the Vav are in your computer , you will get and the ones on the screen (and not only it) interesting things. Only the trouble is that even though there are only two or three of these lines, there is no way to truly understand them and you cannot do anything similar on your own. Who are you in t and whom the case - the programmer or copyist characters? Here it is! And you can't get away from this fact. Programming - is the mountain , which is not obedesh is the same .granite of science, which is n p and children camping nibble

Tip: gnaw quietly, do not try to bite off just a big piece - mo w but .break teeth

This is where real programming begins. Its purpose is to guide you from creating the simplest programs to the most complex ones. You will learn such serious things as variables, loops, branches, procedures , graphics , working with time, computer control with the mouse and keyboard . Details versed with s danie three pretty solid projects. Ends of the assignment for self I tion creation project . It is assumed that after completing this task, you should have a feeling of omnipotence . In You should feel about Vat, now you on the .shoulder of any software posted e ra

This part consists of 10 chapters. Not all readers will walk through their fire and water. But with those who pass, you can go to reconnaissance. Along the way you will find a lot of interesting things: graphics, designs, patterns, movement of objects on the screen , cartoons, dialogue with the computer ohms measured and Tel shustrosti determinant - "psychic" Do you, your password on the calculator, the project "B y dilnik " and ne p -hand in your life own game " Race " . Let's hit the road!

Chapter 5. Variables

To do in VB something more substantial than a primitive e calculators I Torr and a player, we have to dramatically increase the amount of knowledge about VB . The first step of the forest so Nice embankment ora knowledge of th - variables. Beats on of discontent more to Holmes than Jackie Chan . But ?what can make Jackie Chan Ho l IOOC

N onyatie of claim AC line x values ah . .5.1 Assignment operator

You know the concept of a variable from school mathematics. Let n ie how many years ago was equal to your height 130 cm Let this fact so:. *R = 130* . Now it is equal to 1 6 0 cm, that is, *r = 160* . It turns out that the value of *r* has changed. Therefore, it is called s INDICATES variable. Numbers . 130, 1 6 0 are termed **values** variable e masks *r*

Any programming language knows how to handle variables. Without them it would have been very weak and could be drawn from the computer is only possible Kalk in insulator . Likewise, algebra without a variable would turn into arithmetic. However, the advantages of the use of variables in the computer , we will open later, and now our task - to them, etc. , and become .used

The concept of a Emperor of the e assignment .5.1.1

So what can we do with variables while programming in VB ? Pre w de of all, we can ask the computer the value of a particular variable. This we can do with the help of an operator, which is called **the operator APPROPRIATE and Bani** . So, if we want to say that *as* a znach e of 6 , you should just n You can record in the code window *and = 6* . W Recording *a = 6* and is called the assignment operator. It is said that the value *a is* assigned the value 6 . Since performing e of the operator *a = 6,* the computer used in children to remember that *as* equal to six. No need to check it out while on the computer e , I 'll tell you when it is necessary for him with a .ditsya

Note : In older versions of BASIC assignment operator for the clarity of and credited with the so

."- *for Let a = 6* , which meant "Let a = 6

We made extensive use of the assignment operator in part I of . For example, in the opera about re

Button2.BackColor = Color.Red

we set the color property of our button to red. For now, so as not to distribute attention, we will not often touch objects, their properties and values in the assignment operator. We confine mostly variable quantities having chi with .lennye receptacle and cheniya

II After performing the following program fragment

```
a = 2 * 3 + 4
b = a
y = a + b + 1
```

the computer will know that *a* is 10, *b* is 10, *y* is equal to 21. P Check the in . the mind

It turns out that using the assignment operator, you can calculate too. We see that with the right of the equal sign in an assignment statement mo w but write not only the number but also the Move n values by, and **expressions** . VB expressions can be of different types, but we'll talk about that later. P eye under expressed , and it will be understood **arithmetic expression** , that is, so th , you are used to in school mathematics e . Here we see three in s expressions: *2 + 3 * 4* , *a* , *a + b + 1* . Any arithmetic .expression has a numerical receptacle and chenie

Now a little about the properties of objects. An expression can include not only variable values, but also properties of objects that have a suitable meaning. For example and measures

y = a + b + Button 1. Width

Let *a* be 1, *b* be 1 , and the button width is 1 00 . Then *y* becomes equal . to 102

On the left side of the assignment operator, there can also be not only a ,variable, but also an object property. For example

Button 1 .Width = a + b + d

D The action of the last statement and you will immediately see on the screen. Let *a* be 1 00 , *b* be 1 00 and *d* be 100 . Then the width of the button .will be 300

:A couple more examples

REMEMBER THAT Comp U TER	FRAGMENT OF GRAMS
v ***8*** h ***80*** s ***88.01***	*v = -2 + 10: h = 10 * v: s = v + h + 0.01*
t ***0*** n ***40*** z ***-40***	*t = 0: n = 2 * t + 40: z = -n*

Determine orally to Which value is assigned to the variable *t* n for
: follows execution fragment

*k = 1 + 2 : s = 2 * k : t = 6 - s*

How to see the value of a variable .5.1.2

It should be remembered that if to the left of the equal sign is the variable and on, then VB executes the assignment operator " in the mind " . This means that the result of it in s complements is not displayed, but only remembered. You will look at an ekran, on which nothing has changed, and you will not know not only the value of the variable, but even whether the operator was executed at all or not. How, then, do you know the meaning of a .variable? The first method is t and Coy

The m YTC e and text m field . With adites at the computer, with Compose your project, it - button, label and a text box . Record for the button
: so as kuyu procedure

```
Private Sub Button1_Click (ByVal sender As System.Object, ByVal e As
System.EventArgs) Handles Button1.Click
    Dim a, b
    a = 5 + 5
    b = a + 2
    Label1. Text = a
    TextBox1. Text = b
End Sub
```

The *Dim* statement "declares" variables and ranks. Why this is needed is written in 5.2 , but for now I will say that all the variables must first be

declared, and only then used. Z and a blank project, and then clicking on the button, you will see in the label the number 10, and in the text box - number .12

With Debug.WriteLine . D A training purposes and to debug programs im e hundred labels and text fields it is more convenient to use the operator **the Debug . WriteLine** (its use is already described in 2.1.2) . : Create a second button and record for her facials at ru

```
Private Sub Button2_Click ( ByVal sender As System.Object, ByVal e As
System.EventArgs) Handles Button2.Click
    Dim a, b
    a = 5 + 5
    b = a + 2
    (Debug.WriteLine (a
    ( Debug . WriteLine ( b
End Sub
```

After running the project and clicking on the button, you will see the numbers 10 and 12 in the *Output* window . If u do not see the window and *the Output* , then your actions: **View Take ® Other the Windows ® the . Output**

There are many other ways to see the values of variables, but they are .on of the same

?What is the use of variables .5.1.3

Very tiny and simple benefit is that using them is convenient to solve carried Well nye computing tasks. For example, given the parties etc. I rectangle: a = 27018, b = 3954. Calculate its area and perimeter (remember .(that the perimeter is the sum of the lengths of the sides of the rectangle

Create forth about EKT button. The problem is solved by the following :program

```
Private Sub Button1_Click ( ByVal sender As System.Object, ByVal e As
System.EventArgs) Handles Button1.Click
    Dim a , b , S , p
    a = 27018                'Rectangle length
    b = 3954                 'Width of the rectangle
    S = a * b ' Area
    p = a + a + b + b        ' Perimeter
```

```
(Debug.WriteLine (S
(Debug.WriteLine (p
End Sub
```

After starting and clicking on the button, you will see the following two : numbers in the *Output* window

```
106829172
61944
```

The first of these numbers is the area, the second is the perimeter. Without the use of the values of variables operators to receive more : cumbersome and less item on ble

```
S = 27018 * 3954                    'Area
p = 27018 + 27018 + 3954 + 3954     'Perimeter
```

which is especially noticeable in large programs. In addition, to change the length or width, you just need to change the number in one place in the .program, and not in three

There are lots and other benefits that you will feel in the next Chapter and the .Islands

The meaning of the assignment operator .5.1.4

Keep in mind that in an assignment operator, the left of the equal sign can not be a number or an expression. You can write *c = 34* , but you can't write *34 = c* . You can n and sat *z = f-v + 990* , but you can not write the *f-v + 990 the z =* . This rule is adopted based on the mechanics of the assignment operator . About Emperor of the assignment is designed so that at first looks or calculates how important the right part, and then, etc. and allocates a value is what stands in the left part. The fact that the right of the equal sign is assigned to the fact that ff e Islands of the equal sign, and not vice versa. It .makes no sense to assign a value to the number or s rage e NIJ

Note that even on one important point. When a student sees an expression (for example, *d + 2d*), he does not necessarily evaluate it. He can transform it or, say, simplify it (getting *3d*) , without even asking what *d* is equal to . On the other hand, the computer, seeing an expression, may first simplify it in the mind , but then be sure to subtract it from lit. And for this, he must know the numerical values of the quantities included in him (in our case, this is the value *d*). Thus, calculating the right-hand side of the pr and pivoting operator (for example, *y = a + b + 1*), the computer must know in advance what the variables that make up this right-hand side are equal to (in

our case, these are *a* and *b*). About t respectively for this knowledge lies entirely on the programmer. Let forgetful prospect of programmers recorded a snippet: ... *a = 10: y = a + b + 1* ... anywhere in the program are not giving *b* irrelevant. Of course, the calculation of the regex and zheniya *a + b + 1*, the computer will not know what is the *b* . In such a situation, different programming languages act differently. Some " evil " languages simply refuse to evaluate an expression e , " Cova p ny " Turbo Pascal can substitute im e hundred and *b* any nonsense (and molchok, by the way) . VB kind - it .Substituting e r im e hundred and *b* is zero

Let's check. Force the VB in our latest project about the rectangle is not ful l nyat operator *a = 27018* . To do this, you can just delete, but more convenient *zakomment and Rowan* : put a quotation mark in front of him, :then VB will think that this is a comment and ful l nyat will not

```
a = 27018 ' Rectangle length '
```

:Here's the result

```
0
7908
```

.Indeed, everything was done as if *a* were equal to zero

Declaring Variables .5.2

When the ball to the Countess N comes the Duke of M , the servant announces to the whole room, "Ge p COX M ! ", Why all the guests will recognize that in front of them is a duke, and not just any b and Ron, and turning to him accordingly. Everything is good, everything is decent. If the servant missed and announced guest, forgetting to specify its title , in most cases, ie about the same nothing terrible has happened. Sometimes, however, could still befall d of sore misunderstandings. For example, the Duke might be addressed like this: “ Hey, man, champagne ! " , And this is already ska n !gave

Ball is a project. Guests - the variables. A servant is a programmer . Until now, m s declared all guests . And they did it with the *Dim* operator . However, it declared Democrat hours but without titles . About dnako no excesses were not, everything went like clockwork. This claim about the fact , that the VB smart, he's eyes determines who the Duke, and who is not. But not every guest's eyes tell the truth. We need to sort this out properly. But

. before we study another useful and pleasant e e ment the VB - *the InputBox*

InputBox .5.2.1

How can we ask , inform the computer of any quantity? The assignment operator is times. Now let's consider another convenient and simple way - . **InputBox**

In your project on the rectangle you a hundred times to press the button - p e results of all time will be the same. This is because iskho e nye data *a = 27018* and *b = 3954* no change. Boring. It would be nice if the computer asked us every time the button was pressed what the sides of the rectangle were equal to this time . And we have his holes e Chali. And paragraph about .how he figured

To do this, we need to slightly change the program . Copy the folder of the previous project and work on the copy. Here's what you should have in :your button click procedure

```
Private Sub Button1_Click ( ByVal sender As System.Object, ByVal e As
System.EventArgs) Handles Button1.Click
    Dim a, b, S
    ("a = InputBox ("Enter the length of the rectangle
    ("b = InputBox ("Enter the width of the rectangle
    S = a * b                    ' Area
    (Debug.WriteLine (S
End Sub
```

Compare with the previous program . As you can see, the first two lines have been replaced: *a = 27018* and *b = 3954* . Also, for educational reasons, .everything related to the perimeter was thrown out

Design

("a = InputBox ("Enter a rectangle for another

there is an assignment operator, and means she ordered the computer from . the display on the first e to a window wounds , as Fig. 5 . 1

Figure: 5 . 1

P After this second person enters into the white text field of the window any chi with lo and click OK. Next, the computer assigns *a* value entered and .continues to p a bot, that is, proceeds to cl e blowing the operator

If you do not like the line *WindowsApplication1* in the title bar, you can : h and give his line, adding to the operator as follows

("a = InputBox ("Enter the length of the rectangle", "Data input

. You can see the result in Fig. 5 . 2

Figure: 5 . 2

.Run the project. Check it out

If you enter you press is not OK, and click *Cancel* or cr e stick, the VB said that you do not "cancel the entry," and introduced the so-called empty string (what it is, it turned out, now it is enough to know that multiplies over .(it is pointless with strings empty and VB will generate an error

Do not enter as non-numeric information, or VB for the same reason . will give oshi b ku

The type must be specified in the declaration .5.2.2

:Let's try to return our program to the ability to calculate the perimeter

```
Private Sub Button1_Click ( ByVal sender As System.Object, ByVal e As
```

```
System.EventArgs) Handles Button1.Click
    Dim a, b, S, p
("a = InputBox ("Enter the length of the rectangle
("b = InputBox ("Enter the width of the rectangle
    S = a * b                    ' Area
    p = a + a + b + b            ' Perimeter
(Debug.WriteLine (S
(Debug.WriteLine (p
End Sub
```

Z Start the project. Enter the length and width of 3 2. You will receive :the results unexpected s tat

```
6
3322
```

The area is 6. That's right. But the perimeter should be 10, not 3322. Where did these 3322 come from? Why, this is nothing but two threes and . two twos! And they were derived from *a* + *a* + *b* + *b*

Here is the same case as with the calculator (see 3.3). T s was the same ? with a mine, until we ave and menili *Val* . Why does this happen

First : the VB considers the contents of the text field in the window *the InputBox* (exactly the same as regular text fields) text instead of chi with scrap, even if it consists of numbers. If so, then the value of variable x *a* and *b* , taken from the text field, it also MF and melting teksto m instead of !numbers and E. Do not drink from a hoof - you will become a kid

Second: the VB knows that the rules of the language sign **+** has several meanings in the head and pending on that to which it is applied. In particular, when working with numbers, this is an order to add, but when working with text, this is not an order to add, but an order to line up. And if so, then the .result was the same as we saw

Why, then, was the area calculated normally? Because there was not complicated e of, and multiplication. VB knows that according to the rules of language multiplication sign ***** is not smy with la when working with text. This means (the smart and helpful VB guesses) my lord - the person does not want the variables *a* and *b to* have a text value , otherwise he would not use the multiplication sign, because my master is smart. So have they Chyslau th znach e of ! Check: slip the computer at the numbers instead of letters - the . VB will protest when in s numerical square of the show mercy

When calculating the area, VB guessed correctly, but when calculating the perimeter, his mind was not enough. Well what ? - now apply *Val* everywhere when adding ? I do not want, etc. of gram s are prepared cumbersome. And is it really all about addition? How did s naeshsya, where ?in the next st conductive time podstelit straw

So : in order not to overtax the mental capacity VB and protect themselves from trouble, it is necessary when declaring variables values . (! indicate their so-called s Vai **type** (at the ball - a title

What it is? Let's make a small addition to the *Dim* statement . The rest of the East and Wim procedure unchanged. Here that at us will (I add bold : (Sri f that

```
Private Sub Button1_Click ( ByVal sender As System.Object, ByVal e As
System.EventArgs) Handles Button1.Click
    Dim a, b, S, p As Integer
    ("a = InputBox ("Enter the length of the rectangle
    ("b = InputBox ("Enter the width of the rectangle
    S = a * b                    ' Area
    p = a + a + b + b            ' Perimeter
    (Debug.WriteLine (S
    (Debug.WriteLine (p
End Sub
```

The operator of byavleni I

Integer	**As**	a, b, S, p	**Dim**

:translated like this

intact s Numbers and	**as**	a, b, S, p	**N declare AC line s Velich and n s**

ie you agree VB contact with variable bubbled values s *a* , *b* , *the S* , *p* , both whole and Numbers s and nothing else. It says that the variable s *a* , *b* , *S* , *p* . IME w t *type Integer*

For now, we will write operators about declarations in the code window only inside procedures , at their beginning . For more information on places to make announcements and methods , see 21.9 , but it's too early for you to . know

.Run the project. Check it out. Everything is working fine now

Variable types in VB .5.2.3

The types Integer and Double . How many rights at our ball at Baron *Integer* ? Dare t Rome. Let's try to enter fractional value of the length of the straight gon - two point one decimal I thuja - 2.1 . That's right - with a comma, not a period . (I assume that your Windows is set to Russia, where a comma is written in fractional numbers. Otherwise, you need to write the h - ku, as is done in the West , and not only in the program, but in the text item on Lyakh) . Let's see the result. VB in the calculation considers *a* whole number 2. And now - 2.8 . VB considers *a to be* an integer 3. That's right. Our Baron , as he expected, the height of the coma river but refuses to be a fractional number, and so any number that we offer to him, rounded up. T ak . Device for en type of *an Integer*

Suppose we are unhappy with this. We wish that all numbers could be and fractional too . You are welcome! To do this, it is enough to declare all :the guests princes. And of meniv operas and torus *Dim* in on this

```
Dim a, b, S, p As Doub le
```

The operator of byavleni I

Doubl e	**As**	a, b, S, p	**Dim**

:translated like this

ten hours n s crushed and dual hydrochloric accuracy	**as**	a, b, S, p	**I declare**

ie you agree VB to deal with variables *a* , *b* , *the S* , *p* , both with ten hours GOVERNMENTAL fractions and nothing else. Decimal - is not only the number, less united and Tzu. No - this is generally any number that can have an integer and fractional part (for example , measures, 27.3908) . In particular, it is any integer. On sense dual hydrochloric PoPo accuracy of the .same

The legitimate question is: why do we need an *Integer* type when *Doub le is* obviously " much richer and better " ? Answer: Not at all, and it is not always better, otherwise it would not have CREATE and Vat at all. . Undermining used her about it - in 5.4.5

List of VB types . The ball to the VB enter not only the barons and

princes, that is, type n e variables in VB is quite a lot and each is useful in its own way. I will consider new types as needed. The types of data you'll find in this book are:

A type	Description	Link
Byte	(Integer - very short, positive (0 to 255	5.4.3
Short	(Integer - short (-32768 to 32767	5.4.3
Integer	Integer - regular (-2147483648 to (2147483647	5.4.3
Long	Integer - Long (-9223372036854775808 (to 9223372036854775807	5.4.3
Single	A esyati h Nye fraction NORMAL second precision	5.4.4
Double	D esyati h naya fraction of double precision	5.4.4
Dec i mal	A esyati h Nye fraction long , absolute hydrochloric accuracy	5.4.5
Char	Symbol (any of 65536 characters, letters, (numbers and other B m oxen	19.1.2
String	(String (free text	19.1
Boolean	(Boolean type (true-false	7.5.4
Date	Date and time	13.1
	Array s	Chapter .15
Enumerations	Enumerations. Many types of transfers .and all with different s mi names	13.3
Structure s	Structures are custom types. There are many types of structures and all with .different names	16.3
Classes	Types of classes and a huge number of .all different it is, us	21.4
Object	Object - all-encompassing chameleon type	11.5.1

For each of the types mentioned in the table, you can declare variables

. using the *Dim* statement

Why You Can't Not Declare Variables .5.2.4

The hostess of the ball makes sure that the servant announces all the guests, even without mentioning the title. VB makes sure that the programmer declares all variables, even without specifying a type (this is called *Option Explicit*) . Therefore, e If you try to do without the operator *the Dim* , the VB will give wo b schenie error. If it does not issue, then *Option* . *Explicit* off , in which case h and thaw 4.4.7

In 5.2.2 I showed you the benefits of specifying a type in a *Dim* statement . What is the use of the *Dim* operator without specifying the type ? What is the use of a scabbard without a saber? We'll see. One g dy, a long time ago, I was tired of having to spend time on the "unnecessary" ad. And I set up a VB so that it is no longer strictly watch the announcement Move n GOVERNMENTAL. Then he wrote *Dim* "every other time". For a while all :went well, but then one chap at the pom snippet stalled. Here's this snippet

```
x = 2    ...
( Debug . WriteLine ( x
```

What do you think he printed for me? If you think 2, you are wrong. Didn't print anything. I thought: This is probably because I didn't write *Dim* . : I added *Dim*

```
Dim x = 2      ...
( Debug . WriteLine ( x
```

Did not help. Again it does not print anything. How so? - I thought - I did everything as it should be. You never know that I have *Option Explicit* disabled? - After all, *Dim* is there ! VB didn't find any errors in the program. I could not understand what was going on, until he decided to de st update themselves for a long time to justify themselves to the principle: If you can not catch the mistake, bring everything else up naivozmozhneyshego ideal, even if it seems bessmy with lennym. "Meaningless" and The ideal would Inc. th chit *the Option Explicit statement* , and I turned it on. And immediately VB underlined the name *x* in the second line of the fragment and said that it was not declared. How is it “not announced”? - I thought, but then I realized that the first line of the fragment, etc. , and absent the Latin letter .""X", and in W a swarm of line I accidentally wrote the Russian letter "ha

So you can see that on our human distraction, many variables can slip

the ball unannounced , at least because , that we are easily confused it Latin,
: and a steam with skie letters of the same outline. Conclusion s

. Never disable Option Explicit
. The Always announced yte variable values
. N When this possibly indicating yte their type

Two options . You may have noticed that is variable th values in possible first Ob I am in , and be , and subsequently , etc. and svoit value to :it

```
Dim a As Integer
... ... ... ...
a = 20
```

:but you can do it with one operator

```
Dim a As Integer   = 20
```

Variables and memory .5.3

The Countess took each guest a room in the palace . The size of the room depends on the title. For us, the palace is RAM , guests are variables, and rooms are so-called *cells* . By l Nosta mechanism of the variables do not understand, if you do not know how they are stored in RAM to m pewter as .well

RAM memory is needed a computer in order to store in performing e Nia program the program itself, and data (including variables) with to about torymi program works. You can read about the slave about those memories. .in Appendix 1

When starting a project and later, the computer allocates space for each variable in memory. It is called a **cell** . In the palace size rooms of are measured in square meters, in the memory cell size is expressed in bytes. Prince need a VM e stitelnaya to m nata than Baron. Type variable *Doub le* need more capacious cell than n e V-belt type *Integer* . So it really is: a variable of type *Integer* in the memory cell is given ra h mer 4 bytes, and a variable type of *the Doub le* - 8 byte s . Just do not think that the pen is, meline type *Integer* can be a number of longer than 4 digits, and the variable type *the Doub le* can not be a number greater than 8 digits . To our PC and

with polzuet to record numbers in a more compact cell with istemu than for .recording characters on fishing : in 1 byte fit two odd qi f ry numbers

Also , do not think that the "walls" between the cells in the RAM are physically tight, like the walls between the rooms of a palace. When performing each following e of the project , and even in the process of performing the same "palace" is reconstructed . But we are not interested in the physical level. We may yet think that on the whole Length e RI "holiday" .any change n naya lives quietly in his cell unchanged

What does the assignment operator with n .5.3.1 and myatyu

I have just talked about the assignment operator, using regex such as zheniya as " the computer knows " , " computer remembers " "," computer assigns " . But we need a more rigorous understanding of the work of the . operator closer to the understanding of " w e climb " to RAM

: Let's consider an example program

```
Private Sub Button1_Click ( ByVal sender As System.Object, ByVal e As
System.EventArgs) Handles Button1.Click
    Dim a, b, y As Integer
    a = 10
    b = 6
    y = a + b + 1
    (Debug.WriteLine (y + 200
End Sub
```

There are three variables in the program, all of them are declared . The computer will allocate three four hbyte cells in memory for them . Here's how : the pile and pile operator will work

When executing an assignment operator (for example, `y = a + b + 1` **) , the computer first looks at its right side (** `a + b + 1` **) . If it encounters a variable (in this case** `a` **and** `b` **) , then the computer before calculating seeking their values allotted n GOVERNMENTAL memory cells (and finds a 10 and 6, as they are there to this recorded op e operator A = 10 and b = 6) , plugs these values into the right-hand side and computes it. W and the calculated value (17), the computer writes to the memory location allotted N hydrochloric Move under N**

. (hydrochloric posed on the left side (y

Thus, when we say "Computer remembered that *as* equal to 2", we implied a zumevaem "The computer recorded in the memory cell is designed ."N hydrochloric for *a* number 2

Now let's see how cells *a* , *b* , *y* will be filled with information during the execution of our program. In a rather IOM beginning of the program they contain yatsya zeros . Operator *a = 10* is executed first . According to the just given Mr. Nome definition of the operator to assign to a Nia in a cell *a* will written the number 10. Then, the operator *b = 6* and a cell *b* appears six. Then, the camera: a torus *y = a + b + 1* . The computer looks at what is in the cells *a* and *b* , there spots 10 and 6 at d forcing them into the expression *a + b + 1* , 17 receives and records the cell *y* . Finally perform I etsya op e Rathore *Debug.WriteLine (y + 200)* . The computer looks at cell *y* . , sees 17 there, calculates 17 + 200 and prints 217 in the *Output* window

Schematically, this process can be depicted as follows (time flows from : (top to bottom

WHAT WE SEE IN THE WINDOW *Output*	WHAT IS IN THE PAM I TI CELLS			OPERATOR EXECUTION PROCEDURE
	Cell for y	Cell for b	Cell for a	
	0	0	0	
	0	0	ten	a = 10
	0	6	ten	b = 6
	17	6	ten	y = a + b + 1
217	17	6	ten	Debug.WriteLine (y (+ 200

: Now we can also clarify how the *Debug.WriteLine* operator *works*

If the operator *Debug.WriteLine* there are expressions with variables in e guises, the VB is in memory the values of these quantities, substitutes them in the regex and zheniya, calculates the expression and . displays the result. And nothing new is recorded in the memory

Answer orally, h will occur if n Omen be sometimes op e tors b = 6 ? and y = a + b +1

Interrupt mode. Stepping mode performs e .5.3.2 Nia program

The computer performs yaet programs have with frightening speed. We did not have time to press the button - and the results have already been received. If things were going in so normally on it so the fact of us admiring and is. But suppose that the results have turned out not what we should be . This means that there are bugs in the program. And it's hard to find them . In this situation, the computer speed early and naet annoying. I would like to see the computer worked hesitated n it , and we had time to follow his work. And it would be good sometime during peek receptacle and of the variables in the memory, it would help to find the error. Etc. Mode e ryvaniya for this and designed. Details on interrupts and debug mode, etc. on the grams discussed in Chapter 9 , but now briefly and without much explanation, will focus on .the optionally 's about dimom

Click in our project on the vertical gray bar on the left side of code, etc. of the TIV-line procedure header . The bar will be a black dot, and the entire row will be highlighted Th p nym (see . Fig. 5 . 3). It was created so-called **point and interrupt** (**Breakpoint**). Breakpoints are removed in the same .way as set - with a mouse click

Figure: 5 . 3

Zapus whith those projects in the usual manner (Button *Start* or press

the **F 5**). The project will start running as usual. A form will appear on the screen. Press, as usual, *Button1* . And then you will feel the difference. Instead of completely performed os and show the result, draft stop at the point of interruption , in which this character string ca and zhetsya highlighted in yellow and it AUC w a yellow arrow (Fig. 5 . 4) . So tol s to just starting performed yatsya, our project is " frozen " as long as we have it is not worn out on the link m . "The moment stopped . " The project is said to . have entered what is known as **interruption mode**

Figure: 5 . 4

I wonder what during a stop are the values of the variables in the COMP memory th tera? To find out, put enough without clicking the mouse cursor on an a value of the variable in the text of the procedures are found in the code window - in the tooltip you will see the value of the variable (in the Fig. 5 . 4 you can see that *y = 0*) . Move the mouse over the variables. As it " . should be, while " there are only n u li on the scoreboard

Let's continue the project and not with the *Start* button , as we are used to, but with the **F 11** key on the keyboard. This is a hot button for **the Debug ® the Step Into** (Incidentally, if your profile is called a slender to the Visual Basic 6 , the hot key is *the F8* , rather than *the F11*). Nothing happens tol s to strip backlight jumps to line *a = 10* . Operators *Dim* carried out in advance, . so the band of illumination across the operator *Dim* per e jumped

F 11 . VB performs *a = 10* , and the following statement is highlighted.

. Check us w hell , which is now equal to *a* , *b* , *y* in memory

F 11 . VB performs *b = 6* , and the following statement is highlighted.

. Check what *a* , *b* , *y* are now equal to in memory

So, the rule is simple - when you press the F 11 VB performs regular operas and programs torus and highlights the operator, which .is to be performed n nym

Note that the VB mode has changed . We know as long as two directed and ma: design mode [*design*] and mode [*the run*] . Now you see the word **[break]** in the header of the main VB window . This means that VB is now in the direct mode e ryvaniya . We can say that the interrupt mode - this mode of existence of the project, when the ful l nenie program is interrupted, but not finished . When you press **F 11** , you order BASIK to stop at each operator. This kind p e benching pr e ryvaniya called **step by step mode** . .Interrupt mode can be customized and more rarely stay in ki

F 11 . VB serves *y = a + b + 1* , and the next operator is highlighted. . Check, which is now equal to *a* , *b* , *y* in the memory I tee

F 11 . VB performs *Debug.WriteLine (y + 200)* , and the next operator . illuminant t smiling. The number 217 appears in the *Output* window

F 11 . The light goes off because the procedure is complete. You can . press *Button1 again*

So, the computer did all the things that would have made when you .press *Etpu Start* , but only IU d lennom pace

At any moment you step through the program, you can instead of the **F11** push *Etpu Start* , which on this occasion is renamed **the Continue** , and .so on gram will continue to run as usual

When VB is in the interrupt mode, the screen you see the Tools panel n .(comrade for debugging - *the Debug* (Fig. 5 . 5

Figure: 5 . five

When you see fit to finish debugging, you can do it the usual FPIC on a bong, covering a cross shape, and can and press *the Stop Debu g ging* panel *the Debug* (it has the form of a black square). If the panel *Debug* not ve d but . put it in there in the usual way: **View Take ® for Too l bars ® Debug**

Note : When debugging in step-by-step mode, you often want to see the main VS window and the launched form on the screen at the same time .

But when you press the **F 11** fo r ma often disappears from view, being closed and the other windows. To prevent this from happening and lo, even before starting debugging, reduce the main VS window to a part of the screen. Then, after ZAPU with ka debugging drag the shape to the vacant portion of the screen. Then the VS window will not overlap the form and the .problem will disappear

Operator assignment changes the value per e .5.3.3 variables values

So far I have not considered programs in which variables change their : value. Now it's time a prospect of gram consider

```
Private Sub Button1_Click ( ByVal sender As System.Object, ByVal e As
System.EventArgs) Handles Button1.Click
    Dim k As Integer
    k = 10
    (Debug.WriteLine (k
    k = 25
    (Debug.WriteLine (k
    k = 4
    (Debug.WriteLine (k
End Sub
```

As the program runs, the contents of cell *k* changes. Thus, when the operator *k = 25,* there im e hundred value 10 appears 25. And where does a dozen? She erased, meaning that the PC Zab 's Vaeth her forever. There is a :general principle of all to m Pewter

If you like th someday cell or even in some memory location or di with ka recorded new information, the old information recorded there wounds s Chez automatically erased, ie even with whether it is someone .and need

Just now instead of 10 in the cell *k* is 25, then the operator *Debug.WriteLine (k)* is printed and melts already 25. (The word " print " out of date, it remains from the time when the Mon and tori was not there, but instead they were Printers . When I was in the old saying that Mr. formation print, I mean that it appears on the screen in a particular window.) Sl e blowing the operator *k = 4* will record the place of 25 four and

.Debug.WriteLine (k) it napechat and et. Etc. on trust

Let us write schematically the process of changing information in cell *k*

:

WHAT WE SEE IN THE *Output* WINDOW	WHAT IS IN THE MEMORY LOCATION k	OPERATOR EXECUTION PROCEDURE
	ten	k = 10
ten	ten	Debug.WriteLine ((k
	25	k = 25
25	25	Debug.WriteLine ((k
	4	k = 4
4	4	Debug.WriteLine ((k

?What will the next program print

```
Private Sub Button2_Click ( ByVal sender As System.Object, ByVal e As
System.EventArgs) Handles Button2.Click
    Dim f As Integer
    f = 30
    (Debug.WriteLine (f
    f = f + 4
    (Debug.WriteLine (f
End Sub
```

The operator *f = 30* will write the number 30 in cell *f* . Then the operator *Debug.WriteLine (f) will* print 34. What will the strange operator *f = f + 4* do? Don't think this is an equation or anything like that. This is an assignment operator, and thus he executed a publicly definition of an assignment statement . With the start VB calculates the right side of *f + 4* at d puts there instead *f* its value is taken from the cell and the receive 34. Then, the number 34 is written to the cell that dedicated Move N hydrochloric indicated on the left side, that is, again the into cell *f* . In this case, the old

.value 30 will be erased. Operator *Debug.WriteLine (f)* n and prints 34

Thus, the operator *f = f + 4* merely increases the number in the .cell *f* in Th t Verka or, in other words, to increase and Vaeth *f* 4

Why is this needed? Find out later. Very much even necessary. The operators of this type of Br e chayut so often that even invented a shorthand :notation for them. Practice at emsya

Act	Condensed s and pis	Operator
The variable is increased by 1	a + = 1	a = a + 1
The variable decreases by 25	s - = 25	s = s - 25
The variable is doubled	b * = 2	b = b * 2
The variable is reduced by 3 times	t / = 3	t = t / 3

Determine without using a computer that will be printed when the ff e :blowing fragments of programs

- *(* a = 100: a = 10 * a + 1: *Debug.WriteLine (a*
- *(* a = 100: a = -a: *Debug.WriteLine (a*
- a = 10: b = 25: a = ba: b = ab: *Debug.WriteLine (a)* : *(Debug.WriteLine (b*

Variable names . By keyword of VB .5.3.4

We have already discussed how to properly name VB elements . Repeat : y

Name may consist of letters, numbers, and underscore characters and Niya, and not should begin with numbers . And it .should not consist only of handwriting and Nij

True, in VB the name is defined somewhat more broadly, but the rule I .gave can be considered good advice

In school claim AC line value m s letters used denote the (*a* , *s* , *d* ...

). Bol s shinstvo programming languages, including the VB , lets you designate Move n nye names of many letters . Here are two equivalent Fra g :ment of the program

;Summa = 3	;a = 3
;SHURA = 4 + Summa	;b = 4 + a
Debug.WriteLine (Summa, (SHURA	(Debug.WriteLine (a , b

In either case, the numbers 3 and 7 will be printed . Obviously, the computer s serial ports do not care what their names we denote variables or other elements in the meaning of names he did not ponder and is not surprised that the variable *Summa* no su m of mine is not so smiling, but .simply the number 3

:Examples of correct spelling of names

a
s 25
oshibka
polnaja_ S um mm ma
1_
tri_plus_dva __ a1b88qq QQQQQQQQQQQQ _____
It _ not _ name _ I g _ trust _

:Examples of incorrect spelling of names

contains characters (space), not a letter, qi £ swarm or sign subordinated p nodding	**polnaja summa**
starts with a number	**1**
starts with a number	**8as**
It contains the & non-boo to howl, number, or by e by deleting	**Domby & Son**
contains characters (space), not a letter, qi £ swarm or sign handwriting and van Ia	**That's the right name yes yes yes**

VB ignores the difference in names between uppercase and lowercase boo to you. So, for him **S umma** and **sU mmA** are the same name. But he Prisma t Riva to ensure that in the code window that particular it I was on was written about always the same, namely so as to embrace in the county of

(the truth, I'm not sure what he looks for all the Russian letters , in the Sun I .(lump case the Visual Basic 6.0 is not always Prisma t Rival

Key words . Keywords (*Keywords*) are service words of VB , which he widely uses in the program text. We already know about us with some of them: *the Dim* , *an Integer* , *Double Room* , *Sub* , etc. The table below p. E . rechisleny all cells th chevye words of the VB

And	Alias	AddressOf	AddHandler
Assembly	As	Ansi	AndAlso
Byte	ByRef	Boolean	Auto
Catch	Case	Call	ByVal
CDate	CChar	CByte	CBool
CInt	Char	CDbl	CDec
Const	CObj	CLng	Class
CType	CStr	CSng	CShort
Default	Declare	Decimal	Date
Do	DirectCast	Dim	Delegate
ElseIf	Else	Each	Double
Error	Erase	Enum	End
False	ExternalSource#	Exit	Event
Function	Friend	For	Finally
Handles	GoTo	GetType	Get
In	Imports	Implements	If
Is	Interface	Integer	Inherits
Long	Like	Lib	Let
Module	Mod	Me	Loop
MyClass	MyBase	MustOverride	MustInherit
Not	Next	New	Namespace
Object	NotOverridable	NotInheritable	Nothing
Or	Optional	Option	On
Overrides	Overridable	Overloads	OrElse
Property	Private	Preserve	ParamArray
ReadOnly	RaiseEvent	Public	Protected
RemoveHandler	REM	Region#	ReDim

Set	Select	Return	Resume
Single	Short	Shared	Shadows
String	Stop	Step	Static
Then	SyncLock	Sub	Structure
Try	True	To	Throw
Variant	Until	Unicode	TypeOf
WithEvents	With	While	When
ExternalSource#	Const#	Xor	WriteOnly
&	-	Region#	If ... Then ...# # Else
/	= *	*	= &
^	= \	\	= /
=	= +	+	= ^
			= -

Obviously, you invented names do not match the keyword the VB , otherwise in of niknet confusion. VB tries scrutinizing be behind it and can point you to an error if you accidentally use the keyword as the name. If you : still need to do this, put the name in square brackets, for example , like this

Dim [Sub] As Integer
Sub] = 99]

I also do not recommend the use of at first as names for CREATE and Vai your variable names popular obe such as are for their properties, methods, and others. N in April and measures, if you name a variable name of *the Button1* , *the BackColor* or *the WriteLine* , then the VB , m of Jette, .and cn have thawed, but to confuse you

We call it in Russian . Take a look at the text of any program. It consists of Ba in the prefecture of kl th chevyh words in English (in Latin) and names. Moreover , there are so many of those and others that the eyes run up and it is not clear where the keyword is and where the name is . And if it's the name - with the s given whether it is a software or taken from the .NET Framework class library - just do not understand . We, the Russian-speaking, is more convenient : we can give names to py with skom (Cyrillic). Ra of Nice between the British and Russian letters immediately catches the eye and deal with the text of the program becomes easier as an example: all that we have created ourselves, called in Russian, and all that is inherited from the

.VB - in English. So I e e barking

Numeric variables. Mathematics . .5.4 Accuracy

Simple arithmetic is better to do on a calculator than to m pewter e, but .complicated - on the contrary

Mathematical actions and functions .5.4.1

In this under the section, you will learn how to enter into the computer the complex forms in ly. If you - the student is not the most senior classes, not all mathematical functions that are mentioned , a bud from so you know . Do not worry, with further reading ting on -understand things you do not .hope for byatsya

:**Arithmetic operations are** denoted in VB as follows

MEANING	RESULT	DE J STVIE
a plus	five	2 + 3
m and nous	3	4 - 1
multiply	6	2 * 3
to split	five	2 / 10
integer division (17 divided by 5, it (turns out 3, with a remainder of 2	3	5 \ 17
remainder of the division	2	17 **M od** 5
remainder of the division	7.2	37.2 **M od** 10
two cubed) - exponentiation) 3^2	8	3 ^ 2

In math lessons, we used to write *ab + cd* , meaning: " *a* y m multiply by *b* plus *c* multiplied by *d* " . In VB , we must write this expression like this: *a * b + c * d* . Otherwise the computer will think that you need a variable that has the name of *the ab* , add per e mennuyu having the name of the *cd* . To avoid ambiguity, the multiplication sign is necessary to write entirely e GDSs , including before the parentheses . For example, *a * (b + c)* . *(*instead of *a (b + c*

Brackets . Because the keyboard is necessary to enter all the information character by character in one line, two-input x storey expressions such as

difficult. For this reason , a forward slash has been selected to denote a division in programming . The given expression on VB should be written as follows: *a + 1) / (b + 1)* . If we do not put brackets, the expression would be obtained in such *a + 1 / b + 1* , which is N e is correct, as the computer as we always before addition and subtraction performs Multiply e of and division, so in the latter case not knowing our intentions, he would first . section e lil 1 to *b* , and then the result etc. and Basville to *a* and 1

Remember that the computer performs arithmetic operations in the usual manner for us: first, to evaluate the function, raises the power, remove the roots, then ful l nyaet multiplication and division, then addition with .subtraction

Question: when can parentheses be used in expressions? A: always, when you fuss and mess doubt correct Ocher d Nosta action. An extra pair of : parentheses won't hurt. Example: write an expression in VB

:Let's write it like this for now

*a / (2 + a * b)) / (3 + a) * b + 1)*

Understand this entry. With false expression with a large number of brackets in the eye are perceived with difficulty, since it is difficult for a specific brackets see her Zuko N hydrochloric couple. In this case, I can advise to go " from small to large " , ie removed and chala notice the smallest

of the bracketed Phragmen n you expression . We have it *(3 + a)* and *(2 + a * b)* . Notice their brackets with your eyes . After this, you will already notice better brace for bo Lee large fragment of s , t and the FIR as the *(1 + a ./ (2 * a b +))* , etc

VB comes to your aid , and at the time of entering the next brackets .identifies the floor in bold sri p that her and her pair

Understood? The above post I was not satisfied, because we do not know that VB will do before - e e pour *(1 + a / (2 + a * b))* to *(3 + a)* or multiply *(3 + a)* to *b* ... And the result depends on it. Let's add a couple of :parentheses to be sure

*a / (2 + a * b)) / (3 + a)) * b + 1))*

.Now everything is all right

Point or comma in decimals s? Almost all programming languages and even, to a course, in the VB , in the software code in the text box, made in decimals instead of a comma art and to reach an end. Example: *y = 62 . 8* - sixty two point eight. However, if you remember, p e results of in a text box VB displays a semicolon. What's the matter? The VS , being the App e Niemi the Windows , partially receives from it a habit to use a comma in Russia. Os on the fight the problem is not here. In the vast majority of cases , what I have already said will help you make the right choice. In other cl in teas Apply yte method of " scientific spear " - use , for example, point, and if VS complains or starts d f lat that something is not right, change it to .the app I thuja

Mathematical functions . Additionally n ie how many arithmetic operations VB can perform other mathematical operations, for example, extract the f of the square root . However, there is no icon key on the computer , so VS has a special function - *Sqr t* . For example, the root 25 of Symbol and chaetsya so - *Sqr t (25)* , the root of *a + b* so - *Sqr t (a + b)* . Here *Sqr t* - cut from an r liyskogo expressions *Square then root* - square root which . The argument, that is, so on, from what well Well but remove the

.root , is written in brackets

Here is a partial list of mathematical functions the VB : Almost all of . them are m e todami class of *the System* . **Math**

MEANING	RESULT	FUNCTION
A is the absolute	8	Math. **Abs**

value (modulus) of a number		((-8
By Oren square	five	Math. **Sqrt** ((25
About rounding to the nearest whole	18	Math. **Round** ((17.952
	17	Math. **Round** ((17.48
About rounding to 2 decimal places	51.24	Math. **Round** (51.23708, (2
P otolok" - b " lizhayshee integer , bol shee or equal to argument	46	Math. **Ceiling** ((45.23
	5 4 -	Math. **Ceiling** (- (45.23
P of l ' - b " lizhayshee integer , changed shee or equal to the argument	8	Math. **Floor** ((8.92
	nine-	Math. **Floor** ((- 8.92
C elaya part number (the fractional part (is discarded	nine	(**Fix** (9.47
	nine-	(**Fix** (-9.47
Sign for all positive numbers is 1	1	Math. **Sign** ((300
Sign for zero is 0	0	Math. **Sign** ((0

Sign for all negative numbers is -1	1-	Math. **Sign** ((-480
The maximum of two numbers	44	Math. **Max** ((29, 44
The minimum of two numbers	29	Math. **Min** ((29, 44
Number π	3.14159265358979	Math. **PI**
Number e - base of natural logarithm	2.71828182845905	Math. **E**
e^{2} - number e to a given power	7.38905609893065	Math. **Exp** ((2
Natural logarithm for *l n 35*	3.55534806148941	Math. **Log** ((35
Decimal logarithm for $log_{10} 1000$	3	Math. **Log10** ((1000
A random number from the range (0 - (1	0.7055475	**Rnd**

Besides this , there are function *Sin* , *Cos* , *Tan* (that is tangent), . *Asin* (that means arcsine), *Acos* , *At a n* and some other prob . Working with random variables is described in 7.3.1

Test the math functions you are interested in by writing a program like :this

```
Imports System.Diagnostics.Debug, System.Math
Public Class Form1
    Inherits System.Windows.Forms.Form
    Windows Form Designer generated code
Private Sub Button1_Click ( ByVal sender As System.Object, ByVal e As
System.EventArgs) Handles Button1.Click
        (WriteLine (PI
    ((WriteLine (Round (17.48
    End Sub
End Class
```

: Check in your mind

Expression **(2 + 1) ^ 2** when calculated will give **9** Expression **1+ (2+ 8) ^ 3** when calculated will give **1 001** Expression **1+ Abs (5-8)** when calculated will give **4** Expression **2 ^ 4 + Sqr t (35+ 1)** when calculating give **22** expression **Sqr t (for 8+ Floor (41. 8))** when calculating give **7** expression **21 \ (Round (2.54 +1))** gives the calculation **5**

: Determine orally, without a computer, what the procedure will print

```
Private Sub Button2_Click ( ByVal sender As System.Object, ByVal e As
                                    System.EventArgs) Handles Button2.Click
                                                Dim a, b As Double
                                    a = (2 ^ 2 + 1) * (20 - (2 ^ 2) ^ 2) - 11
                                ((b = Round ((a + 2) / 10) * (11 \ (a - 4
                                        (WriteLine (a ^ 2 + b - 1
                                                End Sub
```

Is it clear to you that here you need to round *((a + 2) / 10)* and not *((a + 2) / 10) * (11 \ (a-4))* ? If it is not clear, then carefully look for its pair for . *(*each parenthesis, starting with the fragment *(a + 2*

Numeric l iterals . The word "number" is too broad. Therefore, it is advisable to call specific numbers in the program text in the code window sometimes differently. We will call them **numeric literals** . So, in the line

*y = 20 * a - Round (-2.54) + 0.01*

there are three numeric literals: *20* , *-2.54* , *0.01* . One of them is a two - draw b GOVERNMENTAL. Literals has traditionally called constants, but .this term in VB n e as another value

In the future, I will expand the concept of a literal to non-numeric data . types

Numeric data types .5.4.2

Data - that's over what the program works to get p e result. A **program** is an instruction on what to do with the data in order to get the result. So, in our project Calculator, the data was the numbers we entered in the two upper text boxes, and the result was the number in the lower text box. The project data were player audio and video files on the disk, and p e result - sound .through the headphones and from about mapping screen

Data often acts as the values of variables. Therefore, we have the right to talk about data types in the same way as we talk about variable types. This is the same t and nN. Literals , which we write in the program - the same data, since the program is a servant of the melt and on them too. Place the mouse cursor on a pleasure th literal text program s and you will see and they hint to the name of the data type to which , in the opinion of the VB , apply it so literal . Therefore, in the future, I will not distinguish between data types . and types of variables

Briefly and without explanation, VB data types are listed in 5.2.3 . In .this subsection , we will only talk about the types of h and word data

In VB has 7 numeric types. In the table they are listed and explained in detail e Nia - in the following sections . For reference, I gave information on .simple non-numerical t and memory

Range of values of type	Size yache th ki (in the ba th (minute	Description	A type
O t 0 to 255	1	Integer - polozhitel s Noah	**Byte**
About t -32768 to 32767	2	Integer - to about Rothko	**Short**
About t -2147483648 2147483647	4	Integer - Oba h Noah	**Integer**
About t -9,223,372,036,854,775,808 to 9223372036854775807	8	Integer - length Mr. Noe	**Long**

From ± 3.4028235 * 10 $^{+38}$ to ± 1.401298 * 10 $^{-45}$. The accuracy is approximately 7 significant digits.	4	D esyatichnaya fraction NORMAL second fine of STI	**Single**
From ± 1.79769313486231570 * 10 $^{+308}$ to ± 4.94065645841246544 * 10 $^{-324}$. The accuracy is about 17 significant digits.	8	D esyatichnaya double fine fraction of the STI	**Double**
Number 29 digits in length, and ten hours Nye point can be placed in any IU with those. : With as my big number is ± 79228162514264337593543950335 , :and on and my little - is ± 0.00000000000000000000000001	sixteen	A esyatichnaya fraction long , absolutes m hydrochloric accuracy	**Dec i mal**
Just two meanings: true - false	2	Boolean type	**Boolean**
Any of 65536 characters, letters, numbers and other characters on the fishing	2	Symbol	**Char**
Arbitrariness s ny text	*	Line	**String**
Any date from the birth of Christ until 9999 . A nd fight time.	8	Date and time	**Date**

cell size for a string variable depends on the line size and from to m - * .pewter

An Integer , Long , Short , a Byte - c e ite .5.4.3 number

:Create a project with a button and enter a program like this

```
Private Sub Button2_Click ( ByVal sender As System.Object, ByVal e As
System.EventArgs) Handles Button2.Click
    Dim a, b, c As Integer
    a = 201234567
    b = 201234567
    c = a + b
    (WriteLine (c
End Sub
```

Each of the variables *a* and *b* has a value somewhere around 200 :million. The program works fine. Here's the result

402469134

Let's see how large numbers our program can perceive. D on bavim : couple of digits to znach e NIJ *a*

a = 20123456789

VB emphasizes the literal and produces a pop-up tooltip error, in which ?we see the word of *an Integer* . What's the matter

As you can see from the table , a part of the *Integer* type occupies 4 bytes in memory . Hence, under the n th belt *and* and *b* computer assigns a memory cell of 4 bytes each. 4 bytes - it n e bol s sho th memory and it can fit into an integer is not too large , and it is , as we see in the table - in the range of plus or minus two billion slightly . We, having added two figures exceeded this number, in of VB and ck and room to us what type *Integer* not by rank to . be so large

:Let's set *a* and *b* values to two billion

```
a = 2012345678
b = 2012345678
```

VB does not mind, but when we run the project and click on the button gives with a communication error, in which m we see the word of *the* ."*overflow* , which means "overflowing

The fact that the summing , subtracting or multiplying numbers such as *an Integer* , the VB prismatr and INDICATES to fit in the result type of *an Integer* . After d va plus two billion Millia district shall be four billion, and . that If w whom much for *an Integer*

What if we need more ? In this case, n ashi variables must- s to be calculated for a longer s num well . To variable had the right to take the values of very large n e mated numbers, it should be about byavle at not as *an*

Integer , but as **Long** (long integer). Under type variable *Long* computer as we and dim table, assigns memory in 8 -byte s and so it can take the values h . and sat length of 19 digits

Declare all three variables as *Long* . Then everything will be all right.
You may say: this is illogical, it was enough to declare one variable *c as Long* ! Try and claim a look whether the program is working. Does not work. *Overflow* again ! VB thinks it's safer this way. Strictly, but nothing to do. Continue to declare all variables uch as stvuyuschie in the calculations, the most spacious type of those that may be needed for the results of calculating . th

Why is *Integer* needed if there is a *Long* ? Well, at least to ehkon about Mit memory. For reasons of economy, you can use both **Short** and **Byte** . (types (see table

The population of Moscow is a = 9,000,000 inhabitants. The population of Vasyukov is b = 1000 inhabitants. All of Moscow moved to Vasyuki. How many inhabitants are there? Execu s zuyte variables - first type *Short* , and . when it will not work - *an Integer*

Single and Double Room - decimal fractions .5.4.4

With the type of fractional numbers *Double* we met ilis in 5.2.3 . To acquaint m camping closer to him and to the type **Single** . Both are designed to work with entire core and b quantum numbers and differ according to the .table, accuracy and range of values

:Create a button in and enter the following program

```
Private Sub Button3_Click ( ByVal sender As System.Object, ByVal e As
System.EventArgs) Handles Button3.Click
    Dim a, b, c As Single
    a = 23.456
    b = 100
    c = a + b
(WriteLine (a): WriteLine (b): WriteLine (c

    Dim d, f, g As Double
    d = 123456789
```

```
f = 0.000555
g = d + f
(WriteLine (d): WriteLine (f): WriteLine (g
End Sub
```

Note that the *Dim d, f, g As Double* statement is not at the very top of the procedure. It's nothing. It is only necessary to advert variable was located ".in the procedure before it "as used e Nia

:Run the project. Here's the result

```
23,456
one hundred
123,456
123456789
0.000555
123456789,000555
```

Everything is correct. Let us now try to check the type *Single* strength, we prov e ryali type of *an Integer* . Try to write in m esto *a = 23.456* a very :long shot

a = 23.456789012345678901234567 89

But VB doesn't let us live in luxury. Right in the code window, he :shortens the line to

```
a = 23.456789012345681
```

and even slightly distorts the number of. Put your mouse over it so literal and have seen podska of Ku « *Double Room* ». Why? After all, it is variable nd *a* type is *Single* ! The fact is that VB , while the project is not running, does not bother guessing the meaning of literals in the code window and “cuts all fractional literals under *Double* ”. Let's start the project. Here are the first : three numbers p e result s

```
23.45679
one hundred
123.4568
```

As you can see, VB has truncated our long numbers to 7 significant digits. He did it to claim on the fact that the 4 bytes reserved for the memory cell to a variable of type Single , unable to accommodate more. If we want to have greater accuracy b, then of ads I eat our variables kind om *Double Room* . Under variable type *Double* Comp w ter assigns memory in 8 -byte s . and therefore it can be g of Sec longer than - 1 7 significant digits

:Let's try to set a very small number. Instead of *f = 0.000555,* we write

f =

)000000000000000000000000000000000000000000000000000543
But VB does not like in the fractional number greater than the number of zeros and now , even though the record and so forth and the villa, and quite tough type *Double Room* , VB directly in the code shortens the page of :Ku to

```
f = 5.43E-62
```

This is also a correct entry, and what it means is explained a little later, in . 5.4.7

Let's talk about the accuracy of the calculations. If you still think that the computer performs all calculations absolutely accurately, then you are mistaken. The computer is just very accurate. Very , not absolutely . And you will soon be convinced of this. Let's start with the fact that VB doesn't know how to work with fractions. He does not know what is 1/3, the poet of the mu us forth and hoditsya ask the computer instead of 1/3 of the number of 0.333333333333. H of this is not an exact fraction is 1/3, but only an approximate value. Make it accurate VALUE and eat the number of decimal places must be infinite, and this number does not mind e Stith in the computer's memory. Consequently, even in theory, the computer can not be .absolute about lutely accurate

Secondly, due to the limitation of the *Double* 17 type to significant digits, operations even on very precise numbers are not performed very : ? accurately. What, for example, a print Fra g cop

```
Dim d , f , g As Double
d = 123456789
f = 0.00987654321098765
g = d + f
(WriteLine (d): WriteLine (f): WriteLine (g
```

If you think the exact amount is *123456789.00987654321098765* , you are wrong , because such a long number will not fit in the *Double* type . And :it will be printed from that

```
123456789
0.00987654321098765
123456789,009877
```

.Ten digits from the exact value of the sum are truncated

Integer literals VB automatically assigns to the type of *an Integer* , and if they are too length n nye for *an Integer* - a type *Long* . VB classifies fractional literals as *Double* . All this he does not particularly paying attention

to what is variable s in the same op e Rathore e have others in the Gentile .type

Whole numbers or decimals? Numeric .5.4.5 Decimal

You may ask: why use integer types *Integer* and *Long* , if types ten I - particle fractions *Single* and *Double* allow us to work like a whole , so with fractional numbers? The point here is the presence or absence of the absolute accuracy of the calculations. N When using a type of decimals is quite conceivable situation where two plus two is not exactly 4, and, say, 4.00000000000381. This is due to the peculiarities Representat in Lenia decimals in the computer ah. In most real-world problems such Malin s kaya error is negligible, but there are tasks where precision is needed abs on the lute . When using integer types, VB makes sure that all numbers and results are perfectly accurate integers. When adding subtracting and Britain and multiplying it, you know, not a problem, but by dividing the computer at the .prospect and hoditsya okra g lyat

Tip : If you are sure that it is always the variable must be an integer, and the nickname of the GDSs fraction, declare it an integer type, if there is little .doubt - draw b nym

However, absolute precision is also needed when working with fractional numbers. H as an example, in the financial calculations of dollars or rubles - this is a whole part of a decimal, and cents or pennies - hundredths of this fraction . You can't make a mistake here. The *Decimal* type is ideal for this situation . Although this type has a fractional part, it performs addition and subtraction of fractional numbers absolutely exactly. Number znach and . (boiling figures had a record (see. That b face in 5.4.2

. Let's try now to check the operation of the *Decimal* type

```
Private Sub Button1_Click ( ByVal sender As System.Object, ByVal e As
System.EventArgs) Handles Button1.Click
    Dim a, b, c As Decimal
    a = 1234567890123456789
    b = 0.0098765432109
    c = a + b
    (WriteLine (a): WriteLine (b): WriteLine (c
End Sub
```

:Run the project. Here's the result

1234567890123456789
0.0098765432109
12345678901234567890,0098765432

As you can see, VB , as expected, trimmed the sum to 29 significant .digits

Type symbols . Note that the value per e Mann fired *a* and *b* , I gave not long, type *Decimal* can swallow and much longer. P about trying to add numbers to the literal is variable oh *well* . VB emphasizes the literal and prompts an error in the cat about the swarm, we see the word of *the overflow* . What's the matter? Let's shorten the literal again , leave the mouse cursor on it and see the " *Long* " prompt . Poch f th? After all, is variable th *a* has type *a Decimal* ! The fact is that VB does not want to classify literals to the *Decimal* type without an explicit order from people about the century . And order so : oh - letter **D** at the end of L and tera

```
a = 12345678901234567890123456789 D
```

Now everything is all right. The letter referred to **the symbol type** (*the* . (*Literal of the type character*

If you want to force VB to assign a literal to a specific numeric m and :ny, then here are the corresponding type symbols

Decimal	Double	Single	Long	Integer	Short
D	**R**	**F**	**L**	**I**	**S**

Type conversion .5.4.6

Difficulties . So we see that when it comes to types, the VB shows nekot about Rui etc. and dirchivost . N For example, as we have just seen, it does not permit literals without special characters to have any type. Or it requires that the result of addition , subtraction, and subtraction , or multiplication of numbers of a certain integer type fit into the same type . For example , and measures , after the start-up procedures s

```
Private Sub Button1_Click ( ByVal sender As System.Object, ByVal e As
System.EventArgs) Handles Button1.Click
    Dim a As Byte = 200
    Dim b As Byte = 100
    (Debug.WriteLine (a - b
    (Debug.WriteLine (a + b
```

```
End Sub
```

will overflow error is issued , as 200 + 100 to 255 - limiting values e Nia type . *a Byte*

You will come across as a picky features that require arguments themselves objectified ie certain type and complaining if they have a different . type

In all these cases, you will be faced with the question: to yield to VB requirements or not to yield. It would seem: what is easier! - give in and that's it. For example, I was advised to prevent overflow to declare all the variables involved in a calculated and s, most spacious type of those that may be .required for calculating the results of e Nij . And that's good advice

Another tip: if all else fails, give the type of ads in lane e variables that cause quibbles , resigned to the negative effects of the type of uncertainty. . That is, just write *Dim a*

And it's not hard to agree with a picky function, too, and declare a . variable the way it requires

But not always, these concessions come in th t programs istu convenient s . We still face a to n indiscrete and cases I m and so th inconvenience . And so now I will show you how to cd f lat so that the wolves were fed and the .sheep p e ly

Type conversion functions . There are special features, the purpose of which - to transform its args from cop to a certain type. For example, a function I **CInt** transformation as a type of its argument to *an Integer* . Therefore, orator

```
((Debug.WriteLine ( CInt (2.4 + 10
```

.will print 12, not 12.4

Suppose that in the previous example, about 200 + 100 We do not want to change the type is variable 's *Byte* at a bigger *Short* . How can then get rid of the overflow? Apply **CShort** - function th conversion in its argument type : *Short* . Like this

```
Private Sub Button1_Click ( ByVal sender As System.Object, ByVal e As
System.EventArgs) Handles Button1.Click
    Dim a As Byte = 200
    Dim b As Byte = 100
    (Debug.WriteLine (a - b
```

```
((Debug.WriteLine ( CShort (a) + CShort (b
End Sub
```

Here, a function I *CShort* converts before addition value is variable 's *a* and *b* of the type *Byte* in type *Short* . Not is variable s, and their values, and not everywhere, but only in the ME with those. Sami is variable s as they were, and have remained such as *a Byte* , and their meanings in other IU with . max programs s (where subtraction) also remained the type of *a Byte*

The transformation took place, and therefore the addition performed yalos on numbers such as *Short* and the result (300) bound was the mind .schat sya it is in this type, which he freely and fit L Xia

Here is a table of type conversion functions, where I have included : conversion functions to non-numeric types for reference

Converts to type	Function
Byte	CByte
Short	CShort
Integer	CInt
Long	CLng
Single	CSng
Double	CDbl
Decimal	CDec
Char	CChar
String	CStr
Boolean	CBool
Date	CDate
Object	CObj

Implicit type conversion . When performed enii arithmetic operations (in other cases) VB often unnoticed by us and very converts the number of one type, etc. have goy. For example, by calculating the expression *5 + 2 .8* , VB converts the number 5 of the type *Integer* in type . *Double* , and then add the numbers p and receives e result *7 .8* type *Double*

,At all

performing arithmetic operations on numbers of different types, the

VB Conv and zuet them to a common type that would could accommodate the types of the two numbers , then prismatr and . INDICATES to RESULT and so , too, fit in this type

:Here is a list of numeric types in ascending order of capacity

Byte, Short, Integer, Long, Single, Double, Decimal

Example: on execution, snippet a

```
Dim a As Short = 32700
Dim b As Byte = 100
( Debug . WriteLine ( a + b
```

VB converts the number 100 in the type *Short* , and then will give the overflow, because the result of addition does not mind e schaetsya in type . *Short*

When dividing the integer types VB converts the dividend and the divisor in about a novnom the type *Double* and the result is the same . type *Double*

For example , by dividing 2 by 3 VB sees that the result does not get the whole, the poet of th he first converts both integers in type *Double* and the .result gets the same type

There are certain types of implicit conversion rules, in which we will .not detail hp and vatsya

The type of arithmetic expression . Let VB Comput with by the expression *a + b * c* . The question is, what type will the evaluated expression have? Ra with is discussed in the chain. First VB multiplies *b * c* and the type of product objectified e wish to set up on just mentioned about th rule. Then, already having a product and knowing its type, the VB adds it to *a* ., determining the type of the amount again on the same prospect as Vila

Thus, the chain performed yaemyh action , the VB determines the type .of the longest of any of the first expression

Formatting numbers .5.4.7

The same values may look different. For example, in a school notebook you can write down the same date like this - *12/25/03* and so - *December 25, 2003* . You can write the same number like this - *500* and so - $5 * 10^2$. As for the VB , it selects the external view of the data, on the basis of his findings, which are not always PPSR and give to our desires. In this case, it is

necessary to strictly specify to m pewter have, in any form (**format**) we .want litsezr et a particular receptacle and chenie

: Take a look at a program like this

```
Private Sub Button1_Click ( ByVal sender As System.Object, ByVal e As
System.EventArgs) Handles Button1.Click
Dim a As Double = 12300000000000000000
(WriteLine (a
End Sub
```

:Run the project. Here's the result

```
1.23E + 18
```

What does it mean? This means that VB does not like too long numbers and represented l I is them you not familiar to the average person , and the so-called **exponential s nom** or **scientific forms and ones** . Number *1,23E + 18* is the same as the number *12300000000000000000* , only s looking the other way. The construction *E + 18* simply means multiplying by 10^{18}. Thus, *1,23E + 18* ozn and chaet 1.23×10^{18}. In another way, you need to move the . comma to the 18 positions the first right - and you get Normal s Noe number

: Now take a look at this program

```
Private Sub Button2_Click ( ByVal sender As System.Object, ByVal e As
System.EventArgs) Handles Button2.Click
Dim a As Double = 0.00000000000654
(WriteLine (a
End Sub
```

:Result

```
6.54E-12
```

The construction *E -12* simply means multiplying by 10^{-12} or, what is the same, dividing by 10^{12}. So *6.54E-12* means $6.54 * 10^{-12}$. In another way, you need to move the app I thuja 1 2 position in the le in - and you get but p .minimum number

If, after these explanations, you still did not like the exponential format, you can order a computer from him to you not tired of them , and show the results of a human being. For this purpose, the operator *WriteLine (a)* need to use the **function fo r matting** , that is, the appearance of the control. This function is called **Format** . To be specific, let us take the first of our two programs , one in which prisuts t exists large integer chi with Luo. Here you need to write *Format (a, "#")* instead of *a* . It will turn out

((("#" ,WriteLine (Format (a

Symbol # VNU t ri quotes means that you want to see the number in the usual form without fractional second parts and . That's the same result in a : (new format (made of trust

1230000000000000000

And now analyze the most popular function capability *Format* of the format and tion numbers (a format it can , by the way, data and other types). : Opportunities for STI , these illustrates the following program

```
Private Sub Button3_Click ( ByVal sender As System.Object, ByVal e As
System.EventArgs) Handles Button3.Click
    Dim a As Double = 12345 . 67890123
    Dim b As Double = -0 . 0729
    ((("#" ,WriteLine (Format (a
    ((("WriteLine (Format (a, "0
    ((("### . #" ,WriteLine (Format (a
    ((("WriteLine (Format (a, "0 . 000
    ((("########## . ########" ,WriteLine (Format (a
    ((("WriteLine (Format (a, "00000000.00000000000
    ((("########## . ########" ,WriteLine (Format (b
    ((("WriteLine (Format (b, "00000000.00000000000
    ((("########## . WriteLine (Format (b, "0
    ((("WriteLine (Format (b, "P
    ((("WriteLine (Format (a, "E
    ((("WriteLine (Format (a, "C
WriteLine ( Format ( a , " F ### Y ### H ### K ### A #### . # K ### A
    ((" ### W ### T ## A # H # K # A
End Sub
```

:Run the project. Here are the results

```
12346
12346
12345,679
12345,679
12345,67890123
00012345,67890123000
0729 ,-
00000000.07290000000-
0.0729-
7.29%-
1.234568E + 004
RUR 345.68 12
```

Let us explain what we saw. In brackets function *Format* arranged through the app I thuja two arguments. The first argument - that is what we are formatting - variable chi with lo expression. The second argument - a quoted string, with which we control and I eat the appearance of. Characters in quotes is called **a symbol s format** and **specifier s format** (we will not go into details, explaining what some of them). Here is an explanation of dis t Via these characters (the full meaning of the symbols # and *0* becomes clear : (after reading a few lines of the one used Litsa

Explanation	Symbols and p Performan s
You want to see the number in the usual way , and no fractional cha with ti	#
	12346
You want to see the number in the usual manner and no fractional cha with ti	0
	12346
You want to see the number in its usual form. After the point you wish to see no more than 3 characters u e barking of the shot it will still be shown in full	### .#
	12345,679
You want to see the number in its usual form. After the decimal point, you want to see exactly 3 signs, the integer part of the fraction will still be shown in full	0.000
	12345,679
After the point you wish to see no more than 10 receptacle and Cove . E If the integer part consists of a single zero , the zero before the decimal point is shown not used in children	########## .#########
	12345,67890123 0729 , -
After the point you want to see exactly 10 characters, untill a melting space will be filled with zeros. E If the integer part of the fraction less than 8 digits, missing .sites are of a full zeros	00000000.00000000000
	00012345,67890123000 00000000.07290000000 -
To the left of the decimal point are "laws " # 0", and the right - the "Coll of us	########## .0
	0.0729 -

The number is translated to a percentage by multiplying by 100 and d of bavleniem %sign	P 7.29%-
The number is shown in exponential form	E 1.234568E + 004
The number is shown as the currency of the country that Windows is configured for	C RUR 345.68 12

The last line of the procedure "with bugs and Kashtanka" indicates that the inside of Kav 's check, we can add any character and they will be displayed on the respective th boiling places. This opens up the possibility of unusual fo r matting. It is only necessary that these characters do not coincide with the standard ones, otherwise VB .will get confused

Improving the calculator . Here you and the first improvement for our calculator, that it is not indicating l long results in exponential form as those. Instead

(Result. Text = Val (Number1. Text) / Val (Number2. Text

can write

.Result .Text = Format (Val (Number 1.Text) / Val (Number 2.Text), "0 ("####################

Just keep in mind - if your result is so small, that the twenty numbers located me after the point is not enough , there is nothing other than zero, you will not see, but the exponential format will show you the result , let and ... unusual for Thu e Nia

More about the benefits of variables .5.4.8

Variables are fast . The values of variables are not required, like those to the stu el e ment s control , appear " on the surface of slow project " . They spryat and us deep into the ultra-fast RAM computer and there above them conveniently etc. of the drive computing and various logical transformations. In fact, all thought and work of the computer is carried out on variable quantities. And only sometimes, when a person is required, they are displayed " on the surface " in the form of soda p extensible text fields or

.anything else

Creating a calculator, we did not know about any variable, so instead of

Izzy u Foot

Res = Number1 + Number2

wrote bulky

(Result. Text = Val (Number1. Text) + Val (Number2. Text

Generally, attempts to use for computing the variables are to Stow fields like trying clumsy slowpoke Kings (text fields) from various state in darstva agree among themselves. After unsuccessful attempts they cause their sh in stryh Foreign Ministers (variables) that in no dogs and Riva so camping and give a contract ready for signature by the kings (meaning that the result of the .(calculations show s INDICATES in the text box

Improving the calculator . With that said, let's try to improve the :calculator program

```
Public Class Form1
    Inherits System.Windows.Forms.Form
    Windows Form Designer generated code
    Dim Num 1 of As Double Room ' variable , containing a number of text fields Number 1
    Dim Number2 As Double 'Variable containing a number from the Number2 text box
Dim Res As Double 'Result variable intended for the Result text field

    Private Sub Cl _ addition _Click (ByVal sender As System.Object, ByVal _ (e As System.EventArgs
        Handles Cl _ addition . Click
Chis1 = Chislo1.Text 'initial data values transferred from the text fields in lane e mennye
        Number2 = Number2.Text
        Res = Num1 + Num2 'Process variables to get the result
Result.Text = Res ' The result value is passed from the variable to the text field
    End Sub

    Private Sub Cl _ subtraction _Click (ByVal sender As System.Object,
```

```
_ (ByVal e As System.EventArgs
   Handles Cl _ subtraction .Click
Numbers 1 = Number 1.Text
   Number2 = Number2.Text
  Res = Number1 - Number2
      Result.Text = Res
          End Sub
```

It does not show the procedures s multiplication and division, since they are completely anal about tech procedures am adding and subtracting. Variables *Chis1* , *Chis2* and *res* occur in each procedure is, therefore , in this case, is convenient to declare they are not in every facials at p e, and a time - out procedures . Now each procedure and may these is variable bubbled uses of vatsya. Variable bubbled is declared within procedures, s, can only use the procedures and, where they are declared, and others may not. Traditions n but the announcement of such "common" is variable 's done before the text of all .procedures as sd e lal me

The rest is clear from the comments to the text of the program . We can not say that the progra m m and is shorter , we do not yet know how to write a short program s . But we got into s possibility to work quietly with Variable bubbled, and this possibility, we need to used at duschem when we will fight .with the introduction of non-numeric text fields Inform a tion

This scheme, when the information of the text fields (or other means of job and with initial data) is transferred to the variables is then processed, and so then of Lane e variables passed back - into the text field - very reasonable .and I would recommend it uses about vatsya

Three tips . Roads tion of the reader and the novice ! In from you :three of my advice, in their strength approaching indisputable orders

Programs that you see in the book, you need to enter in a .1 computer w ter, and do them, even if they seem to be clear to you s E, and even if I do not ask for it explicitly. In some cases, you get unexpected results, from which to conclude that these programs do not you realize until the end , and "as a chorus of Shaw, I'm not too lazy to . "check them

In each of these programs experiment, that is, in different ways .2 change in them what I am explaining at this moment. For example, if I am about to Kommersant yasnyayu operator For i = 1 To 5 , try For i =

.1 To 10 and QS t Rita, that is Complete and check with the answer all tasks. This, of course, the .3 SFA in ny council of three. Note that verified with the correct answer the slave of the melting program - your victory, Reconciled to answer correctly p a bot program - a temporary setback, rejection of .reconciliation - ra of thunder

If you regret the time and neglect to these tips, then a n e number of pages may find difficulty in understanding the material and will soon not be .able to prospect and correctly comp and build most programs

Prefixes .5.4.9

Why do you think we wrote

```
Dim Number 1 As Double
Dim Numbers 2 As Double
Dim Res As Double
```

and not

```
Dim Number 1 As Double
Dim Number 2 As Double
Dim Result As Double
```

Because the names *Number1* , *number2* and *Result* already have text fields, and called s Vat same name different objects - then make awry himself and to m pewter. And yet, and exchange *Chis1* , *Chis2* and *res* invented is not very good, because of him not even guess what words they cut may be "?that "cleaning" and "cramps n ka

In practice, programming all the time there are situations when one and kovymi want to call the names of different objects. In such cases, some professiona with sional programmers start each name with the so-called **prefix** that says so forth of programmers (not a computer at!) About who owns the name (button (prefix *btn*), string n e belt (prefix *of str*) or someone .(something else

:In our case, we could name the text fields like this

, txt Number1 , txt Number 2 , txt Result

:and variables like this

. dbl Number1 , dbl Number 2 , dbl Result

And now about the sense of proportion. Some of the names, whether it

is a variable name, shape, or other element in the program s used very often. For example, the so-called loop variable, which we will go through with you, is usually called the letter *i* . If such variables to give " Science " long names with prefixes in the text and the necks of the program will be very cumbersome. Therefore, no one you will not be offended t camping, if you instead

int LoopVar = int LoopVar + 1
*(y (int CycleVariable) = 2 * a (int CycleVariable*

just write

i = i + 1
*(y (i) = 2 * a (i*

There is a more or less generally accepted agreement about what objects and what types to denote with what prefixes. I myself don't like prefixes and rarely use them.

How to create a simple computational .5.5 project

Everyone knows the saying “Order beats the class”. This was clearly demonstrated by the Greek national team at the 2004 European Football Championship. This saying is also true in the progra m mation. Even an inexperienced newbie will succeed if he follows the proper order of the project creation.

All that is said is true not just for computing but also for all other prospect of OBJECTS. Consideration will be carried out by example.

Problem : Given the dimensions of a matchbox. Calculate the area of the base koro b ki , its about b eat and the total surface area .

:Project creation procedure

The programmer himself must know the solution to the problem. .1
After the program - it is ins t ruktsiya to deal with it. Should not be given, and n struction, not knowing how to deal with. Keep in mind that the computer itself knows absolutely nothing and all he has to explain to s Vat.

In our case, the programmer must know the formulas to calculate

: everything that is needed

- **base area = width x thickness**
- **volume = base area x height**
- **base perimeter = two widths + two thicknesses**
- **lateral surface area = base perimeter x height**
- **the total surface area of two playground and base + the = lateral surface area of STI**

As you can see, I for harmony computing introduced the base perimeter .and area of the side of the howling of the surface

The programmer should know the solution to the problem absolutely clearly, absolutely precisely, absolutely correctly. The slightest lack of clarity, inaccuracy or incorrectness will lead to incorrect results. You may not know that a professional prospect of programmers, get serious problem is not evident immediately at the computer, and the first for days, weeks or even months, surrounded by books, solves the problem of "the mind" or on the . boom and n

.2 You need to come up with names for the variables. The name of a variable should speak of its meaning. If the meaning is the width of the box, there is no fief and Tes, and do not call her *and* because six months later, sorting out his half-forgotten program, you will long to rub his forehead and thinking - What the hell, I will be denoted by *a* ? Call its *width* (ie with if you do not know English) or, for example, *the Width* (if you know). So do all the prospect of occupational programmers (and all they are known - notorious lazy and hate to work for nothing ϑ), receptacle and cheat, whatever it is ? they need

:Let's be satisfied with these names

Width - width Thickness - thickness Height - height S _ bases - base area V - volume

Perimeter - base perimeter S _ side - lateral surface area S _ full - total surface area

It is necessary to determine what type the variables will be. As we **.3** advance n th known whether the original data will be safe, we declare all :variables - *Double Room* . Pe p stems line programs will be so , and E

Program for calculating base area, volume and "
total surface area
matchbox according to its known dimensions "

```
Private Sub Button1_Click ( ByVal sender As System.Object, ByVal e As
System.EventArgs) Handles Button1.Click
    Declaring variables'
    Dim Width As Double ' Box Width
    Dim Thickness As Double ' Thickness
    Dim Height As Double ' Height
    Dim S_foundation As Double 'base area
    Dim V As Double ' volume
    Dim Perimeter As Double 'Base Perimeter
    Dim S_side As Double 'lateral surface area
    Dim S_complete As Double 'total surface area
```

Create a project with a button. Enter these lines. Don't forget *End Sub* . Run the project. Nothing that the results are still far away! VB did not issue any errors ? So, while in the Council of Europe in smacking e ke. Do not forget that after you enter the next 2-3 lines of the project need to run and .work, etc. for facility to verify

Before calculations, you need to specify the initial data for solving .4 the problem. To do this, you need to come up with a way in which the user is behind a Vat dimensions of the box - with the help of text fields, function . *InputBox* , or somehow different. Let's *select InputBox* :Here are the following lines of the program

```
    Input of initial data'
    ("Width = InputBox ("Enter the width of the box
("Thickness = InputBox ("Enter the thickness of the box
    ("Height = InputBox ("Enter the height of the box
```

Now you need to specify the computer the actions that it should .5

.be done with the original data, to the floor at the chit result

```
Calculation of results'
S_bases = Width * Thickness
V = S_bases * Height
Perimeter = 2 * Width + 2 * Thickness
S_side = Perimeter * Height
S_complete = 2 * S_bases + S_side
```

Remember that the order in which statements are written can be critical. .So, for example , measure, you cannot swap the last two lines of a fragment

Upon receipt of the result s of their need to show people . Indeed, **.6** the computer performs all assignment operators " in the mind " . After their execution, the memory cells will contain the numerical results of solving the problem. That they have to , and de five, a person can use the text field, as we have just done, ulu h Shaya calculator. I use the familiar *Debug.WriteLine* , :but with innovations

```
Displaying results'
(Debug.WriteLine ("Base area =" & S_bases
(Debug.WriteLine ("Volume =" & V
(Debug.WriteLine ("Total surface area =" & S_full
End Sub
```

Please note that here in the first statement *Debug.WriteLine* - two elements of the output divided by the symbol **&** : textual explanation *"Footprint = "* and sobs so venno variables that we want to see - *S_osnov* . . The same is true for the other *Debug.WriteLine statements*

Run the project and make sure it works. Enter as ra h mer boxes of **.7** :4, 2, 5. Here are the results

```
Base area = 8
Volume = 40
Total surface area = 76
```

.Try to enter fractional numbers and check the result

You ask - why was it so long to work to solve such a simple problem? Indeed, for simple initial data, this problem is faster to solve even in the mind. About Dr. Nako, compliance given me the procedure for compiling the program will facilitate you in the future programming of real-world problems for your computer. Let me remind you: start with data input, then convert them into variables, and then, on the b rabatyvaya variables, we get a result,

then we convert the results s tat of the variables into something in and dimoe .. This procedure is generally accepted

Output options . Results can be displayed in a text paragraph of les, : and tag, and in *the MsgBox* . Here are some examples

```
TextBox1. Text = "Base area =" & S_foundations
Label1. Text = "Volume =" & V
(MsgBox ("Total surface area =" & S_full
```

Compare this to displaying the same results with *Debug.WriteLine* . .You can see that the output is exactly the same

:And here are the same results, but formatted in the same way

```
("### .TextBox1.Text = "Base area =" & Format (S_bases, " 0
("### .Label1.Text = " Volume =" & Format (V, " 0
(("### .MsgBox ("Total surface area =" & Format (S_full, " 0
```

The car drove for 3 hours at a speed of 80 km / h and for 2 hours at a speed of 90 km / h. In the s number of middle vehicle speed (it is equal to the total path divided by Su m Marne time). The values of variables are set assignment operators, the result is called a typing oper of rum . *Debug.WriteLine* with 3 characters after the decimal point

In the very corner of the rectangular courtyard stands a rectangular house. Calculate the area of the house, the free area of the yard and the length of the fence. Note: in a corner where the house is, Zab about ra, of course, not. The dimensions of the house and yard are entered using *the InputBox* , results with n of yasneniyami from about maps the text boxes and labels . All . numbers are integers

.In Task 18 you will draw a blueprint for the yard and house

Calculate the circumference and area of a circle. The radius of the circle set in oper about re *the Dim* . Results 5 characters after the decimal point and . with the notes display in two *the MsgBox*

String variables .5.6

Variables can be more than just numeric. The next type of Move n GOVERNMENTAL, with whom we met, this string variables. String variables are very important. Without them, for example, a conversation with a computer is impossible . Anyway, the receptacle and the considerable part .(of the information with which to work the computer - text (ie, lines of Vai

Getting to know strings .5.6.1

:Create a two button project with a program like this

```
Private Sub Button1_Click ( ByVal sender As System.Object, ByVal e As
                                    System.EventArgs) Handles Button1.Click
                                                Dim a As Integer
                                                    a = 98
                                               (Debug.WriteLine (a
                                                End Sub

Private Sub Button2_Click ( ByVal sender As System.Object, ByVal e As
                                    System.EventArgs) Handles Button2.Click
                                                Dim b As String
                                            "! b = " Hello everyone
                                               (Debug.WriteLine (b
                                                End Sub
```

. Let's compare the two procedures

Declared of *Dim a As Integer* indicates that the variable *a* is required to have the number of a new value, and therefore, in the first procedure is, the . operator *a = 98* records into a cell *a* chi a lo 98

Declared of **Dim b As String** indicates that the variable *b* must have rows of howling (text) value . This means hours then its value is not a number, and arbitrary s naya string of characters, such as *Hello everyone!* or *rpN2N (+ * FD6: u* . About Emperor of the *b = "! Hello world"* written in the memory *b* string *Hi all!* operator. *Debug.WriteLine (b)* , because it must always displayed on screen with a contents of the cell *b* , *!*CONCLUSIONS e children on the screen text *Hello everyone*

Note that in the code window text should be taken in quotation marks, and it is stored in the memory without Kav 's check and

displays without the quotes. And when you enter it in a text box or .InputBox , you also enter it without quotes

The information in the memory cell by a string variable can be in ful l neniya programs vary in the same way as in the cell to a numeric variable. H as an example , when performing procedures

```
Private Sub Button3_Click ( ByVal sender As System.Object, ByVal e As
System.EventArgs) Handles Button3.Click
    Dim a As String
    a = " Wait a minute !" : Debug.WriteLine (a): a = " Hello !" : A = " Up
(to date !" : D e bug.WriteLine (a
End Sub
```

rows will appear in cell *a*

!Wait a minute! Hello! Goodbye

:and the following lines will be displayed on the screen

```
! Wait a minute
! Up to date
```

The cell sizes for the string variable are not fixed, they directly depend .on the string size

The & operation . Some operations can be performed on strings as well as on numbers. The simplest operation is denoted by the symbol **&** , and we had a sign about us - it just Port and nyaet strings into one. Consider the : program

```
Private Sub Button4_Click ( ByVal sender As System.Object, ByVal e As
System.EventArgs) Handles Button4.Click
    " Dim S As String = " trouble
    Dim T As String
    T = " By " & S
    (Debug.WriteLine (T
    (Debug.WriteLine (T & T & T
    (Debug.WriteLine (T + T + T
End Sub
```

:This is what it prints

```
Victory
VictoryVictoryVictory
VictoryVictoryVictory
```

You can see that the + operation on strings is the same as the **&** . operation

String expressions and literals . Into the expressions we were looking for

Po " & S "

T & T & T

("### .Volume =" & Format (V , "0 "

and others are nothing more than **string expressions** . For the simple reason that actions are performed in them, the result of which is a string. Specific double quoted text strings , which we see in the box, the code will be called **string literals** . So, in the above page about postglacial expressions we see the literal *" On "* , *" volume ="* , *"Goodbye!"* other. Page of kovye same expression, as well as any other statements are not enclosed in double quotes. .They conclude that only for specific e string of letters and ly

The & operation on numbers . If an operation & apply to the numbers, it is their "force" in the line and converts the combined and m as : used in one row. Ra to look at the program

```
Private Sub Button5_Click ( ByVal sender As System.Object, ByVal e As
System.EventArgs) Handles Button5.Click
    Dim a As Integer = 1234
    (" Debug.WriteLine (" There was " & 76 & a & a & 908 & " hares
End Sub
```

:This is what it prints

```
There were 7612341234908 hares here
```

Of course, the string of numbers *7612341234908* is just a string of characters and has nothing to do with the real number. They are just numbers . that have become strings and then concatenated together

:Determine without a computer what the following program will print

```
Private Sub Button6_Click ( ByVal sender As System.Object, ByVal e As
System.EventArgs) Handles Button6.Click
    " Dim a As String = " Loop
        a = a & a
    (Debug.WriteLine (a
        a = a & a
```

(" Debug.WriteLine (a & a & " End of cycle
End Sub

An example . String variables can be defined not only by the operator
: to assign to a Nia, but also a function of *the InputBox*

```
Private Sub Button 7_ Click ( ByVal sender As System . Object , ByVal e
As System . EventArgs ) Handles Button 7. Click
Dim a As String
(" a = InputBox ( " Enter any - any word
Label1.Text = " You entered the word " & a
End Sub
```

Suppose after pressing the button you entered the word *Forward* . In
: from the result in the label

```
You entered the word Forward
```

Note the space between the letters of the second "o" and the quote in a literal *"you enter the word "* . The space is needed so that the words do not
:merge during the output. If it was not, p e result would be this

```
You entered the word
```

Dialogue with the computer .5.6.2

Write a program that would carry out such a dialogue of man with
:Compute e rum

COMPUTER displayed on the screen: Hello, I have to m pewter, and
?what's your name

A MAN TYPES IN FROM THE KEYBOARD: *Kolya*

THE COMPUTER DISPLAYS ON THE SCREEN: Very nice, Kolya.
?How old are you

PERSON TYPES FROM KEYBOARD: *16*

COMPUTER DISPLAYS ON SCREEN: Wow! As many as 16 years! You
!are already quite an adult

It is obvious that the person in the course of the conversation has the
.right to enter the keyboard sorts them e on and whatever the age

Let the computer asks its questions in the *InputBox* , the person enters his answers into the same *InputBox* , and the computer sends the last replica to the *MsgBox* . To store in the memory of a person's name contrive variable
. *imya* , and for in s rasta - *vozrast*

: Here's the program

```
Private Sub Button8_Click ( ByVal sender As System.Object, ByVal e As
System.EventArgs) Handles Button8.Click
    Dim imya As String
    Dim vozrast As Integer
("?imya = InputBox ("Hello, I'm a computer, what's your name
("?vozrast = InputBox ("Nice, " & imya & ". How old are you
MsgBox ( "Wow! Whole " & vozrast & " years! You are already quite
( "!an adult
End Sub
```

Notice the commas, periods, and some spaces inside the quotes. If they : were not there, the words would merge like this

THE COMPUTER DISPLAYS ON THE SCREEN: Very nice Kolya How ?old are you

COMPUTER DISPLAYS ON SCREEN: Wow! As many as 16 years! You !are already quite an adult

The dialogue will differ only in the information that the person enters. So, at other times it will be carried out on the trail of the same program th :conductive dialogue

COMPUTER: Hello, I'm a computer, what's your name? ***MAN:*** *Fantômas!* **COMPUTER:** Very nice, Fantomas! ... How old are you? ***PERSON:*** *100* **COMPUTER:** Wow! As many as 100 years! You are !already quite an adult

Don't think that this program is very smart. It does not analyze at all what information the person entered from the keyboard. Therefore, such a :dialogue is possible with her

COMPUTER: Hello, I'm a computer, what's your name? ***MAN:*** *Get out of my sight!* **COMPUTER:** Very nice, Get out of my sight !. How old are you? ***HUMAN:*** *-2* **COMPUTER:** Wow! As many as -2 years! You are !already quite an adult

Write a program for the following task: Computer zaprash and Vaeth the names of two planets, their orbits radii (in millions of kilometers) and speed DV and zheniya in orbit (in millions of kilometers per day). After that, :he calculates cont and telnost year on planets and gives the result in the form *The duration of the year on Earth - 365 days, while the planet Eoella - 12 in . the current*

Result is needed in three variants: the output in those to STOV field , . mark and *MsgBox*

Note for those who do not know the physics and geometry : Year is a time for a company in its orbit, and the orbit is equal to the length divided by .the IC on growth in its orbit . Orbit length is $2 \pi R$, where R - radius of r bits

Let's look around .5.6.3

Well, with Variable bubbled values of s we understand. Learned something. And have we learned as a result of this knowledge to do something new and interesting? It seems no. Nothing particularly pleasant, with the exception, perhaps, of dialogue with the company th Intermediate ?location. Nda-ah ... Why all the torment

Here's why : The material in this chapter - the cartridge without a gun, with unnecessary and ma own thing. But the material in the following . chapters is a variety of pistols without cartridges