

Find solutions for your homework

Search

home / study / engineering / computer science / computer science questions and answers / dealing with polymorphism and inheritance 1.1. creat...

## Question: Dealing with Polymorphism and Inheritance 1.1. Create class P...

(1 bookmark)

### Dealing with Polymorphism and Inheritance

- 1.1. Create class Person with attributes name, age and gender.
- 1.2. Create class Student with attribute grade and,
- 1.3. Create class Teacher with attribute salary.

Note: Class Student and Teacher must inherit from Person. In main class, create and print two students and one teacher. Apply inheritance, setter and getter.

Sample Output: Student 2:

Name: Rachel

Age: 15

Gender: F

Grade: 2.50

### Expert Answer ⓘ



Anonymous answered this  
428 answers

Was this answer helpful?



Hi,

Please find below code as per your requirement.

Hope this answer helps you.

Thanks.

Answer:-

```

/*****Java Code*****/
/*****Person.java*****/

public class Person {
    //instance variables
    private String name;
    private int age;
    private char gender;

    /**
     * Default constructor
     */
    public Person() {

    }

    /**
     * Parameterized constructor with 3 parameter to build Person object
     * @param name
     * @param age
     * @param gender
     */
    public Person(String name, int age, char gender) {
        super();
        this.name = name;
        this.age = age;
        this.gender = gender;
    }

    /*Getters and Setters*/
    public String getName() {
        return name;
    }
    public void setName(String name) {
        this.name = name;
    }

```

### Post a question

Answers from our experts for your tough homework questions

Enter question

Continue to post

20 questions remaining



Snap a photo from your phone to post a question

We'll send you a one-time download link

888-888-8888

Text me

By providing your phone number, you agree to receive a one-automated text message with a link to get the app. Standard messaging rates may apply.

### My Textbook Solutions



Add a textbook

Fundament... Mechanics...

7th Edition

5th Edition

... (2)

View all solutions

```

    }
    public int getAge() {
        return age;
    }
    public void setAge(int age) {
        this.age = age;
    }
    public char getGender() {
        return gender;
    }
    public void setGender(char gender) {
        this.gender = gender;
    }
}

/**
 * This method returns String representation of Person
 */
@Override
public String toString() {
    return "Name: "+name+"\nAge: "+age+"\nGender: "+gender;
}

}

/*****Student.java*****/
/

/**
 * This class is subclass of Person and represent as Student
 *
 */
public class Student extends Person{

    //instance variable
    private double grade;

    /**
     * Default constructor
     */
    public Student() {
        super();
    }

    /**
     * Parameterized constructor which takes parameter two construct Student
     * @param name
     * @param age
     * @param gender
     * @param grade
     */
    public Student(String name, int age, char gender,double grade) {
        //calling superclass constructor
        super(name, age, gender);
        this.grade = grade;
    }

    /**Getters and Setters*/
    public double getGrade() {
        return grade;
    }

    public void setGrade(double grade) {
        this.grade = grade;
    }

    /**
     * This method returns String representation of Student
     */
    @Override
    public String toString() {
        //using super.toString to get person toString
        return super.toString() +"\nGrade: "+grade;
    }

}

}

/*****Teacher.java*****/

```

```

/**
 * This class is subclass of Person and represent as Teacher
 *
 */
public class Teacher extends Person{

    //instance variable
    private double salary;

    /**
     * Default constructor
     */
    public Teacher() {
        super();
    }

    /**
     * Parameterized constructor which takes parameter two construct Teacher
     * @param name
     * @param age
     * @param gender
     * @param salary
     */
    public Teacher(String name, int age, char gender,double salary) {
        //calling superclass constructor
        super(name, age, gender);
        this.salary = salary;
    }

    /*Getters and Setters*/
    public double getSalary() {
        return salary;
    }

    public void setSalary(double salary) {
        this.salary = salary;
    }

    /**
     * This method returns String representation of Teacher
     */
    @Override
    public String toString() {
        //using super.toString to get person toString
        return super.toString() + "\nSalary: "+salary;
    }

}

/*****Driver.java*****/
/

/**
 * This is driver class to test Student and Teacher
 *
 */
public class Driver {

    public static void main(String[] args) {

        //Creating student object with parameterized constructor
        Student student1 = new Student("Rachel", 15, 'F', 2.50);

        //printing student1 object using toString
        System.out.println("Student 1:");
        System.out.println(student1.toString());

        //Creating student object with default constructor
        Student student2 = new Student();
        //Setting all other values to student using setters
        student2.setName("Helen");
        student2.setAge(20);
        student2.setGender('F');
        student2.setGrade(4.8);

        //printing student2 object using toString
        System.out.println("\nStudent 2:");
        System.out.println(student2.toString());
    }
}

```

```
//Creating teacher object with parameterized constructor
Teacher teacher = new Teacher("Richa", 40, 'F', 50000);

//printing teacher object using toString
System.out.println("\nTeacher: ");
System.out.println(teacher.toString());

}
}

/*****Output*****/
```

```
Student 1:
Name: Rachel
Age: 15
Gender: F
Grade: 2.5

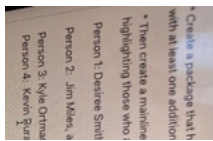
Student 2:
Name: Helen
Age: 20
Gender: F
Grade: 4.8

Teacher:
Name: Richa
Age: 40
Gender: F
Salary: 50000.0
```

[View comments \(2\) >](#)

## Up next for you in Computer Science

java programming/ each person should include name and last name, age, ethnicity, religion, email a...



[See answer](#)

For this exercise, you need to write TWO classes. (To enable two classes to coexist in the same .java file, only one of them can be public (and shares its name with the name of the .java file). The structure of

[See answer](#)

[See more questions for subjects you study](#)

## Questions viewed by other students

Q: Create Java Project java\_inheritance\_yourNameLastname that; SECTION3 (under source directory called, src, create a package csci3444.inheritance and put all code in it) Create an interface called "Person" with below methods; public String getName(); public String getDetails(); Create an abstract class called "PersonBaseImpl" that implements "Person" interface. has attribute...

A: [See answer](#) 100% (1 rating)

Q: How much heat is absorbed when 40.00g of C(s) reacts in the presence of excess SO<sub>2</sub>(g) to produce CS(l) and CO(g) according to the following chemical equation 5C(s)+2SO<sub>2</sub>(g)-->CS<sub>2</sub>(l)+4CO(g) ^H=239.9kJ

A: [See answer](#) 100% (2 ratings)

[Show more v](#)

---

**COMPANY**

[About Chegg](#)  
[Chegg For Good](#)  
[College Marketing](#)  
[Corporate Development](#)  
[Investor Relations](#)  
[Jobs](#)  
[Join Our Affiliate Program](#)  
[Media Center](#)  
[Site Map](#)

**LEGAL & POLICIES**

[Advertising Choices](#)  
[Cookie Notice](#)  
[General Policies](#)  
[Intellectual Property Rights](#)  
[Terms of Use](#)  
[Global Privacy Policy](#)  
[DO NOT SELL MY INFO](#)  
[Honor Code](#)  
[Honor Shield](#)

**CHEGG PRODUCTS AND SERVICES**

[Cheap Textbooks](#)  
[Chegg Coupon](#)  
[Chegg Play](#)  
[Chegg Study Help](#)  
[College Textbooks](#)  
[eTextbooks](#)  
[Flashcards](#)  
[Learn](#)  
[Chegg Math Solver](#)

**CHEGG NETWORK**

[EasyBib](#)  
[Internships.com](#)  
[Thinkful](#)

**CUSTOMER SERVICE**

[Customer Service](#)  
[Give Us Feedback](#)  
[Help with eTextbooks](#)  
[Help to use EasyBib Plus](#)  
[Manage Chegg Study Subscription](#)  
[Return Your Books](#)  
[Textbook Return Policy](#)

