LOKESHVISWA M

22/07/2025

1)Create a class named Student.Include the following protected member variables.

name,id,age,grade,address

Include appropriate getters and setters

Include a default constructor and a 5-argument constructor. The order of arguments in the 5- argument constructor is name, id, age, grade and address.

Include the following public methods in the Student class.

void display()

Display the details of the student.

boolean isPassed()

A student is said to have passed if his/her grade is above 50. This method returns true if the student has passed.

Create a subclass of Student named UGStudent .

Include the following private member variables.

degree

stream

Include appropriate getters / setters

Include a default constructor and a 7-argument constructor. The order of arguments in the 7-argument constructor is name, id, age, grade, address, degree and stream.

Include the following public methods in the UGStudent class.

void display()

Display the details of the ug student in the format as specified in the output.

boolean isPassed()

A ug student is said to have passed if his/her grade is above 70. This method returns true if the student has passed.

Create another subclass of Student named PGStudent .

Include the following private member variables.

specialization

noOfPapersPublished

Include appropriate getters / setters

Include a default constructor and a 7-argument constructor. The order of arguments in the 7-argument constructor is name, id, age, grade, address, specialization and number of papers published.

Include the following public methods in the PGStudent class.

void display()

Display the details of the pg student in the format as specified in the output.

boolean isPassed()

A pg student is said to have passed if his/her grade is above 70 and if he/she has published atleast 2 papers.This method returns true if the student has passed.

Create a class called Main to test the above classes.

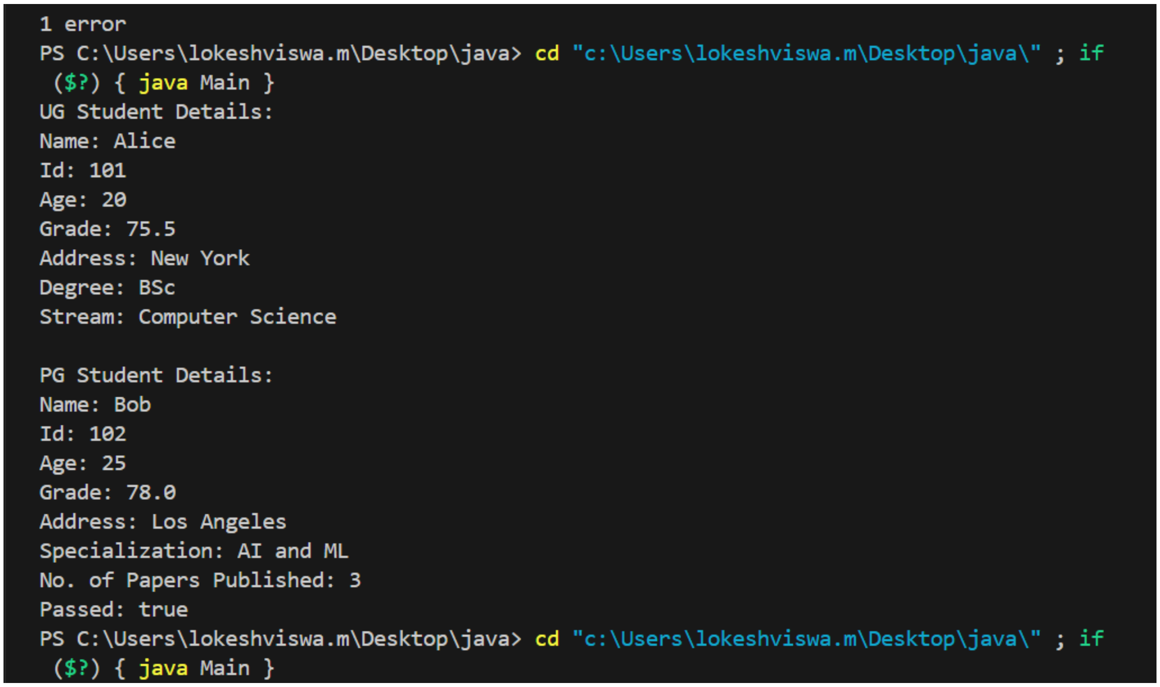
OUTPUT:

```
class Student {

        protected String name;

        protected int id;

        protected int age;

        protected double grade;

        protected String address;
```

```
public Student() {
    this.name = "";
    this.id = 0;
    this.age = 0;
    this.grade = 0.0;
    this.address = "";
}


public Student(String name, int id, int age, double grade, String address) {
    this.name = name;
    this.id = id;
    this.age = age;
    this.grade = grade;
    this.address = address;
}


public String getName() { return name; }
public void setName(String name) { this.name = name; }

public int getId() { return id; }
public void setId(int id) { this.id = id; }

public int getAge() { return age; }
```

```
    public void setAge(int age) { this.age = age; }

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

    public String getAddress() { return address; }
    public void setAddress(String address) { this.address = address; }


    public void display() {
        System.out.println("Name: " + name);
        System.out.println("Id: " + id);
        System.out.println("Age: " + age);
        System.out.println("Grade: " + grade);
        System.out.println("Address: " + address);
    }

    public boolean isPassed() {
        return grade > 50;
    }
}


class UGStudent extends Student {
    private String degree;
    private String stream;
    public UGStudent() {
        super();
        this.degree = "";
        this.stream = "";
    }


    public UGStudent(String name, int id, int age, double grade, String address, String degree, String stream) {
        super(name, id, age, grade, address);
        this.degree = degree;
        this.stream = stream;
    }


    public String getDegree() { return degree; }
    public void setDegree(String degree) { this.degree = degree; }

    public String getStream() { return stream; }
    public void setStream(String stream) { this.stream = stream; }


    public void display() {
        super.display();
        System.out.println("Degree: " + degree);
        System.out.println("Stream: " + stream);
    }

    public boolean isPassed() {
        return grade > 70;
    }
}


class PGStudent extends Student {
    private String specialization;
    private int noOfPapersPublished;


    public PGStudent() {
        super();
        this.specialization = "";
        this.noOfPapersPublished = 0;
    }


    public PGStudent(String name, int id, int age, double grade, String address, String specialization, int noOfPapersPublished) {
        super(name, id, age, grade, address);
        this.specialization = specialization;
        this.noOfPapersPublished = noOfPapersPublished;
    }


    public String getSpecialization() { return specialization; }
    public void setSpecialization(String specialization) { this.specialization = specialization; }

    public int getNoOfPapersPublished() { return noOfPapersPublished; }
    public void setNoOfPapersPublished(int noOfPapersPublished) { this.noOfPapersPublished = noOfPapersPublished; }


    public void display() {
        super.display();
        System.out.println("Specialization: " + specialization);
        System.out.println("No. of Papers Published: " + noOfPapersPublished);
    }

    public boolean isPassed() {
        return grade > 70 && noOfPapersPublished >= 2;
    }
}


public class Main {
    public static void main(String[] args) {
        UGStudent ug = new UGStudent("Alice", 101, 20, 75.5, "New York", "BSc", "Computer Science");

        System.out.println("UG Student Details:");

        ug.display();

        System.out.println("Passed: " + ug.isPassed());

        System.out.println();



        PGStudent pg = new PGStudent("Bob", 102, 25, 78.0, "Los Angeles", "AI and ML", 3);

        System.out.println("PG Student Details:");

        pg.display();

        System.out.println("Passed: " + pg.isPassed());

    }
```

OUTPUT:



```
1 error
PS C:\Users\lokeshviswa.m\Desktop\java> cd "c:\Users\lokeshviswa.m\Desktop\java\" ; if
 ($?) { java Main }
UG Student Details:
Name: Alice
Id: 101
Age: 20
Grade: 75.5
Address: New York
Degree: BSc
Stream: Computer Science

PG Student Details:
Name: Bob
Id: 102
Age: 25
Grade: 78.0
Address: Los Angeles
Specialization: AI and ML
No. of Papers Published: 3
Passed: true
PS C:\Users\lokeshviswa.m\Desktop\java> cd "c:\Users\lokeshviswa.m\Desktop\java\" ; if
 ($?) { java Main }
```

2) The task is to get the details of the vehicle and display the details using a menu driven application.

Write a Java program to Implement this task.

Create a class Vehicle

Include the following protected data members / attributes:

make – of type String

vehicleNumber – of type String

fuelType – of type String

fuelCapacity - of type Integer

cc – of type Integer

Include the following public methods

Create a constructor that initializes all the data members --- public Vehicle(String make,String vehicleNumber,String fuelType,Integer fuelCapacity,Integer cc)

displayMake – Display the make of the vehicle

"displayBasicInfo" – display basic information of the vehicle.

"displayDetailInfo" – An empty method.

Create a class TwoWheeler that extends Vehicle

kickStartAvailable – of type Boolean.

"displayDetailInfo" – displays the availability of kick start.

Create a class FourWheeler that extends Vehicle

audioSystem – of type String.

numberOfDoors – of type Integer.

"displayDetailInfo" - displays the audio system and number of doors.

displayDetailInfo - overridden method

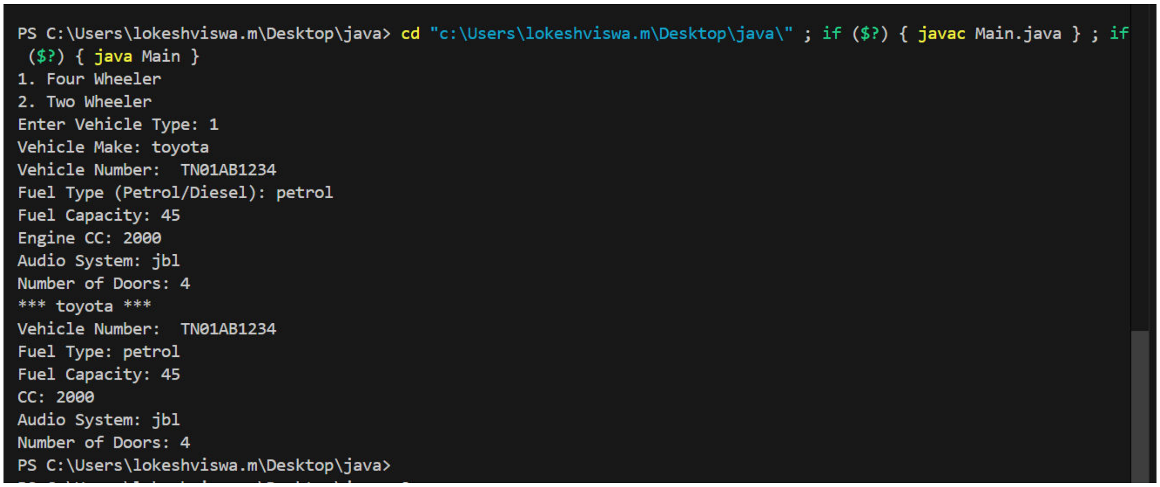
Include getter setters for all the classes.

Create a main class to test the classes defined above.

CODE:

```
import java.util.Scanner;
class Vehicle {
    protected String make;
    protected String vehicleNumber;
    protected String fuelType;
    protected int fuelCapacity;
    protected int cc;

    public Vehicle(String make, String vehicleNumber, String fuelType, int fuelCapacity, int cc) {
        this.make = make;
        this.vehicleNumber = vehicleNumber;
        this.fuelType = fuelType;
        this.fuelCapacity = fuelCapacity;
        this.cc = cc;
    }

    public void displayMake() {
        System.out.println("*** " + make + " ***");
    }

    public void displayBasicInfo() {
        System.out.println("Vehicle Number: " + vehicleNumber);
        System.out.println("Fuel Type: " + fuelType);
        System.out.println("Fuel Capacity: " + fuelCapacity);
        System.out.println("CC: " + cc);
    }

    public void displayDetailInfo() {

    }


    public String getMake() { return make; }
    public void setMake(String make) { this.make = make; }

    public String getVehicleNumber() { return vehicleNumber; }
    public void setVehicleNumber(String vehicleNumber) { this.vehicleNumber = vehicleNumber; }

    public String getFuelType() { return fuelType; }
    public void setFuelType(String fuelType) { this.fuelType = fuelType; }

    public int getFuelCapacity() { return fuelCapacity; }
    public void setFuelCapacity(int fuelCapacity) { this.fuelCapacity = fuelCapacity; }

    public int getCc() { return cc; }
    public void setCc(int cc) { this.cc = cc; }
}


class TwoWheeler extends Vehicle {
    private boolean kickStartAvailable;

    public TwoWheeler(String make, String vehicleNumber, String fuelType, int fuelCapacity, int cc,
boolean kickStartAvailable) {
        super(make, vehicleNumber, fuelType, fuelCapacity, cc);

        this.kickStartAvailable = kickStartAvailable;

    }


    @Override

    public void displayDetailInfo() {

        System.out.println("Kick Start Available: " + (kickStartAvailable ? "Yes" : "No"));

    }




    public boolean isKickStartAvailable() { return kickStartAvailable; }

    public void setKickStartAvailable(boolean kickStartAvailable) { this.kickStartAvailable = 
kickStartAvailable; }

}




class FourWheeler extends Vehicle {

    private String audioSystem;

    private int numberOfDoors;


    public FourWheeler(String make, String vehicleNumber, String fuelType, int fuelCapacity, int cc, 
String audioSystem, int numberOfDoors) {

        super(make, vehicleNumber, fuelType, fuelCapacity, cc);

        this.audioSystem = audioSystem;

        this.numberOfDoors = numberOfDoors;

    }


    @Override

    public void displayDetailInfo() {
        System.out.println("Audio System: " + audioSystem);
        System.out.println("Number of Doors: " + numberOfDoors);
    }


    public String getAudioSystem() { return audioSystem; }
    public void setAudioSystem(String audioSystem) { this.audioSystem = audioSystem; }

    public int getNumberOfDoors() { return numberOfDoors; }
    public void setNumberOfDoors(int numberOfDoors) { this.numberOfDoors = numberOfDoors; }
}


public class Main {
    public static void main(String[] args) {
        Scanner sc = new Scanner(System.in);

        System.out.println("1. Four Wheeler");
        System.out.println("2. Two Wheeler");
        System.out.print("Enter Vehicle Type: ");
        int choice = sc.nextInt();
        sc.nextLine();

        System.out.print("Vehicle Make: ");
        String make = sc.nextLine();

        System.out.print("Vehicle Number: ");
        String number = sc.nextLine();
        System.out.print("Fuel Type (Petrol/Diesel): ");
        String fuel = sc.nextLine();

        System.out.print("Fuel Capacity: ");
        int fuelCapacity = sc.nextInt();

        System.out.print("Engine CC: ");
        int cc = sc.nextInt();
        sc.nextLine();

        if (choice == 1) {
            System.out.print("Audio System: ");
            String audioSystem = sc.nextLine();

            System.out.print("Number of Doors: ");
            int doors = sc.nextInt();

            FourWheeler car = new FourWheeler(make, number, fuel, fuelCapacity, cc, audioSystem, doors);
            car.displayMake();
            car.displayBasicInfo();
            car.displayDetailInfo();

        } else if (choice == 2) {
            System.out.print("Kick Start Available (yes/no): ");
            String kick = sc.nextLine();
            boolean kickStart = kick.equalsIgnoreCase("yes");
            TwoWheeler bike = new TwoWheeler(make, number, fuel, fuelCapacity, cc, kickStart);

            bike.displayMake();

            bike.displayBasicInfo();

            bike.displayDetailInfo();


        } else {

            System.out.println("Invalid choice.");

        }


        sc.close();

    }

}
```

OUTPUT:



```
PS C:\Users\lokeshviswa.m\Desktop\java> cd "c:\Users\lokeshviswa.m\Desktop\java\" ; if ($?) { javac Main.java } ; if
 ($?) { java Main }
1. Four Wheeler
2. Two Wheeler
Enter Vehicle Type: 1
Vehicle Make: toyota
Vehicle Number:  TN01AB1234
Fuel Type (Petrol/Diesel): petrol
Fuel Capacity: 45
Engine CC: 2000
Audio System: jbl
Number of Doors: 4
*** toyota ***
Vehicle Number:  TN01AB1234
Fuel Type: petrol
Fuel Capacity: 45
CC: 2000
Audio System: jbl
Number of Doors: 4
PS C:\Users\lokeshviswa.m\Desktop\java>
```

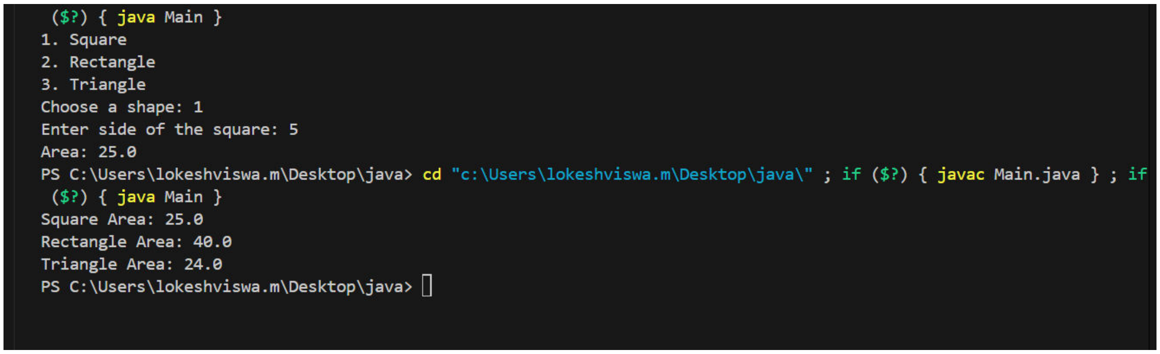
3) Create a class Shape and inherit three classes Square,Triangle and Rectangle. Implement the

method double calculateArea() in Shape class and override the method in the subclasses. Use runtime polymorphism to call the calculateArea() method.

CODE:

```
class Shape {
    public double calculateArea() {
        return 0.0;
    }
}


class Square extends Shape {
    private double side;

    public Square(double side) {
        this.side = side;
    }

    @Override
    public double calculateArea() {
        return side * side;
    }
}


class Rectangle extends Shape {
    private double length;
    private double breadth;

    public Rectangle(double length, double breadth) {
        this.length = length;
        this.breadth = breadth;
    }

    @Override
    public double calculateArea() {
        return length * breadth;
    }
}


class Triangle extends Shape {
    private double base;
    private double height;

    public Triangle(double base, double height) {
        this.base = base;
        this.height = height;
    }

    @Override
    public double calculateArea() {
        return 0.5 * base * height;
    }
}


public class Main {
    public static void main(String[] args) {
        Shape shape;



        shape = new Square(5);

        System.out.println("Square Area: " + shape.calculateArea());



        shape = new Rectangle(10, 4);

        System.out.println("Rectangle Area: " + shape.calculateArea());



        shape = new Triangle(6, 8);

        System.out.println("Triangle Area: " + shape.calculateArea());

    }

}
```

OUTPUT:



```
 ($?) { java Main }
1. Square
2. Rectangle
3. Triangle
Choose a shape: 1
Enter side of the square: 5
Area: 25.0
PS C:\Users\lokeshviswa.m\Desktop\java> cd "c:\Users\lokeshviswa.m\Desktop\java\" ; if ($?) { javac Main.java } ; if
 ($?) { java Main }
Square Area: 25.0
Rectangle Area: 40.0
Triangle Area: 24.0
PS C:\Users\lokeshviswa.m\Desktop\java> 
```

4) A Company provides an initial training for all its employees, once they join the company. During the training phase they call the employees as “Associate”. The initial training is conducted for 60 days for each Associate. In these 60 days they learn various technologies. The first 20 days they learn “Core skills”, the next 20 days they learn “Advanced modules” and the final 20 days they go to the “Project phase”.   Help the Company to find in which phase the associates are in.

Create a class Associate with associateId(int),associateName(String),workStatus(String).

Include getters and setters and constructors.

Add a method trackAssociateStatus

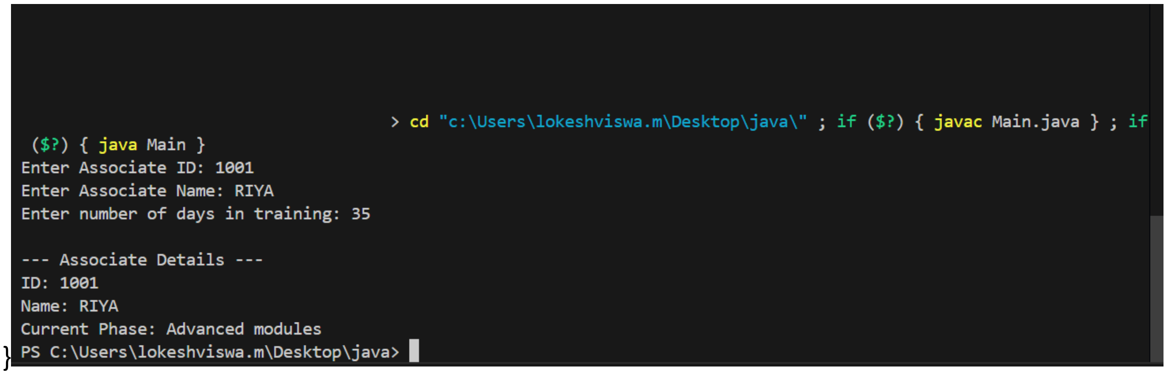
- This method takes the number of days as argument and sets the work status of the associate based on the number of days. If the number of days is greater than 60 days then set the work status as “Deployed in project”.

In the Main class, create an object for the Associate class; Get the details assign the value for its attributes using the setters. Invoke the trackAssociateStatus method and find the work status and display the details.

CODE:

```
import java.util.Scanner;
class Associate {
    private int associateId;
    private String associateName;
    private String workStatus;


    public Associate() {
    }


    public Associate(int associateId, String associateName, String workStatus) {
        this.associateId = associateId;
        this.associateName = associateName;
        this.workStatus = workStatus;
    }


    public int getAssociateId() {
        return associateId;
```

```
}

public void setAssociateId(int associateId) {
    this.associateId = associateId;
}

public String getAssociateName() {
    return associateName;
}

public void setAssociateName(String associateName) {
    this.associateName = associateName;
}

public String getWorkStatus() {
    return workStatus;
}

public void setWorkStatus(String workStatus) {
    this.workStatus = workStatus;
}


public void trackAssociateStatus(int days) {
    if (days <= 20) {
        workStatus = "Core skills";
    } else if (days <= 40) {
        workStatus = "Advanced modules";
        } else if (days <= 60) {
            workStatus = "Project phase";
        } else {
            workStatus = "Deployed in project";
        }
    }
}


public class Main {
    public static void main(String[] args) {
        Scanner sc = new Scanner(System.in);


        Associate associate = new Associate();


        System.out.print("Enter Associate ID: ");
        associate.setAssociateId(sc.nextInt());
        sc.nextLine();
        System.out.print("Enter Associate Name: ");
        associate.setAssociateName(sc.nextLine());

        System.out.print("Enter number of days in training: ");
        int days = sc.nextInt();


        associate.trackAssociateStatus(days);
        System.out.println("\n--- Associate Details ---");

        System.out.println("ID: " + associate.getAssociateId());

        System.out.println("Name: " + associate.getAssociateName());

        System.out.println("Current Phase: " + associate.getWorkStatus());


        sc.close();

    }
```



```
                                        > cd "c:\Users\lokeshviswa.m\Desktop\java\" ; if ($?) { javac Main.java } ; if
 ($?) { java Main }
Enter Associate ID: 1001
Enter Associate Name: RIYA
Enter number of days in training: 35

--- Associate Details ---
ID: 1001
Name: RIYA
Current Phase: Advanced modules
PS C:\Users\lokeshviswa.m\Desktop\java> 
```

```
}
```

OUTPUT: