**Tutorial – 4**

1. Create an interface named Shape that contains empty method named Area () and perimeter(). Provide two classes named Triangle and Circle such that each one of the classes implement Shape. Each one of the classes contains only the method Area () and perimeter().that prints the area and perimeter() of the given shape.

**Code:**

interface Shape {

    void area();

    void perimeter();

}

class Triangle implements Shape {

    private double sideA;

    private double sideB;

    private double sideC;

    public Triangle(double sideA, double sideB, double sideC) {

*this*.sideA = sideA;

*this*.sideB = sideB;

*this*.sideC = sideC;

    }

    public void area() {

        double s = (sideA + sideB + sideC) / 2.0;

        double triangleArea = Math.sqrt(s \* (s - sideA) \* (s - sideB) \* (s - sideC));

        System.out.println("The area of the triangle is: " + triangleArea);

    }

    public void perimeter() {

        double trianglePerimeter = sideA + sideB + sideC;

        System.out.println("The perimeter of the triangle is: " + trianglePerimeter);

    }

}

class Circle implements Shape {

   private double radius;

    public Circle(double radius) {

*this*.radius = radius;

    }

    public void area() {

        double circleArea = Math.PI \* radius \* radius;

        System.out.println("The area of the circle is: " + circleArea);

    }

    public void perimeter() {

        double circlePerimeter = 2 \* Math.PI \* radius;

        System.out.println("The perimeter of the circle is: " + circlePerimeter);

    }

}

class Main1 {

    public static void main(String[] args) {

        Shape triangle = new Triangle(3, 4, 5);

        triangle.area();

        triangle.perimeter();

        Shape circle = new Circle(5);

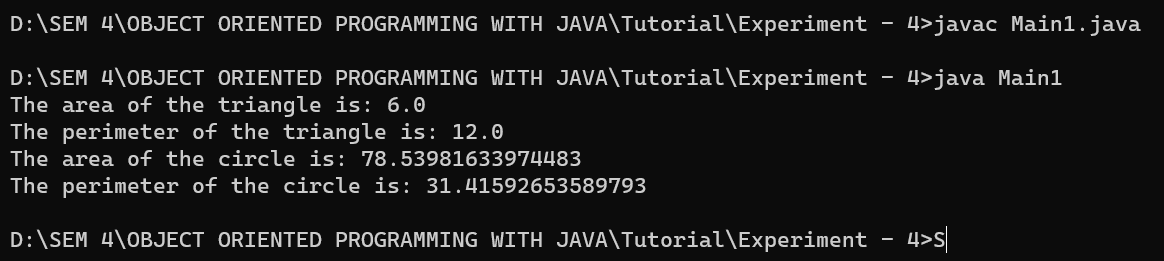
        circle.area();

        circle.perimeter();

    }

}

**Output Screenshot:**

****

1. Create an abstract class Insect that contains abstract method void flystatus(). Provide three classes Temite, Grasshopper and Ant such that each one classes implement flystatus().

**Code:**

abstract class Insect {

    public abstract void flyStatus();

}

class Termite extends Insect {

    public void flyStatus() {

        System.out.println("Termite cannot fly");

    }

}

class Grasshopper extends Insect {

    public void flyStatus() {

        System.out.println("Grasshopper can fly");

    }

}

class Ant extends Insect {

    public void flyStatus() {

        System.out.println("Ant cannot fly");

    }

}

class Main2 {

    public static void main(String[] args) {

        Insect termite = new Termite();

        termite.flyStatus();

        Insect grasshopper = new Grasshopper();

        grasshopper.flyStatus();

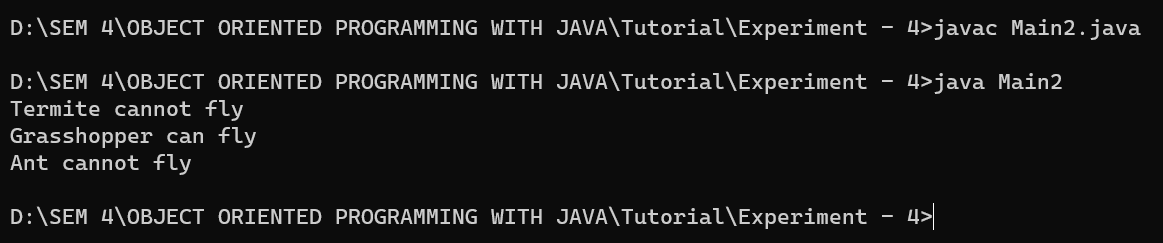
        Insect ant = new Ant();

        ant.flyStatus();

    }

}

**Output Screenshot:**

****

1. Declare a class Vehicle. From this class derive two\_wheeler, three\_wheeler and four\_wheeler class. Display properties of each type of vehicle using member function of the class. (like speed,fuel type,etc.)

**Code:**

class Vehicle {

    private int speed;

    private String fuelType;

    public Vehicle(int speed, String fuelType) {

*this*.speed = speed;

*this*.fuelType = fuelType;

    }

    public void display() {

        System.out.println("Speed: " + speed);

        System.out.println("Fuel type: " + fuelType);

    }

}

class TwoWheeler extends Vehicle {

    public TwoWheeler(int speed, String fuelType) {

*super*(speed, fuelType);

    }

    public void display() {

*super*.display();

        System.out.println("Type: Two-wheeler");

    }

}

class ThreeWheeler extends Vehicle {

    public ThreeWheeler(int speed, String fuelType) {

*super*(speed, fuelType);

    }

    public void display() {

*super*.display();

        System.out.println("Type: Three-wheeler");

    }

}

class FourWheeler extends Vehicle {

    public FourWheeler(int speed, String fuelType) {

*super*(speed, fuelType);

    }

    public void display() {

*super*.display();

        System.out.println("Type: Four-wheeler");

    }

}

class Main3 {

    public static void main(String[] args) {

        Vehicle vehicle1 = new TwoWheeler(60, "Petrol");

        vehicle1.display();

        Vehicle vehicle2 = new ThreeWheeler(50, "Diesel");

        vehicle2.display();

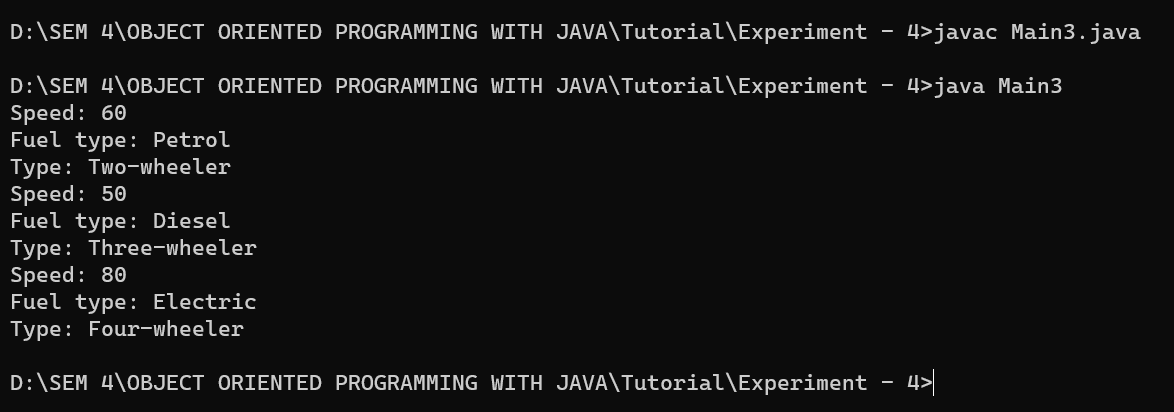
        Vehicle vehicle3 = new FourWheeler(80, "Electric");

        vehicle3.display();

    }

}

**Output Screenshot:**

****

1. Create an interface Vegetable along with method color() and grow(). Implement color and grow methods with three classes Spinach, Potato, Onion, Tomato.

**Code:**

interface Vegetable {

    public void color();

    public void grow();

}

class Spinach implements Vegetable {

    public void color() {

        System.out.println("Color of spinach is green");

    }

    public void grow() {

        System.out.println("Spinach grows above ground");

    }

}

class Potato implements Vegetable {

    public void color() {

        System.out.println("Color of Potato is browny white");

    }

    public void grow() {

        System.out.println("Potato grows under ground");

    }

}

class Onion implements Vegetable {

    public void color() {

        System.out.println("Color of Onion is red");

    }

    public void grow() {

        System.out.println("Onion grows under ground");

    }

}

class Tomato implements Vegetable {

    public void color() {

        System.out.println("Color of Tomato is red");

    }

    public void grow() {

        System.out.println("Tomato grows above ground");

    }

}

class Main4 {

    public static void main(String[] args) {

        Vegetable spinach = new Spinach();

        spinach.color();

        spinach.grow();

        Vegetable potato = new Potato();

        potato.color();

        potato.grow();

        Vegetable onion = new Onion();

        onion.color();

        onion.grow();

        Vegetable tomato = new Tomato();

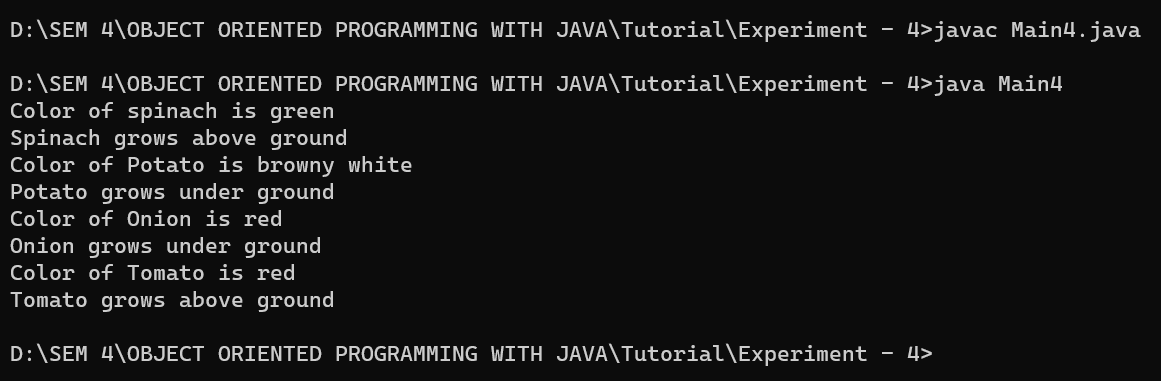
        tomato.color();

        tomato.grow();

    }

}

**Output Screenshot:**

****

1. Create an interface College with field name Name of college, Branch, Student name, Sem, with method Showdetails(). Add two class Department and Student. In Department it will display Branch along with student name by implementing Showdetails() and Student class will display College name,Student name,sem,Branch by implementing Showdetials(). Showdetials(). Enter all the details by user input only. Display all these details using Reference variable of College interface.

**Code:**

import java.util.Scanner;

interface College {

    String COLLEGE\_NAME = "XYZ College";

    String getName();

    String getBranch();

    String getStudentName();

    int getSem();

    void showDetails();

}

class Department implements College {

    private String branch;

    private String studentName;

    @Override

    public String getName() {

        return COLLEGE\_NAME;

    }

    @Override

    public String getBranch() {

        return branch;

    }

    @Override

    public String getStudentName() {

        return studentName;

    }

    @Override

    public int getSem() {

        return 0;

    }

    public void setBranch(String branch) {

*this*.branch = branch;

    }

    public void setStudentName(String studentName) {

*this*.studentName = studentName;

    }

    @Override

    public void showDetails() {

        System.out.println("Branch: " + branch);

        System.out.println("Student Name: " + studentName);

    }

}

class Student implements College {

    private String branch;

    private String studentName;

    private int sem;

    @Override

    public String getName() {

        return COLLEGE\_NAME;

    }

    @Override

    public String getBranch() {

        return branch;

    }

    @Override

    public String getStudentName() {

        return studentName;

    }

    @Override

    public int getSem() {

        return sem;

    }

    public void setBranch(String branch) {

*this*.branch = branch;

    }

    public void setStudentName(String studentName) {

*this*.studentName = studentName;

    }

    public void setSem(int sem) {

*this*.sem = sem;

    }

    @Override

    public void showDetails() {

        System.out.println("College Name: " + getName());

        System.out.println("Branch: " + branch);

        System.out.println("Student Name: " + studentName);

        System.out.println("Semester: " + sem);

    }

}

class Main5 {

    public static void main(String[] args) {

        Scanner scanner = new Scanner(System.in);

        College college;

        System.out.println("Enter 1 for Department or 2 for Student");

        int choice = scanner.nextInt();

        scanner.nextLine();

        switch (choice) {

            case 1:

                college = new Department();

                System.out.println("Enter Branch: ");

                ((Department) college).setBranch(scanner.nextLine());

                System.out.println("Enter Student Name: ");

                ((Department) college).setStudentName(scanner.nextLine());

                break;

            case 2:

                college = new Student();

                System.out.println("Enter Branch: ");

                ((Student) college).setBranch(scanner.nextLine());

                System.out.println("Enter Student Name: ");

                ((Student) college).setStudentName(scanner.nextLine());

                System.out.println("Enter Semester: ");

                ((Student) college).setSem(scanner.nextInt());

                break;

            default:

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

                return;

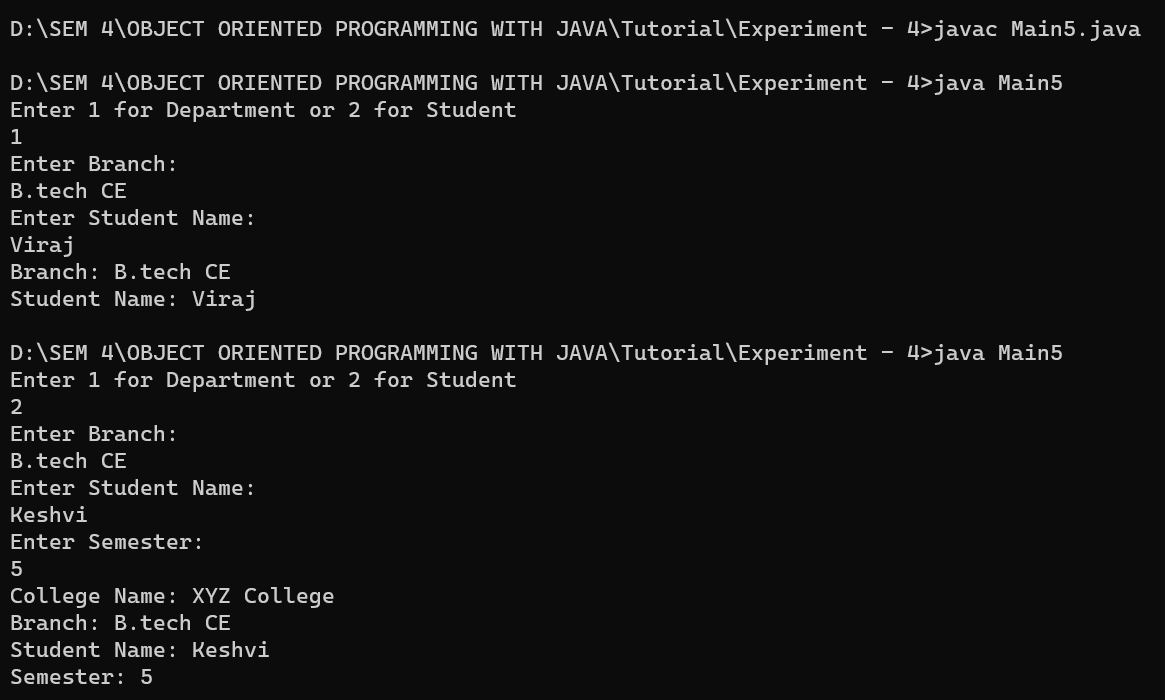
        }

        college.showDetails();

    }

}

**Output Screenshot:**

****

1. Perform following operations:

Create your own package named mypackage in file name A.java  
Define a class and add one method displayA().  
Create a new class b.java and import mypackage.  
Access displayA() methods from these mypackage in b.java file.

**Code:**

package mypackage;

public class A {

    public static void displayA() {

        System.out.println("This is method displayA() from class A in mypackage.");

    }

}

import mypackage.A;

public class B {

    public static void main(String[] args) {

        A.displayA();

    }

}

**Output Screenshot:**

