DAY 2

QUE:1

```
public class Book {

   // Instance variables
   String title;
   String author;
   int year;

   // Default constructor
   public Book() {
      title = "Not specified";
      author = "Unknown";
      year = 0;
   }

   // Parameterized constructor
   public Book(String title, String author, int year) {
      this.title = title;
      this.author = author;
      this.year = year;
   }

   // Method to display book details
   public void displayInfo() {
      System.out.println("Title: " + title);
      System.out.println("Author: " + author);
```

```
        System.out.println("Year: " + year);
        System.out.println();
    }

    // Main method
    public static void main(String[] args) {

        // Book 1 using default constructor
        System.out.println("Book 1:");
        Book book1 = new Book();
        book1.displayInfo();

        // Book 2 using parameterized constructor
        System.out.println("Book 2:");
        Book book2 = new Book("Java Programming", "Sakshi", 2024);
        book2.displayInfo();
    }
}
```

QUE:2

```
import java.util.Scanner;

public class Cube {

    int l, b, h;

    // Default constructor
    }

    Cube() {
        l = 10;
        b = 10;
        h = 10;
        double volume = 1.0 * l * b * h;   // force double
        System.out.println("Constructor without parameter");
        System.out.println("Volume is " + volume);
    }

    // Parameterized constructor
    Cube(int l, int b, int h) {
        this();  // call default constructor
        double volume = 1.0 * l * b * h;   // force double
        System.out.println("Constructor with parameter");
        System.out.println("Volume is " + volume);
    }

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

        int l = sc.nextInt();
        int b = sc.nextInt();
        int h = sc.nextInt();

        new Cube(l, b, h);
    }
}
```

QUE:3

```java
import java.util.Scanner;
import java.util.ArrayList;
import java.util.List;
class Main
{
    public static void main(String[] args)
    {
        Scanner sc = new Scanner(System.in);
        double length=sc.nextDouble();
        double width=sc.nextDouble();
        Rectangle rectangle1 = new Rectangle(length,width);
        double area1 = rectangle1.calculateArea();
        double perimeter1 = rectangle1.calculatePerimeter();
        System.out.println("Rectangle 1: ");
        System.out.println("Length: " + rectangle1.getLength());
        System.out.println("Width: " + rectangle1.getWidth());
        System.out.println("Area: " + area1);
        System.out.println("Perimeter: " + perimeter1);
    }
}
class Rectangle {
    private double length;
    private double width;
    Rectangle(double length,double width){
        this.length = length;
        this.width = width;
    }
    }
    public double getLength(){
        return length;
    }
    public double getWidth(){
        return width;
    }
    public double calculateArea(){
        return length * width;
    }
    public double calculatePerimeter(){
        return 2 * (length + width);
    }
```

QUE:4

```
public class Main {

    public static void main(String[] args) {
        Vehicle v = new Vehicle();
        v.noOfWheels(); // Vehicle method

        Scooter s = new Scooter();
        s.noOfWheels(); // Scooter method

        Car c = new Car();
        c.noOfWheels(); // Car method
    }
}
```

```
// Superclass
class Vehicle {
    void noOfWheels() {
        System.out.println("No of wheels undefined");
    }
}

// Subclass Scooter
class Scooter extends Vehicle {
    @Override
    void noOfWheels() {
        System.out.println("No of wheels 2");
    }
}

// Subclass Car
class Car extends Vehicle {
    @Override
    void noOfWheels() {
        System.out.println("No of wheels 4");
    }
}
```

QUE:5

```
import java.util.Scanner;
import java.util.*;

class Bicycle {
```

```
public int gear;

public int speed;

public Bicycle(int gear, int speed)

{

    this.gear = gear;

    this.speed = speed;

}

public void applyBrake(int decrement)

{

    speed = speed - decrement;

    if (speed < 0)

        speed = 0;

}


public void speedUp(int increment)

{

    speed = speed + increment;

}

public String toString()

{

    return ("No of gears are " + gear + "\n"

            + "speed of bicycle is " + speed);

}

public static void main(String args[])

{

    Scanner sc = new Scanner(System.in);


    int x = sc.nextInt();
```

```
        int y = sc.nextInt();

        int z = sc.nextInt();


        MountainBike mb = new MountainBike(x, y, z);

        System.out.println(mb.toString());

    }

}


class MountainBike extends Bicycle {


    public int seatHeight;


        public MountainBike(int gear, int speed, int startHeight)

    {

        super(gear, speed);

        seatHeight = startHeight;

    }


    public void setHeight(int newValue)

    {

        seatHeight = newValue;

    }


    @Override

    public String toString()

    {

        return (super.toString() + "\nseat height is " + seatHeight);

    }
```

```
}
```