**Problem Statement – 1**

/\*

Create a class named 'Shape' with a method to print "This is This is shape". Then create two other classes named 'Rectangle', 'Circle' inheriting the Shape class, both

having a method to print "This is rectangular shape" and "This is circular shape"

respectively. Create a subclass 'Square' of 'Rectangle' having a method to print

"Square is a rectangle". Now call the method of 'Shape' and 'Rectangle' class by the

object of 'Square' class.

\*/

/\*\*

 \* practical\_5\_problem\_statement\_1

 \*/

class Shape {

    void PrintMessage\_Shape() {

        System.out.println("This is Shape");

    }

}

class Rectangle extends Shape {

    void PrintMessage\_Rectangle() {

        System.out.println("This is Rectangular Shape");

    }

}

class Circle extends Shape {

    void PrintMessage\_Circle() {

        System.out.println("This is a Circle");

    }

}

class Square extends Rectangle {

    void PrintMessage\_Square() {

        System.out.println("Square is a Rectangle");

    }

}

public class practical\_5\_problem\_statement\_1 {

    public static void main(String[] args) {

        Square s = new Square();

        s.PrintMessage\_Shape();

        s.PrintMessage\_Rectangle();

    }

}

**Output:**

Text, email

Description automatically generated

**Problem Statement - 2**

/\*

Create three classes:

Class Vehicle: Vehicle Class will contain a display() function, which will say "This is a Vehicle".

Class Car: Car Class will derive the Vehicle Class and overwrite its display() function. it will say "This is a Car".

Class Bike:Bike Class will derive the Vehicle Class and overwrite its display() function. it will say "This is a Bike".

Write an application that reads an Integer N, which will denote the number of tyres in the vehicle. You have to create an object of the appropriate class according to the value of N and use it display( function.

If N = 2, Create a Bike Object.

If N = 4, Create a Car Object.

Create a Vehicle Object, otherwise.

Definition of Done:

DoD 1: Each class definition is stored in its own .java file.

DoD 2: Switch statement is used for identifying the appropriate class for which the object is

to be invoked.

\*/

/\*\*

 \* practical\_5\_problem\_statement\_2

 \*/

import java.util.Scanner;

class Vehicle {

    void Display() {

        System.out.println("This is a Vehicle");

    }

}

class Car extends Vehicle {

    void Display() {

        System.out.println("This is a Car");

    }

}

class Bike extends Vehicle {

    void Display() {

        System.out.println("This is a Bike");

    }

}

public class practical\_5\_problem\_statement\_2 {

    public static void main(String[] args) {

        Scanner sc = new Scanner(System.in);

        System.out.print("Enter the Number of Wheels: ");

        int N = sc.nextInt();

        switch (N) {

            case 2:

                Bike B1 = new Bike();

                B1.Display();

                break;

            case 4:

                Car C1 = new Car();

                C1.Display();

                break;

            default:

                System.out.println("Invalid Input!");

                break;

        }

    }

}

**Output:**

Text

Description automatically generated

**Problem Statement – 3**

/\*

Define a class Box with the following instance variables: width height and depth, all of type float. Create a new class BoxWeigth that extends Box to include weight as an instance variable. Write an application that tests the functionalities of both these classes.

Definition of Done:

DoD 1: Three java files to be defined. One for each class definition: Box, BoxWeight and BoxWeightDemo.

DoD 2: Box and BoxWeight should have three types of constructors defined: clone of an object, all dimensions specified as arguments, no argument.

DoD 3: Super is used to call base class constructors in derived class

DoD 4: Get and set functions defined as applicable in Box and BoxWeight classes.

DoD 5: Function to display volume in Box class and weight in BoxWeigth class

\*/

/\*\*

 \* practical\_5\_problem\_statement\_3

 \*/

class Box {

    float length;

    float width;

    float depth;

    public Box() {

    }

    public Box(float l, float w, float d) {

        this.length = l;

        this.width = w;

        this.depth = d;

    }

    void setlength(float len) {

        this.length = len;

    }

    void setwidth(float wid) {

        this.length = wid;

    }

    void setdep(float dep) {

        this.depth = dep;

    }

    float getlength() {

        return this.length;

    }

    float getwodth() {

        return this.width;

    }

    float getdepth() {

        return this.depth;

    }

    float volume() {

        float volume = this.length \* this.width \* this.depth;

        return volume;

    }

    class BoxWeight extends Box {

        float weight;

        void setweight(float weight) {

            this.weight = weight;

        }

        float getlength() {

            return this.weight;

        }

        float weight() {

            return weight;

        }

    }

public class practical\_5\_problem\_statement\_3 {

    public static void main(String[] args) {

    }

}

**Output:**

Text, letter

Description automatically generated