NAME: GAURAV KISHOR PATIL
DIV: 2 ROLL NO:54
BATCH: C

Experiment No. 11
Program on Abstract class and abstract methods.
Date of Performance:20/10/23
Date of Submission:23/10/23

Aim:- Program on abstract class and abstract methods.

Objective :- To implement the concept of abstract class and abstract methods through java program for calculating area the geometrical figure.

Theory :- An abstract class in Java is one that is declared with the abstract keyword. It may have both abstract and non-abstract methods(methods with bodies). An abstract is a java

modifier applicable for classes and methods in java but not for Variables. In this article, we will learn the use of abstract class in java.

What is Abstract class in Java?

Java abstract class is a class that can not be initiated by itself, it needs to be subclassed by another class to use its properties. An abstract class is declared using the "abstract" keyword in its class definition.

Illustration of Abstract class

abstract class Shape

{

   int color;

   // An abstract function

   abstract void draw();

}

In Java, the following some important observations about abstract classes are as follows:

An instance of an abstract class can not be created.

Constructors are allowed.

We can have an abstract class without any abstract method.

There can be a final method in abstract class but any abstract method in class(abstract class) can not be declared as final or in simpler terms final method can not be abstract itself as it will yield an error: "Illegal combination of modifiers: abstract and final"

We can define static methods in an abstract class

We can use the abstract keyword for declaring top-level classes (Outer class) as well as inner classes as abstract

If a class contains at least one abstract method then compulsory should declare a class as abstract

If the Child class is unable to provide implementation to all abstract methods of the Parent class then we should declare that Child class as abstract so that the next level Child class should provide implementation to the remaining abstract method

Code:-

abstract class Shape {

   public abstract void calculation();

```java
public abstract void display();

}

class Rectangle extends Shape {
    int len, bre,area;

    Rectangle(int len, int bre) {
        this.len = len;
        this.bre = bre;
    }

    public void calculation() {
      area = len * bre;

    }

    public void display() {
        System.out.println("The Area of the rectangle: " +area);
    }
}

class Circle extends Shape {
    double rad,area;

    Circle(double rad) {
        this.rad = rad;
    }

    public void calculation() {
      area = 2 * Math.PI * rad;
    }
```

```java
    public void display() {
        System.out.println("The Area of the circle: " +area);
    }
}


class Triangle extends Shape {
    double height, base,area;

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


    public void calculation() {
        area = 0.5 * base * height;
    }


    public void display() {
        System.out.println("The Area of the triangle: " +area);
    }
}

class Exp11 {
    public static void main(String[] args) {
        Rectangle r1 = new Rectangle(12, 15);
        r1.calculation();
        r1.display();

        Circle c1 = new Circle(50);
        c1.calculation();
        c1.display();
```

```
        Triangle t1 = new Triangle(12.25, 13.69);

        t1.calculation();

        t1.display();

    }

}
```

```
C:\WINDOWS\system32\cmd.    ×     +    ∨

C:\Users\GAURAV\OneDrive\Desktop>java Exp11
The Area of the rectangle: 180
The Area of the circle: 314.1592653589793
The Area of the triangle: 83.85125

C:\Users\GAURAV\OneDrive\Desktop>
```

Conclusion:

In conclusion, abstract classes in Java are a powerful feature that promotes code reusability and enforces a certain level of abstraction. They allow you to define methods that must be created within any child classes built from the abstract class. A class that contains one or more abstract methods is declared abstract itself, and cannot be instantiated. Abstract classes are used to create a common interface for its subclasses. An abstract class can have parametrized constructors and default methods as well. They are an excellent way to create planned inheritance hierarchies, especially when the full functionality of a class cannot be implemented in an interface. They're often used in combination with interfaces to make the most of each feature.