Don Bosco Institute of Technology, Kurla(W)
Department of Electronics and Tele-Communication Engineering
ECL304 - Skill Lab: C++ and Java Programming
Sem III
2021-22

Lab Number:	**9**
Student Name:	**Deep Patel**
Roll No :	**8**

Title:

1. Write a java program to create an abstract class named Shape that contains two integers and an abstract method named printArea(). Provide three classes named Rectangle, Triangle and Circle such that each one of the classes extends the class Shape. Each one of the classes contain only the method printArea() that prints the area of the given shape.

Learning Objective:

Students will be able to implement abstract class and abstract method programs.

Learning Outcome:

- Understanding the abstraction concept and hiding of the unnecessary code.

Course Outcome:

ECL304.4	1. Implement different programming applications using packaging.

Theory:

- Explain in details about necessity of data hiding in any application / project.

Encapsulation in Java is a mechanism of wrapping the data (variables) and code acting on the data (methods) together as a single unit. In encapsulation, the variables of a class will be hidden from other classes, and can be accessed only through the methods of their current class. Therefore, it is also known as data hiding.
To achieve data hiding in Java :

1. Declare the variables of a class as private.
2. Provide public setter and getter methods to modify and view the variables values.

- Explain abstract class and abstract methods.

An abstract class is a class that contains an abstract keyword, it cannot be initiated, but you can subclass it. An abstract method is a method without implementation. If a class contains at least one abstract method, it should be marked as abstract. The abstract class in Java cannot be instantiated (we cannot create objects of abstract classes). We use the abstract keyword to declare an abstract class.

Algorithm:

Step 1: Start

Step 2: Create abstract class shape.

Step 3: Define attributes and method printarea()

Step 4: Make subclass rectangle, circle and triangle of abstract class shape.

Step 5: Define their attributes and override the method for each shape with formulas of its area.

Step 6: Create main class

Step 7: Make 3 objects in main class

Step 8: Stop

Code:

```
package shape;

abstract class shape {
	int a=5,b=2;
	abstract public void PrintArea();
	}

class rectangle extends shape {
	public int area_rectangle;
	public void PrintArea() {
		area_rectangle=a*b;
		System.out.println("The area of rectangle is:"+area_rectangle);
		}
	}

class circle extends shape {
	int area_circle;
	public void PrintArea() {
		area_circle=(int) (3.14*a*a);
		System.out.println("The area of circle is:"+area_circle);
		}
}

class triangle extends shape {
	int area_triangle;
	public void PrintArea() {
		area_triangle=(int) (0.5*a*b);
		System.out.println("The area of triangle is:"+area_triangle);
		}
	}

public class javashape {
    public static void main(String[] args) {
```

```
rectangle r=new rectangle();
r.PrintArea();
circle r1=new circle();
r1.PrintArea();
triangle t=new triangle();
t.PrintArea();
}
}
```

Input Given:

a = 5 , b = 2

Output:

```
The area of rectangle is:10
The area of circle is:78
The area of triangle is:5
```