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:	**Lovely Varshney**
Roll No :	**15**

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.

Data is the most sensitive and volatile component of a program, which, if manipulated, can result in an incorrect output and harm the integrity of data. This is where data hiding proves essential. data hiding also referred to as information hiding.
Data hiding's underlying objective is to conceal data within a class from unauthorized access and avoid unnecessary penetration from outside the class. Data hiding guarantees constrained data access to ensure object integrity and prevent unintended or intended changes to the program.
Simply put, data hiding takes certain parts of a program code and screens it from object members. If an object member is trying to access hidden data, the program will return an error.

- Explain abstract class and abstract methods.

A class which is declared as abstract is known as an **abstract class**. It can have abstract and non-abstract methods. It needs to be extended and its method implemented. It cannot be instantiated. It can have constructors and static methods also. It can have final methods which will force the subclass not to change the body of the method.

A method which is declared as abstract and does not have implementation is known as an abstract method.

Algorithm :	**1. Start** **2. Create a abstract class - shape and declare necessary methods and attributes.** **3. Create a derived class of shape class – rectangle, circle , triangle and take input of dimensions and print its area** **4. Create the objects of derived classes in Main class and then call them to print the area.** **5. End**

Program:	package com.company;
	import java.util.*;
	
	abstract class Shape {
	int length, breadth, base, height,radius;
	Scanner sc = new Scanner(System.in);
	
	abstract void printArea();
	}
	
	class Rectangle extends Shape {
	void printArea() {
	System.out.print("Enter length & breadth: ");
	length = sc.nextInt();

```
        breadth = sc.nextInt();
        System.out.println("Area of Rectangle is: " + length * breadth);
    }
}

class Triangle extends Shape {
    void printArea() {
        System.out.print("Enter Base And Height: ");
        length = sc.nextInt();
        breadth = sc.nextInt();
        System.out.println("Area of Triangle is: " + (base * height) / 2);
    }
}

class Cricle extends Shape {
    void printArea() {
        System.out.print("Enter Radius: ");
        radius = sc.nextInt();
        System.out.println("Area of Cricle is: " + 3.14 * radius * radius);
    }
}

public class Main {
    public static void main(String[] args) {
        Shape rec = new Rectangle();
        rec.printArea();

        Shape tri = new Triangle();
```

	tri.printArea();
	
	Shape cri = new Cricle();
	cri.printArea();
	}
	}

Input given:	**Length – 4** **Breadth – 5** **Base – 4** **Height - 5** **Radius - 10**
Output Screenshot: