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:	**ABHISHEK MANIK WAGHMARE**
Roll No :	**01**

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 hiding is a software development technique specifically used in object-oriented programming (OOP) to hide internal object details (data members). Data hiding ensures exclusive data access to class members and protects object integrity by preventing unintended or intended changes.
Data hiding also reduces system complexity for increased robustness by limiting interdependencies between software components.
Data hiding is also known as data encapsulation or information hiding.

Explain abstract class and abstract methods.

Data **abstraction** is the process of hiding certain details and showing only essential information to the user.
Abstraction can be achieved with either **abstract classes** or **interfaces**
The `abstract` keyword is a non-access modifier, used for classes and methods:
Abstract class: is a restricted class that cannot be used to create objects (to access it, it must be inherited from another class).

Abstract method: can only be used in an abstract class, and it does not have a body. The body is provided by the subclass (inherited from).An abstract class can have both abstract and regular methods

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

```
        System.out.print("\nEnter Radius: ");
        radius = sc.nextInt();
        System.out.println("Area of Circle is: " + 3.14
* radius * radius);
    }
}

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

        Shape tri = new Triangle();
        tri.printArea();

        Shape cri = new Circle();
        cri.printArea();
    }
}
```

Output Screenshot: