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:	**Sakshi Vadiraj Kaveri**
Roll No :	**33**

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.

Information or data is very crucial resource to us. Thus securing the information becomes all the more necessary. The communication media through which we send data does not provide data security, so other methods of securing data are required. Information hiding plays a very crucial role today. It provided methods for encrypting the information so that it becomes unreadable for any unintended user. This paper reviews the techniques that exist for data hiding and how can these be combined to provide another level of security.

- Explain abstract class and abstract methods.

Abstract class:
A class which contains the abstract keyword in its declaration is known as abstract class. Abstract classes may or may not contain abstract methods i.e., methods without body (public void get();) . But, if a class has at least one abstract method, then the class must be declared abstract. If a class is declared abstract it cannot be instantiated.

To use an abstract class you have to inherit it from another class, provide implementations to the abstract methods in it. If you inherit an abstract class you have to provide implementations to all the abstract methods in it.

Abstract method:
If you want a class to contain a particular method but you want the actual implementation of that method to be determined by child classes, you can declare the method in the parent class as abstract. Abstract keyword is used to declare the method as abstract. You have to place the abstract keyword before the method name in the method declaration. An abstract method contains no method body. Instead of curly braces an abstract method will have a semicolon(;) at the end.

Declaring a method as abstract has two consequences:
- ➔ The class containing it must be declared as abstract.
- ➔ Any class inheriting the current class must either override the abstract method or declare itself as abstract.

Algorithm:

1) Create abstract class shape declare variables dim1, dim2 variables and printArea() method.
2) Inherit class Rectangle and Triangle and Circle from Shape class, and in the method printArea() take input from user in dim1, dim2. Calculate area by, area of rectangle= dim1*dim2, area of triangle= ½ *dim1*dim2 and area of circle= 3.14*dim1*dim1.
3) Display the output on the screen.
4) In the main function create the objects of the inherited classes rectangle, triangle and circle and call the method printArea().

Program:

```
/*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.
 */

package inheritance;

import java.util.Scanner;

abstract class Shape{

	Scanner in=new Scanner(System.in);
	public int dim1,dim2;
	abstract int printArea();
}
class Rectangle extends Shape{
	//Scanner in=new Scanner(System.in);
	int printArea() {
	System.out.println("Enter the length and breadth of the rectangle ");
```

```
		dim1=in.nextInt();
		dim2=in.nextInt();
		float area;
			area=dim1*dim2;
			System.out.println("The formula for area of rectangle is
(length)*(breadth)="+dim1+"*"+dim2);;
			System.out.println("The area of the rectangle is "+area);
			return 0;
		}

}
class Triangle extends Shape{
		//Scanner in=new Scanner(System.in);
		int printArea() {
		System.out.println("\nEnter the base and height of the triangle ");
		dim1=in.nextInt();
		dim2=in.nextInt();
		float area;
			area=(float) 0.5*(dim1*dim2);
			System.out.println("The formula for area of triangle is 0.5*(base)*(height)=
0.5*"+dim1+"*"+dim2);;
			System.out.println("The area of the traingle is "+area);
			return 0;
		}

}
class Circle extends Shape{
		//Scanner in=new Scanner(System.in);
		int printArea() {
		System.out.println("\nEnter the radius of the circle ");
		dim1=in.nextInt();
		//dim2=in.nextInt();
		float area;
			area= (float) 3.14*dim1*dim1;
			System.out.println("The formula for area of circle is
3.14*(radius)*(radius)=3.14*"+dim1+"*"+dim1);;
			System.out.println("The area of the circle is "+area);
			return 0;
		}

}



public class Lab9_33 {

		public static void main(String[] args) {
			Shape r=new Rectangle();
```

```
        r.printArea();
        Shape t=new Triangle();
        t.printArea();
        Shape c=new Circle();
        c.printArea();
        // TODO Auto-generated method stub

    }

}
```

INPUT GIVEN:

Length and breadth of rectangle:- 10,20
Base and height of triangle:- 10,20
Radius of the circle= 9

OUTPUT SCREENSHOT: