

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:	Rohit Gupta
Roll No :	30

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.

Solution:-

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.

Data hiding was introduced as part of the OOP methodology, in which a program is segregated into objects with specific data and functions. This technique enhances a programmer's ability to create classes with unique data sets and functions, avoiding unnecessary penetration from other program classes.

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

- Explain abstract class and abstract methods.

Solution:-

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:

```
□ abstract class Animal {  
□     public abstract void animalSound();  
□     public void sleep() {  
□         System.out.println("Zzz");  
□     }  
□ }  
□  
□
```

- From the example above, it is not possible to create an object of the Animal class:

```
□ Animal myObj = new Animal(); // will generate an error
```

Q1)

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

Output:

A screenshot of a terminal window with a black background and white text. The window title bar at the top shows a minimize icon, a maximize icon, and a close icon, followed by the text 'input'. The terminal output consists of three lines: 'Area of rectangle is :6.0', 'Area of circle is :12.56', and 'Area of triangle is :3.0'. Below these, there is a green prompt '...Program finished with exit code 0' and a green instruction 'Press ENTER to exit console.' followed by a cursor icon.

```
input
Area of rectangle is :6.0
Area of circle is :12.56
Area of triangle is :3.0

...Program finished with exit code 0
Press ENTER to exit console.
```