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:	**5**
Student Name:	**Pratham Amare**
Roll No :	**23**

Title:

4.1 Write a Java program to Create a class Student with two method getData() and printData(). getData() to get the value from the user and display the data in printData(). Create the two objects s1 ,s2 to declare and access the values from class StudentTest.

4.2 Write a Java program for Basic bank Management System

Learning Objective:

- Students will be able to write C++ and java program for using classes and objects.

Learning Outcome:

- Ability to execute a simple C++ and Java program by accepting and displaying values using functions
- Understanding the classes and objects concept in C++ and Java.

Course Outcome:

ECL304.1	Understand object-oriented programming concepts and implement using C++ and Java

Theory:

Explain about Constructor.

In Java, a constructor is a block of codes similar to the method. It is called when an instance of the class is created. At the time of calling constructor, memory for the object is allocated in the memory. It is a special type of method which is used to initialize the object.Every time an object is created using the new() keyword, at least one constructor is called.It calls a default constructor if there is no constructor available in the class. In such case, Java compiler provides a default constructor by default.There are two types of constructors in Java: no-arg constructor, and Parameterized constructor.

Explain about classes and objects in Java

Class are a blueprint or a set of instructions to build a specific type of object. It is a basic concept of Object-Oriented Programming which revolve around the real-life entities. Class in

Java determines how an ob-ject will behave and what the object will contain. Object is an instance of a class. An object in OOPS is nothing but a selfcontained component which consists of methods and properties to make a particular type of data useful. For example color name, table, bag, barking. When you send a message to an object, you are asking the object to invoke or execute one of its methods as defined in the class. From a programming point of view, an object in OOPS can include a data structure, a variable, or a function. It has a memory location allocated. Java Objects are designed as class hierarchies.

How to access class attributes and methods? Explain with example

We can access attributes and method of a class by creating an object. For ex: public class Main { int x = 5; void getvalue(); public static void main(String[] args) { Main myObj = new Main(); myObj.get(); System.out.println(myObj.x); } }

Algorithm :	**1. Start** **2. Define Class Student** **3. Define attributes – Name , Roll_no, cgpa, div , branch** **4. Define and declare method – getdata() to get input** **5. Define and declare method – printdata() to print the values** **6. Define class student test** **7. Define public static Main function()** **8. Create object s1, s2 to call the class functionality.** **9. End.**

Program :	`import java.util.Scanner;`
	`class Student {`
	
	`Scanner in=new Scanner(System.in);`
	`String name;`
	`int roll_no;`
	`float cgpa;`
	`char div;`
	`String branch;`
	
	`void getdata()`
	`{`
	`System.out.println("------------`
	`-------------------------\nEnter your name:");`

```
                name=in.next();
                System.out.println("Enter your
roll number:");
                roll_no=in.nextInt();
                System.out.println("Enter your
CGPA:");
                cgpa=in.nextFloat();
                System.out.println("Enter your
Division:");
                div=in.next().charAt(0);
                System.out.println("Enter
branch:");
                branch=in.nextLine();
                branch=in.nextLine();

            }


            void getdata(String n,int r,float
c,char d, String b)
            {
                name=n;
                roll_no=r;
                cgpa=c;
                div=d;
                branch=b;

            }

            void printdata()
            {
                System.out.println("Name of the
student: "+name);
                System.out.println("Roll-no of
the student: "+roll_no);
                System.out.println("Cgpa of the
student: "+cgpa);
                System.out.println("Division of
the student: "+div);
                System.out.println("branch of
the student: "+branch);
            }
```

	}
	class StudentTest {
	
	public static void main(String[] args) {
	
	Student s1=new Student();
	Student s2=new Student();
	s1.getdata();
	s1.printdata();
	s2.getdata();
	s2.printdata();
	
	}
	
	}

Input given:	

Output Screenshot: