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:	**4**
Student Name:	**Pooja Jagdish Verma**
Roll No :	**21**

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 java program for using classes and objects.

Learning Outcome:

• Ability to execute a simple Java program by accepting and displaying values using functions

• Understanding the classes and objects concept in Java.

Course Outcome:

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

Theory :

1. **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.

2.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 self-contained 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.

3.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);
 }
}
```

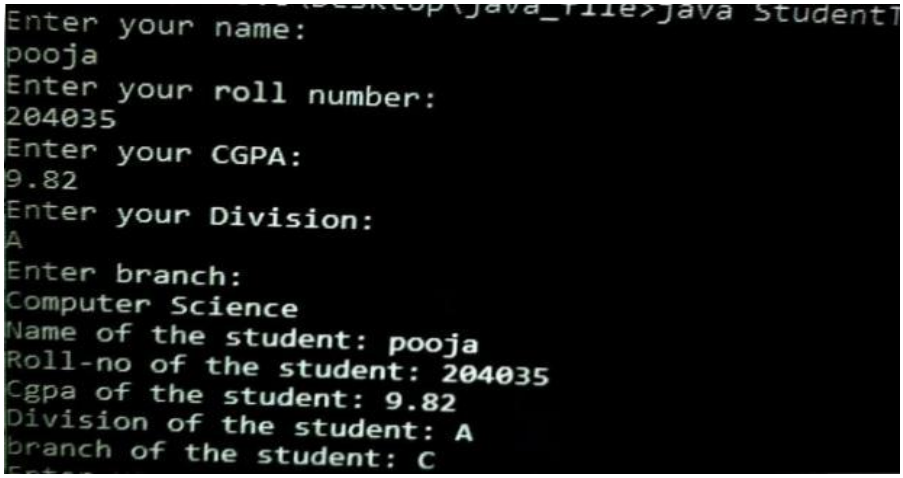
Algorithm :	1. Start 2. Define Class Student 3. Define attributes – Name , Roll_no, cgpa, div , branch 4. Define and declare method – getdata() to get input from user.

	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;
	char branch;
	void getdata()
	{
	System.out.println("Enter 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.next().charAt(0);
	}
	void getdata(String n,int r,float c,char d, char 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);
	}
	};
	public 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:	Name : Pooja Roll number : 204035

	CGPA : 9.82 Div : A Branch : Computer science Name : Pooja Roll number : 204035 CGPA : 9.82 Div : A Branch : C
Output Screenshot:	
Algorithm 2:	1. Start 2. Define Class BankLab 2 3. Define attributes – Name , account_type , account_number, amount, balance 4. Declare attributes by using constructor of class. 5. Define and declare method – deposit() to deposit the amount 6. Define and declare method – withdraw() to withdraw the amount 7. Define and declare method – display() to display the account details

	8. Define static void Main function() 9. Create object b1, b2, b3 to call the class functionality. 10. Do – while loop to repeat the process. 10. End

Problem 2:	import java.util.Scanner;
	public class BankLab2 {
	Scanner in=new Scanner(System.in);
	String name;
	char account_type;
	int account_number,amount;
	float balance;
	public BankLab2(String n,int a, char t, float b) {
	// TODO Auto-generated constructor stub
	name = n;
	account_number=a;
	account_type=t;
	balance=b;
	}
	int deposit()
	{
	System.out.println("Enter the amount to deposit: ");
	int amount=in.nextInt();
	if(amount<0)
	{
	System.out.println("Invalid amount,Enter a valid
	amount");
	return 0;
	}

```
balance=balance+amount;
return 1;
}
int withdraw()
{
System.out.println("Your Balance= " +balance );
System.out.println("Enter amount to withdraw: ");
int amount=in.nextInt();
if (balance<amount)
{
 System.out.println("Insufficient Balance: ");
return 0;
}
if(amount<0)
{
 System.out.println("Invalid amount" );
return 0;
}
balance=balance-amount;
return 1;
}
void display()
{
System.out.println("Name :"+name);
System.out.println("Account Number:"
+account_number);
System.out.println("Account Type:"
+account_type);
```

```
System.out.println("Balance: " +balance);
}
public static void main(String[] args) {
// TODO Auto-generated method stub
Scanner in=new Scanner(System.in);
BankLab2 b1=new BankLab2("salman",1,'s',0);
BankLab2 b2=new BankLab2("makarand",2,'s',0);
BankLab2 b3=new BankLab2("siddharth",3,'s',0);
System.out.println("Menu");
System.out.println("1.Deposit");
System.out.println("2.Withdraw");
System.out.println("3.Display");
 System.out.println("Enter option");
int op=in.nextInt();
char ans;
do
{
System.out.println("Please enter your account number:");
int account_number=in.nextInt();
switch(account_number)
{
case 1: if(op==1)
b1.deposit();
if(op==2)
b1.withdraw();
if(op==3)
b1.display();
break;
```

```
case 2: if(op==1)
b2.deposit();
if(op==2)
b2.withdraw();
if(op==3)
b2.display();
break;
case 3: if(op==1)
b3.deposit();
if(op==2)
b3.withdraw();
if(op==3)
b3.display();
break;
default:
System.out.println("Enter value between 1 to 3");
break;
}
System.out.println("Do you want to continue?[Y/N]");
ans=in.next().charAt(0);
if(ans=='Y' || ans == 'y')
{
System.out.println("Menu");
System.out.println("1.Deposit");
System.out.println("2.Withdraw");
System.out.println("3.Display");
 System.out.println("Enter option");
op=in.nextInt();
```

	}
	}
	while(ans!='N');
	}
	}

Input Given 2:	Option : 1 Account no : 1 Amount to deposit: 20000 Continue : y Option: 3 Account No: 1 Nmae: salman Account Number: 1 Account type:s Balnace:20000 Continue:N

Output Screenshot 2: