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:	**Sarika Laxmikant Galphade.**
Roll No :	**36**

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

CLASSES

A class is a user defined blueprint or prototype from which objects are created. It represents the set of properties or methods that are common to all objects of one type. In general, class declarations can include these components, in order:

Modifiers: A class can be public or has default access (Refer this for details).

class keyword: class keyword is used to create a class.

Class name: The name should begin with an initial letter (capitalized by convention).

Superclass(if any): The name of the class's parent (superclass), if any, preceded by the keyword extends. A class can only extend (subclass) one parent.

Interfaces(if any): A comma-separated list of interfaces implemented by the class, if any, preceded by the keyword implements. A class can implement more than one interface.

Body: The class body surrounded by braces, { }.

OBJECT

It is a basic unit of Object-Oriented Programming and represents the real life entities. A typical Java program creates many objects, which as you know, interact by invoking methods. An object consists of :

State: It is represented by attributes of an object. It also reflects the properties of an object.

Behavior: It is represented by methods of an object. It also reflects the response of an object with other objects.

Identity: It gives a unique name to an object and enables one object to interact with other objects.

How to access class attributes and methods? Explain with example

In the previous chapter, we used the term "variable" for x in the example (as shown below). It is actually an attribute of the class. Or you could say that class attributes are variables within a class

For example:

```
public class Main {
  int x = 5;
 public static void main(String[] args) {
    Main myObj = new Main();
    System.out.println(myObj.x);
```

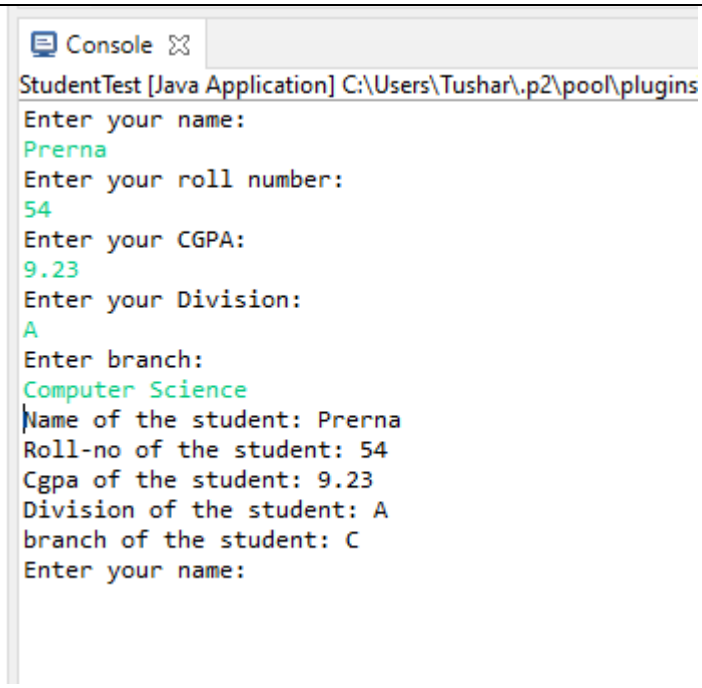
```
 }
}
```

Algorithm :	Step1: Start Step2: Define Class Student Step3: Define attributes – Name , Roll_no, cgpa, div . Step4: Define and declare method – getdata() to get input from user. Step5: Define and declare method – printdata() to print the values Step6: Define Main function() Step7: Create object s1, s2 to call the class functionality. Step8: 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:** Prerna Roll no. 54 Cgpa:9.23 Division: A

Output Screenshot:	

Algorithm 2 :	Step1: Start Step2: Define Class BankLab 2 Step3: Define attributes – Name,account type, account number ,amount ,balance. Step4: Declare attributes by using constructor of class. Step5: Define and declare method – deposit() to deposit the amount Step6: Define and declare method – withdraw() to withdraw the amount Step7: Define and declare method – display() to display the account details Step8: Define Main function() Step9: Create object b1, b2, b3 to call the class functionality. Step10: Do – while loop to repeat the process. Step11: End
Program:	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',2000);
		BankLab2 b2=new BankLab2("makarand",2,'s',2000);
		BankLab2 b3=new BankLab2("siddharth",3,'s',2000);

		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); //char input in variable ans
	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:	Enter option 3 Account no.2

Output Screenshot: