Don Bosco Institute of Technology, Kurla(W)
Department of Electronics and Tele-Communication Engineering
ECL304 - Skill Lab: C++ and Java Programming
SemIII
2021-22

Lab Number:	**4**
Student Name:	**Tejas sanjeev Rokade**
Roll No :	**43**

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 classStudentTest.

4.2 Write a Java program for Basic bank ManagementSystem

Learning Objective:

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

Learning Outcome:

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

Course Outcome:

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

Theory:

Q1. Explain about Constructor.

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. Q2. Explain about classes and objects in Java

Faculty: Ms. Deepali Kayande

CLASS: It 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 in order:

1. Modifiers: A class can be public or has default access class keyword: class keyword is used to create aclass.
2. Class name: The name should begin with an initial letter (capitalized byconvention).
3. Superclass (if any): The name of the class's parent (superclass), if any, preceded by the keyword extends. A class can only extend (subclass) oneparent.
4. Interfaces (if any): A comma-separated list of interfaces implemented by the class, if any, precededbythekeywordimplements.Aclasscanimplementmorethanoneinterface.
5. Body: The class body surrounded by braces, {}.

OBJECTS: 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 basically consists state, behaviour, identity.

Q3.How to access class attributes and methods? Explain with example.

You can access attributes by creating an object of the class, and by using the dot syntax (.)In the example we will create an object of the Main class, with the name myObj. We use the x attribute on the object to print its value.

EG. Create an object called "myObj" and print the value of x:

```
public class Main
 { int x = 5;


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

Or override existing values:

EG. Change the value of

x=25

```
public class Main {
 int x = 10;
```

```
 public static void main(String[]
  args) { Main myObj = new Main();
  myObj.x = 25; // x is now
  25
  System.out.println(myObj.x
  );
 }
```

If you don't want the ability to override existing values, declare the attribute as final:

```
public class Main
 { final int x = 10;

 public static void main(String[]
  args) { Main myObj = new Main();
  myObj.x = 25; // will generate an error: cannot assign a value to a final
  variable System.out.println(myObj.x);
 }
}
```

METHODS: Methods define behaviour of a class. A method contains business logic which is executed when the method is invoked. Methods are the ways to manipulate objects data. Let's take a look at the below example

Syntax of a method :

<access-modifier> <return-type> <name-of-the-method> ({optional}<type-of-parameter> <name-of- the-parameter>){ //method logic //method logic }

example: public void displayPlayerInfo()

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 thedata

in printData(). Create the two objects s1 ,s2 to declare and access the values from class StudentTest.

Algorithm :	STEP 1. Start STEP 2. Define Class Student STEP 3. Define attributes – Name , Roll_no, cgpa, div , branch STEP 4. Define and declare method – getdata() to get input fromuser. STEP 5. Define and declare method – printdata() to print the values STEP 6. Define Mainfunction() STEP 7. Create object s1, s2 to call the class functionality. STEP 8. Print result STEP 9. End.

Program:	`importjava.util.Scanner;`
	`classStudent`
	`{`
	`Scannerin=newScanner(System.in);`
	`Stringname;`
	`introllno;`
	`Stringdepartment;`
	`floatcgpa;`
	
	`//method overloading`
	`voidgetData()`
	`{`
	`Scannert=newScanner(System.in);`
	`System.out.println("Studentname=");`
	`name=t.next();`
	`System.out.println("Studentrollno=");`
	`rollno=t.nextInt();`
	`System.out.println("Studentdepartment=");`
	`department=t.next();`
	`System.out.println("Studentcgpa=");`
	`cgpa=t.nextFloat();`
	`}`
	
	`voidgetdata(Stringn,intr,Stringd,floatc)`
	`{`
	`name=n;`
	`rollno=r;`
	`department=d;`
	`cgpa=c;`
	`}`
	
	`voidprintdata()`

```
 {
        System.out.println("Studentname="+name);
        System.out.println("Studentrollno="+rollno);
          System.out.println("Studentdepartment="+department);
        System.out.println("Studentcgpa="+cgpa);
 }
};
publicclassStudentTest
{
publicstaticvoidmain(Stringargs[])
{
Students1=newStudent();
Students2=newStudent();

s1.getData();//nonparameter
s1.printdata();

s2.getdata("ram",25,"EXTC",(float)9.8);
s2.printdata();

}
}
```

Input given:

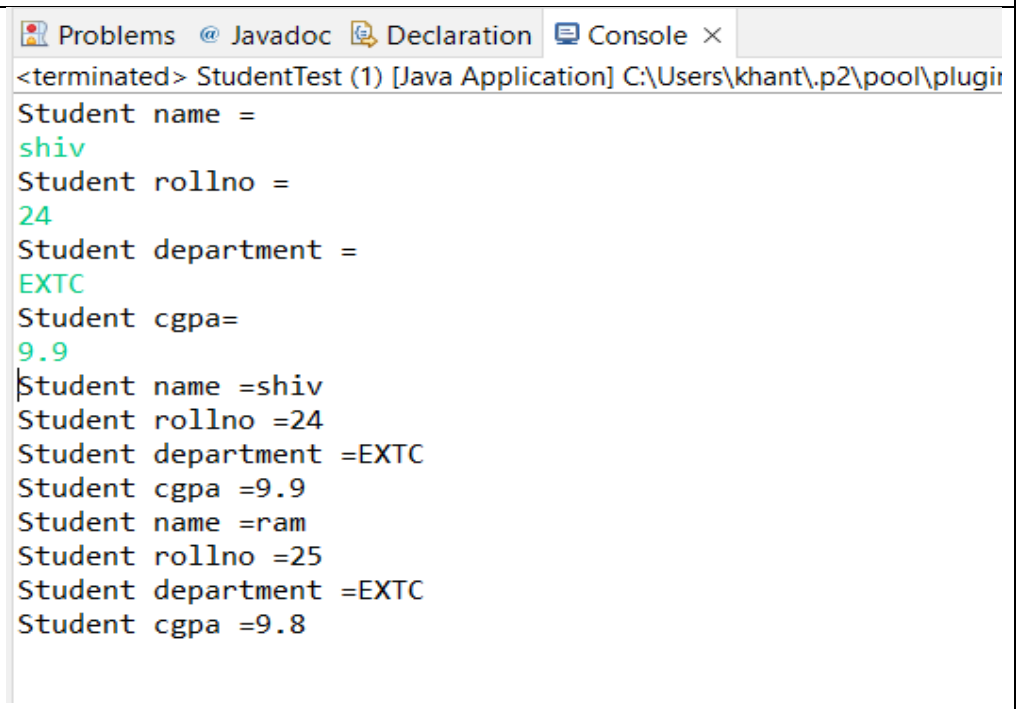
Student name= shiv

Student rollno=24

Student department= EXTC

Student cgpa=9.9

Output Screenshot:



2. Write a Java program for Basic bank ManagementSystem

Algorithm :

STEP 1. Start

STEP 2. Define Class BankLab 2

STEP 3. Define attributes – Name , account_type , account_number, amount, balance \

STEP 4. Declare attributes by using constructor of class.

STEP 5. Define and declare method – deposit() to deposit the amount

STEP 6. Define and declare methods – withdraw() to withdraw the amount

STEP 7. Define and declare methods – display() to display the account details

STEP 8. Define Main function()

STEP 9. Create object b1, b2, b3 to call the class functionality.

STEP 10. Do – while loop to repeat the process.

STEP 11. Print results

STEP 12. end

Program:

```
importjava.util.Scanner;


publicclassBankLab2{


        Scannerin=newScanner(System.in);

        Stringname;

        characcount_type;

        intaccount_number,amount;

        floatbalance;


        publicBankLab2(Stringn,inta,chart,floatb){

                //TODOAuto-generatedconstructorstub

                name=n;
```

```
                account_number=a;
                account_type=t;
                balance=b;

        }

        intdeposit()
        {
                System.out.println("Entertheamounttodeposit:
");
                intamount=in.nextInt();
                if(amount<0)
                {
                        System.out.println("Invalidamount,Entera
validamount");
                        return 0;
                }
                balance=balance+amount;
                return 1;
        }

        intwithdraw()
        {
                System.out.println("YourBalance="        +balance
);
                System.out.println("Enteramounttowithdraw:
");
                intamount=in.nextInt();
                if(balance<amount)
```

```
                {
                        System.out.println("InsufficientBalance:
        ");
                        return 0;
                }
                if(amount<0)
                {
                        System.out.println("Invalidamount");
                        return 0;
                }
                balance=balance-amount;
                return 1;
        }

        voiddisplay()
        {
                System.out.println("Name:"+name);
                System.out.println("AccountNumber:"
+account_number);
                System.out.println("AccountType:"
+account_type);
                System.out.println("Balance:"+balance);
        }

        publicstaticvoidmain(String[]args){
                //TODOAuto-generatedmethodstub
                Scannerin=newScanner(System.in);
                BankLab2b1=newBankLab2("salman",1,'s',2000);
                BankLab2b2=newBankLab2("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("Enteroption");

                intop=in.nextInt();

                char ans;

                do

                {

                        System.out.println("Pleaseenteryour
accountnumber:");

                        intaccount_number=in.nextInt();

                                switch(account_number)

                                {

                                        case1:          if(op==1)

        b1.deposit();

                                                        if(op==2)

        b1.withdraw();

                                                        if(op==3)

        b1.display();


                                                         break;

                                        case2:          if(op==1)

        b2.deposit();
```

```
                                        if(op==2)
        b2.withdraw();
                                        if(op==3)
        b2.display();
                                        break;
                                case3:  if(op==1)
        b3.deposit();
                                        if(op==2)
        b3.withdraw();
                                        if(op==3)
        b3.display();
                                        break;
                                default:
System.out.println("Entervaluebetween1to3");
                                        break;
                                }
                                System.out.println("Doyouwantto
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:	Entered option=2 Entered account number=2 Amount to withdraw=100 Continue Entered option=2 Entered account number=2

Output Screenshot: