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:	**Deep Patel**
Roll No :	**8**

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:

(1) Explain about Constructor.

A constructor in Java is a special method that is used to initialize objects. The constructor is called when an object of a class is created. It can be used to set initial values for object attributes.

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.

There are three rules defined for the constructor.

1. Constructor name must be the same as its class name
2. A Constructor must have no explicit return type

3. A Java constructor cannot be abstract, static, final, and synchronized

There are two types of constructors in Java:

1. Default constructor (no-argument constructor)
2. Parameterized constructor

(2) Explain about classes and objects in Java

Class:

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.

Object:

An entity that has state and behaviour is known as an object e.g., chair, bike, marker, pen, table, car, etc. It can be physical or logical (tangible and intangible). The example of an intangible object is the banking system.

An object is an instance of a class. A class is a template or blueprint from which objects are created. So, an object is the instance(result) of a class. Object can be defined as:

- An object is a real-world entity.
- An object is a runtime entity.
- The object is an entity which has state and behaviour.
- The object is an instance of a class.

When an object of a class is created, the class is said to be instantiated. All the instances share the attributes and the behaviour of the class. But the values of those attributes, i.e. the state are unique for each object. A single class may have any number of instances.

So in software development, methods operate on the internal state of an object and the object-to-object communication is done via methods.

(3) How to access class attributes and methods? Explain with example

To access class attributes:

We used the term "variable" for x in the any example, It is actually an attribute of the class. Or we could say that class attributes are variables within a class, another term for class attributes is fields.

We can access attributes by creating an object of the class, and by using the dot syntax (.), If you don't want the ability to override existing values, declare the attribute as final.

To access class methods:

myMethod() prints a text (the action), when it is called. To call a method, write the method's name followed by two parentheses () and a semicolon; we will often see Java programs that have either static or public attributes and methods. We created a static method, which means that it can be accessed without creating an object of the class, unlike public, which can only be accessed by objects.

PROGRAM 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.

Algorithm:

Step 1: START

Step 2: Create class Student; with parameters which are needed as roll_no, name, branch, cgpa etc.

Step 3: Give two methods as getdata(), and printdata() in a class student.

Step 4: In main function create two objects s1,s2 to declare access values from class student.

Step 5: END.

PROGRAM:

```
package extc;

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();
        }
}
```

OUTPUT:

CPU Time: sec(s), Memory: kilobyte(s) compiled and executed in 15.38 sec(s)

```
Enter your name:
Deep Patel
Enter your roll number:
8
Enter your CGPA:
8.5
Enter your Division:
B
Enter your branch:
EXTC
Name of the student: Deep Patel
Rollno of thew student: 8
CGPA of the student: 8.5
Division of the student: B
Branch of the student: EXTC
Enter your name:
Deep
Enter your roll number:
8
Enter your CGPA:
8
Enter your Division:
B
Enter your branch:
EXTC
Name of the student: Deep
Rollno of thew student: 8
CGPA of the student: 8
Division of the student: B
Branch of the student: EXTC
```

PROGRAM 2: Write a Java program for Basic bank Management System.

Algorithm:

Step 1: START

Step 2: Create class bankcode, declare parameters which are needed as name, account number etc.

Step 3: Create Constructor, and functions like deposit(), withdraw(), display().

Step 4: In main function declare objects, and add do-while loop and in it use switch case to add input from user, add input option from given 3.

Step 5: END.

PROGRAM:

```
package extc;
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) {
		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) {
        try (
        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');
        }
    }
}
```

OUTPUT:

```
Menu
1.Deposit
2.Withdraw
3.Display
Enter option
3
Please enter your account number:
1
Name :salman
Account Number:1
Account Type:s
Balance: 2000.0
Do you want to continue?[Y/N]
y
Menu
1.Deposit
2.Withdraw
3.Display
Enter option
3
Please enter your account number:
3
Name :siddharth
Account Number:3
Account Type:s
Balance: 2000.0
Do you want to continue?[Y/N]
```