

**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**

<b>Lab Number:</b>	<b>3</b>
<b>Student Name:</b>	<b>Russel D'mello</b>
<b>Roll No :</b>	<b>32</b>

**Title:**

3.1 Write a C++ 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.

3.2 Write a C++ 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:**

<b>ECL304.1</b>	Understand object-oriented programming concepts and implement using C++ and Java
-----------------	--

**Theory:**

**Difference between procedural and object oriented language**

In procedural programming, program is divided into small parts called functions.

In OOP, program is divided into small parts called objects.

Procedural programming follows top down approach.

OOP follows bottom up approach.

There is no access specifier in procedural programming.

OOP have access specifier like private, public, protected etc.

Adding new data and function in Procedural Programming is not easy.

Adding new function and data in OOP is much easy.

**Faculty: Ms. Deepali Kayande**

**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**

### **Application of object orientation**

OOP is used in manufacturing and designing applications, as it allows people to reduce the effort involved. For instance, it can be used while designing blueprints and flowcharts.

Other applications of OOP are:-

- Office automation system.
- Neural networks and parallel programming.
- AI and expert systems.

### **Brief introduction to C++ and Java**

#### **C++**

Object-oriented programming was developed since drawbacks were discovered in languages such as C, Simula, ALGOL, BPCL etc. Bjarne Stroustrup, a Danish computer scientist began working on C++ in 1979. At that time these languages already existed and had varied uses. His idea was to use C as a base and include features of Simula (An OOP language) that would promote features such as code reusability, general level abstraction etc. According to Stroustrup the '++' in C++ (post increment operator) symbolizes the evolutionary changes made to C.

In object-oriented programming (OOP) the logical arrangement of the code is changed. Instead of storing functions into different files, functionality is tightly grouped with the type that it operates on. The OOP style groups all the operations together according to what they operate on. It no longer requires any special skill to keep them separate. The language itself makes it most convenient to proceed. The strength of OOP helps the most when writing large programs, in teams and while packaging the code into libraries for use by others.

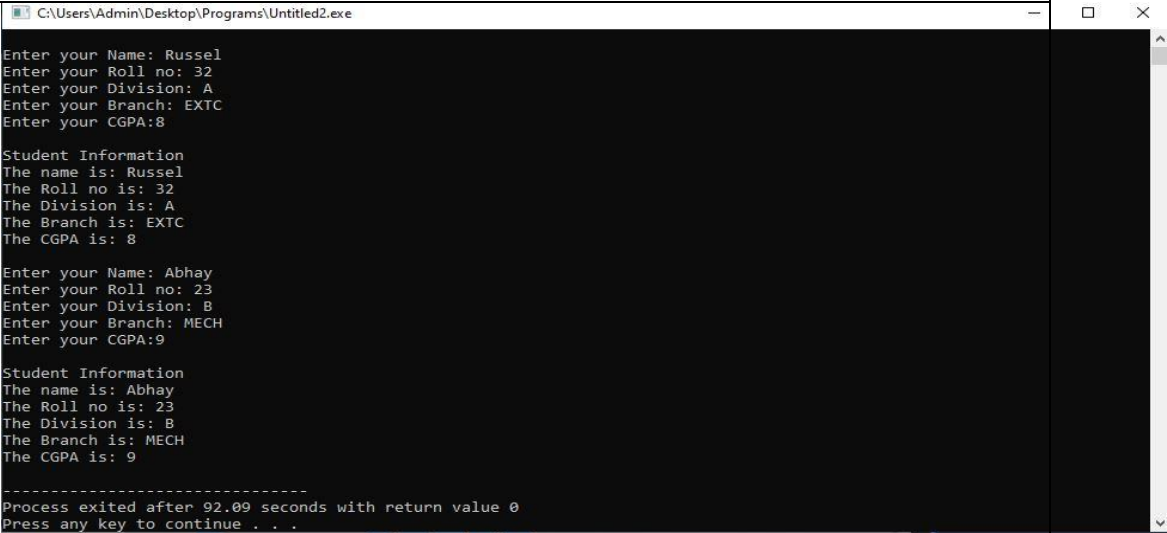
#### **JAVA**

JAVA was developed by James Gosling at Sun Microsystems Inc in the year 1991, later acquired by Oracle Corporation. It is a simple programming language. Java makes writing, compiling, and debugging programming easy. It helps to create reusable code and modular programs.

Java is a class-based, object-oriented programming language and is designed to have as few implementation dependencies as possible. A general-purpose programming language made for developers to write once run anywhere that is compiled Java code can run on all platforms

**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**

that support Java. Java applications are compiled to byte code that can run on any Java Virtual Machine. The syntax of Java is similar to c/c++.

<b>Algorithm m :</b>	<p>STEP 1. Start</p> <p>STEP 2. Define Class Student</p> <p>STEP 3. Define attributes – Name , Roll_no, cgpa, div , branch</p> <p>STEP 4. Define and declare method – getdata() to get input from user.</p> <p>STEP 5. Define and declare method – printdata() to print the values</p> <p>STEP 6. Define Main function()</p> <p>STEP 7. Create object s1, s2 to call the class functionality.</p> <p>STEP 8. Print result</p> <p>STEP 9. End.</p>										
<b>Program:</b>	<a href="https://github.com/russ070/Skill-lab-with-OOPM/blob/main/32_Lab3.cpp">https://github.com/russ070/Skill-lab-with-OOPM/blob/main/32_Lab3.cpp</a>										
<b>Input given:</b>	<table border="0"> <tr> <td>Russel</td> <td>Abhay</td> </tr> <tr> <td>32</td> <td>23</td> </tr> <tr> <td>A</td> <td>B</td> </tr> <tr> <td>EXTC</td> <td>Mech</td> </tr> <tr> <td>8</td> <td>9</td> </tr> </table>	Russel	Abhay	32	23	A	B	EXTC	Mech	8	9
Russel	Abhay										
32	23										
A	B										
EXTC	Mech										
8	9										
<b>Output Screenshots:</b>	 <p>The screenshot shows a Windows command prompt window titled 'C:\Users\Admin\Desktop\Programs\Untitled2.exe'. It displays the execution of a C++ program that takes student details as input and prints them. The input for the first student (Russel) is: Name: Russel, Roll no: 32, Division: A, Branch: EXTC, CGPA: 8. The output for Russel is: Student Information, The name is: Russel, The Roll no is: 32, The Division is: A, The Branch is: EXTC, The CGPA is: 8. The input for the second student (Abhay) is: Name: Abhay, Roll no: 23, Division: B, Branch: MECH, CGPA: 9. The output for Abhay is: Student Information, The name is: Abhay, The Roll no is: 23, The Division is: B, The Branch is: MECH, The CGPA is: 9. The program ends with 'Process exited after 92.09 seconds with return value 0' and 'Press any key to continue . . .'</p>										

<b>Algorithm m:</b>	STEP 1. Start
---------------------	---------------

**Faculty: Ms. Deepali Kayande**

**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**

	<p>STEP 2. Define Class BankLab 2</p> <p>STEP 3. Define attributes – Name , account_type , account_number, amount, balance.</p> <p>STEP 4. Declare attributes by using constructor of class.</p> <p>STEP 5. Define and declare method – deposit() to deposit the amount</p> <p>STEP 6. Define and declare methods – withdraw() to withdraw the amount</p> <p>STEP 7. Define and declare methods – display() to display the account details</p> <p>STEP 8. Define Main function()</p> <p>STEP 9. Create object b1, b2, b3 to call the class functionality.</p> <p>STEP 10. Do – while loop to repeat the process.</p> <p>STEP 11. Print result</p> <p>STEP 12. end</p>
<b>Program:</b>	<a href="https://github.com/russ070/Skill-lab-with-OOPM/blob/main/32_Lab3.cpp">https://github.com/russ070/Skill-lab-with-OOPM/blob/main/32_Lab3.cpp</a>
<b>Input:</b>	<p>1</p> <p>1</p> <p>Y</p> <p>3</p> <p>1</p> <p>N</p>

**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**

**Screenshot:**

```
C:\Users\Admin\Desktop\Programs\Untitled24.exe
Menu
1.Deposit
2.Withdraw
3.Display
Enter option
1
Please enter your account number:
1
Enter the amount to deposit: 1000
Do you want to continue?[Y/N]Y
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: 3000
Do you want to continue?[Y/N]N
-----
Process exited after 21.32 seconds with return value 0
Press any key to continue . . .
```

**Faculty: Ms. Deepali Kayande**