**LIBRARY MANAGEMENT SYSTEM**

**Submitted in partial fulfillment of the requirement for the Internal Continuous Assessment in the Subject “Object Oriented Programming” in Second Year B.Tech (Information Technology) Semester-IV**

**Submitted To**



**SVKM’s NMIMS,**

**Mukesh Patel School of Technology Management & Engineering,**

**Shirpur Campus (M.H.)**

**Submitted By :**

**PRATYUSH AVI Enrollment No–70011118004**

**Under The Supervision of:**

**RITESH DHANARE**

**(Department of Information Technology, Assistant Professor)**

**DEPARTMENT OF INFORMATION TECHNOLOGY**

**Mukesh Patel School of Technology Management & Engineering**

**SESSION: 2019-20**

### CERTIFICATE

This is to certify that the work embodies in this Mini-Project entitled **“Library Management System”** being submitted by

“**PRATYUSH AVI” (Roll No.: A203)**

**-----------**

**-------------**

**----------------------**

**For the Internal Continuous Assessment in the Subject “Object Oriented Programming” in Second Year B.Tech(Information Technology) Semester-IV d**uring the academic year 2019-20**at “Department of Information Technology”, MPSTME, Shirpur (M.H.).**

**APPROVED & SUPERVISED BY:**

**Ass. Prof Ritesh Dhanare**

**(Department of Information Technology)**

**FORWARDED BY**

# **Dr. N. S. Choubey**

H.O.D., IT

MPSTME, Shirpur Campus

**DEPARTMENT OF Information Technology (14pt)**

**Mukesh Patel School of Technology Management & Engineering (12)**

### DECLARATION

### We,

### PRATYUSH AVI

### The students of B.Tech.(Information Technology) Second Year, Session: 2019-20, MPSTME, Shirpur Campus, hereby declare that the work presented in this Project entitled “Library Management System” is the outcome of our work, is bonafide and correct to the best of our knowledge and this work has been carried out taking care of Engineering Ethics. The work presented does not infringe any patented work and has not been submitted to any other university or anywhere else for the award of any degree or any professional diploma. [14pt]

**(Pratyush Avi)**

#### SAP ID.: 70011118004

**Date: 10/04/2020**

**DEPARTMENT OF INFORMATION TECHNOLOGY**

**Mukesh Patel School of Technology Management & Engineering**

**ACKNOWLEDGEMENT**

After the completion of this Mini-Project work, words are not enough to express my feelings about all those who helped me to reach my goal; feeling above this is my indebtedness to The Almighty for providing me this moment in life.

It’s a great pleasure and moment of immense satisfaction for me to express my profound gratitude to Prof. **Ritesh Dhanare sir**, **Assistant Professor**, Information Technology Department, MPSTME, Shirpur, whose constant encouragement enabled me to work enthusiastically. Their perpetual motivation, patience and excellent expertise in discussion during progress of the project work have benefited me to an extent, which is beyond expression. Their depth and breadth of knowledge in the field made me realize that theoretical knowledge always helps to develop efficient operational software, which is a blend of all core subjects of the field. I am highly indebted to them for their invaluable guidance and ever-ready support in the successful completion of this project in time. Working under their guidance has been a fruitful and unforgettable experience.

We express my sincere thanks and gratitude to Dr. N. S.Choubey, Head of Department, Information Technology Department, MPSTME, Shirpur, for providing necessary infrastructure and help to complete the project work successfully.

# We also extend my deepest gratitude to Dr. R.S. Gaud, Director, Shirpur Campus and Dr. Nikhlesh Kumar Sharma, Director (Engineering Programs), Shirpur Campus for providing all the necessary facilities and true encouraging environment to bring out the best of my endeavors.

We would like to acknowledge all my friends, who have contributed directly or indirectly in this Mini Project work.

The successful completion of a Mini Project is generally not an individual effort. It is an outcome of the cumulative effort of a number of persons, each having their own importance to the objective. This section is a vote of thanks and gratitude towards all those persons who have directly or indirectly contributed in their own special way towards the completion of this project.

PRATYUSH AVI

**TABLE OF CONTENTS**

|  |  |  |
| --- | --- | --- |
| **Sr. No.** | **Chapter No.** | **Page** |
| 1 | Introduction | 6 |
| 2 | Problem Statement | 7 |
| 3 | Objectives | 8 |
| 4 | Project Description | 9 |
| 5 | Class Diagram | 10 |
| 6 | Test Cases | 11 |
| 7 | Source Code | 14 |
| 8 | Output Screen | 19 |
| 9 | User Manual | 21 |
| 10 | Conclusion | 26 |
| 11 | References | 27 |

**INTRODUCTION**

Library is a very important part of student’s life it is a curated collection of sources of information and similar resources. A Library Management System is a project that has been developed to handle basic housekeeping functions of a library.

It’s a well organized solution for a library. It helps to provide information on any book present in library to the user.

It keeps a record of book issued and books added to library.

**PROBLEM STATEMENT**

Based on the observation, there isn’t a proper library management system for the students of NMIMS. So, students need to go to the library to check whether a certain book is available or not. They can register themselves to the library from this only they need not go to library for this sole purpose.

**OBJECTIVE**

* It helps in maintaining a track of records of all students and their complete details
* It manages the record of book and all the details of book
* It provides timely access to request materials

**PROJECT DESCRIPTION**

Library Management System which is used to manage the record of books and students.

The project is a menu driven program as when we run the main class we get seven options that will be displayed as:

1. Add new Book.
2. Increase Quantity of a Book.
3. Show All Books.
4. Register Student.
5. Show All Registered Students.
6. Issue Book.
7. Exit

When we select the 1st option i.e. Add new book we can add a new book in the record of library with details such as book name, author name, SNo and Quantity.

When we select 2nd option i.e. Increase quantity of books we can increase the quantity of any book which is present in the record.

When we select 3rd option i.e. Show all books we will see all the books present in the library with all their details.

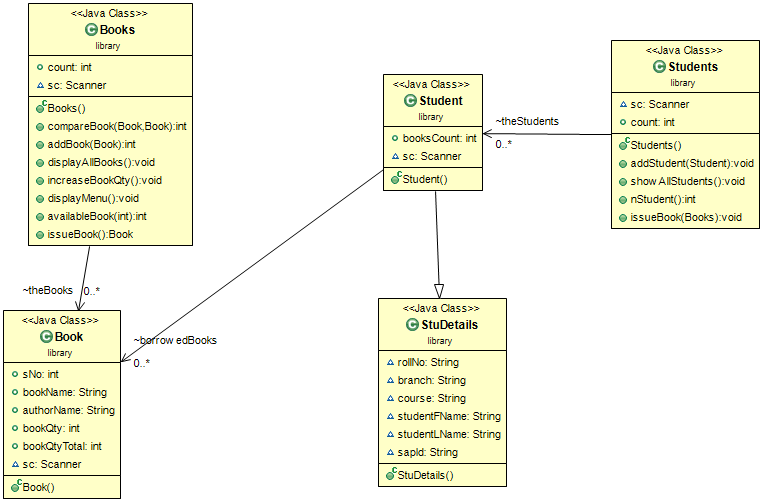
When we select 4th option i.e. Register Student we can register a student in the record by entering the details of students such as First Name, Last Name, Sap ID, Course, Batch etc.

When we select 5th option i.e. Show all registered students we will see all the students registered in the record with their details.

When we select 6th option i.e. Issue Book only the registered student can issue a book from the library and if the student is not registered then he’ll have to first register himself then only he’ll be able to register book.

When we select 7th option i.e. Exit we will be exited out of the program.

**CLASS DIAGRAM**

****

**TEST CASES**

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| **Test Case** | **Test Scenario** | **Test Description** | **Test Step** | **Test Data** | **Result** | Status |
| LMS\_1\_1 | To verify to add books | To add name of book and author and s.no | 1-Add serial No  2-Add Book name  3-Add Author | Sr no-001  Book name-Goosebumps  Author-R.L.Stine | Details are added | Successful |
| LMS\_1\_2 | To Verify is Same book is added or not | To add same detail as above in Case LMS\_1\_1 | 1-Add serial No  2-Add Book name  3-Add Author | Sr no-001  Book name- Goosebumps  Author- R.L.Stine | Book of this Name Already Exists.  (Details Not Added) | Successful |
| LMS\_1\_3 | To verify same Sr no but different book name, author | To add serial No in above case LMS\_1\_1 | 1-Add serial No  2-Add Book name  3-Add Author | Sr no-001  Book name-Ulysses  Author-James | Book of this Serial No Already Exists.  (Details Not added) | Successful |
| LMS\_1\_4 | To verify same book different sr no | To add same book in case LMS\_1\_1 | 1-Add serial No  2-Add Book name  3-Add Author | Sr no-002  Book name-Goosebumps  Author-R.L.Stine | Book of this Name Already Exists.  (Details not added) | Successful |
| LMS\_2\_1 | To Verify to upgrade books | To Add Sr no then add qty | 1-add sr no of book  2-Add qty of book | Sr no-001  Add Qty-2 | Added | Successful |
| LMS\_2\_2 | To verify non existing sr no is considered or not | To add invalid sr no | 1-add sr no of book | Sr no-003 | Not Existed | Successful |
| LMS\_3 | To show all Books | To show books in library | 1-press 3 in output and enter | Press 3 in output | Books Displayed | Successful |
| LMS\_4\_1 | To register Student details | To add details of student | 1-Stu name  2-Sap ID  3-Course  4-Branch  5-Roll No | 1-Raghav  2-Punglia  3-70011118046  4-Btech  5-IT  6-A238 | Details  Added | Successful |
| LMS\_4\_2 | To verify same Sap ID but different name | To add Details of student | 1-Stu name  2-Sap ID  3-Course  4-Branch  5-Roll No | 1-Maaz  2-Ansari  3-70011118046  4-Btech  5-textile  6-A201 | Student of SAP ID 70011118046 is Already Registered.  (Not Added) | Successful |
| LMS\_5 | To show all registered Student Details | To display the detail of student registered | Press 5 in output | Press 5 in output | Details Showed | Successful |
| LMS\_6\_1 | To check out the books | To verify the SapID and enter sno of book | 1-Sap ID  2-Enter Sr no | 1-70011118046  2-001 | Book is Available.  Issuing book....  adding book to your account | Successful |
| LMS\_6\_2 | To check out with to invalid sap id | To enter invalid Sap id | 1-Sap ID  2-Enter S no | 1-70011118010 | Sorry, You are not Registered!!!  Please Register yourself first!!! | Successful |
| LMS\_6\_3 | To check out with invalid sr no | To enter valid Sap Id and Invalid Sr No | 1-Sap ID  2Enter Sr no | 1-70011118046  2-004 | No Book of Serial Number. Available in Library. | Successful |

**SOURCE CODE**

Main Class

**package** library;

**import** java.util.Scanner;

**public** **class** Library {

**public** **static** **void** main(String[] args) {

Scanner sc = **new** Scanner(System.***in***);

System.***out***.println("\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*SVKM'S NMIMS Library!!!\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*");

System.***out***.println(" Please Select From The Following Options: ");

Books ob = **new** Books();

Students objStudent = **new** Students();

**int** choice;

**int** searchChoice;

**do**{

ob.displayMenu();

choice = sc.nextInt();

**switch**(choice){

**case** 0:

System.***out***.println("You have successfully exited the application!!!");

**break**;

**case** 1:

Book b = **new** Book();

ob.addBook(b);

**break**;

**case** 2:

ob.increaseBookQty();

**break**;

**case** 3:

ob.displayAllBooks();

**break**;

**case** 4:

Student s = **new** Student();

objStudent.addStudent(s);

**break**;

**case** 5:

objStudent.showAllStudents();

**break**;

**case** 6:

objStudent.issueBook(ob);

**break**;

**default**:

System.***out***.println("CHOICE SHOULD BE BETWEEN 0 TO 6.");

}

}

**while** (choice!=0);

}

}

Book Class:

**package** library;

**import** java.util.Scanner;

**public** **class** Book

{

**public** **int** sNo;

**public** String bookName;

**public** String authorName;

**public** **int** bookQty;

**public** **int** bookQtyTotal;

Scanner sc = **new** Scanner(System.***in***);

**public** Book()

{

System.***out***.println("Enter Serial No of Book:");

sNo = sc.nextInt();

System.***out***.println("Enter Book Name:");

bookName = sc.next();

System.***out***.println("Enter Author Name:");

authorName = sc.next();

System.***out***.println("Enter Quantity of Books:");

bookQty = sc.nextInt();

bookQtyTotal = bookQty;

}

}

Student Class:

**package** library;

**import** java.util.Scanner;

**public** **class** Student **extends** StuDetails {

Book borrowedBooks[] = **new** Book[15];

**public** **int** booksCount = 0;

Scanner sc = **new** Scanner(System.***in***);

**public** Student()

{

System.***out***.println("Enter First Name:");

studentFName = sc.next();

System.***out***.println("Enter Last Name:");

studentLName = sc.next();

System.***out***.println("Enter Sap ID:");

sapId = sc.next();

System.***out***.println("Enter Course:");

course=sc.next();

System.***out***.println("Enter Branch:");

branch=sc.next();

System.***out***.println("Enter Roll No.:");

rollNo=sc.next();

}

}

StuDetails Class:

**package** library;

**public** **class** StuDetails

{

String rollNo;

String branch;

String course;

String studentFName;

String studentLName;

String sapId;

}

Students Class:

**package** library;

**import** java.util.Scanner;

**public** **class** Students {

Scanner sc = **new** Scanner(System.***in***);

Student theStudents[] = **new** Student[50];

**public** **int** count = 0;

**public** **void** addStudent(Student s)

{

**for** (**int** i=0; i<count; i++){

**if**(s.sapId.equalsIgnoreCase(theStudents[i].sapId)){

System.***out***.println("Student of SapID " + s.sapId + " is Already Registered.");

**return**;

}

}

**if** (count<=15){

theStudents[count] = s;

count++;

}

}

**public** **void** showAllStudents(){

System.***out***.println("First Name\t\tLast Name\t\tReg Number\t\tCourse\t\tBranch\t\tRollNo.");

**for** (**int** i=0; i<count; i++){

System.***out***.println(theStudents[i].studentFName + "\t\t" + theStudents[i].studentLName + "\t\t" + theStudents[i].sapId + "\t\t" + theStudents[i].course + "\t\t" + theStudents[i].branch + "\t\t" + theStudents[i].rollNo);

}

}

**public** **int** nStudent(){

System.***out***.println("Enter Reg Number:");

String sapId = sc.next();

**for** (**int** i=0; i<count; i++){

**if** (theStudents[i].sapId.equalsIgnoreCase(sapId)){

**return** i;

}

}

System.***out***.println("Sorry,You are not Registered!!!");

System.***out***.println("Please Register yourself first!!!");

**return** 2;

}

**public** **void** issueBook(Books book){

**int** studentIndex =nStudent();

**if** (studentIndex!=2){

Book b = book.issueBook();

System.***out***.println("issuing book....");

**if** (b!= **null**){

System.***out***.println("adding book to your account");

}

}

}

}

Books Class:

**package** library;

**import** java.util.Scanner;

**public** **class** Books{

Book theBooks[] = **new** Book[15];

**public** **int** count;

Scanner sc = **new** Scanner(System.***in***);

**public** **int** compareBook(Book b1, Book b2)

{

**if** (b1.bookName.equalsIgnoreCase(b2.bookName)){

System.***out***.println("Book of this Name Already Exists.");

**return** 0;

}

**if** (b1.sNo==b2.sNo){

System.***out***.println("Book of this Serial No Already Exists.");

**return** 0;

}

**return** 1;

}

**public** **int** addBook(Book b){

**for** (**int** i=0; i<count; i++){

**if** (compareBook(b,theBooks[i]) == 0)

**return** 0;

}

**if** (count<15){

theBooks[count] = b;

count++;

}

**else**{

System.***out***.println("No Space to Add More Books.");

}

**return** 0;

}

**public** **void** displayAllBooks(){

System.***out***.println("\t\t\t\tSHOWING ALL BOOKS\n");

System.***out***.println("S.No\t\tName\t\tAuthor\t\tAvailable Qty\t\tTotal Qty");

**for** (**int** i=0; i<count; i++){

System.***out***.println(theBooks[i].sNo + "\t\t" + theBooks[i].bookName + "\t\t" + theBooks[i].authorName + "\t\t" +

theBooks[i].bookQtyTotal + "\t\t" + theBooks[i].bookQty);

}

}

**public** **void** increaseBookQty(){

System.***out***.println("\t\t\t\tINCREASE QUANTITY OF BOOK\n");

System.***out***.println("Enter Serial No of Book");

**int** sNo = sc.nextInt();

**for** (**int** i=0; i<count; i++){

**if** (sNo == theBooks[i].sNo){

System.***out***.println("Enter No of Books to be Added:");

**int** addingQty = sc.nextInt();

theBooks[i].bookQty += addingQty;

theBooks[i].bookQtyTotal += addingQty;

**return**;

}

}

}

**public** **void** displayMenu(){

System.***out***.println("----------------------------------------------------------------------------------------------------------");

System.***out***.println("1. Add new Book.");

System.***out***.println("2. Increase Quantity of a Book.");

System.***out***.println("3. Show All Books.");

System.***out***.println("4. Register Student.");

System.***out***.println("5. Show All Registered Students.");

System.***out***.println("6. Issue Book. ");

System.***out***.println("0. Exit"); }

**public** **int** availableBook(**int** sNo){

**for** (**int** i=0; i<count; i++){

**if** (sNo == theBooks[i].sNo){

**if**(theBooks[i].bookQtyTotal > 0){

System.***out***.println("Book is Available.");

**return** i;

}

System.***out***.println("Book is Unavailable");

**return** 2;

}

}

System.***out***.println("No Book of Serial Number " + " Available in Library.");

**return** 2;

}

**public** Book issueBook(){

System.***out***.println("Enter Serial No of Book to be Checked Out.");

**int** sNo = sc.nextInt();

**int** bookIndex =availableBook(sNo);

**if** (bookIndex!=2){

theBooks[bookIndex].bookQtyTotal--;

**return** theBooks[bookIndex];

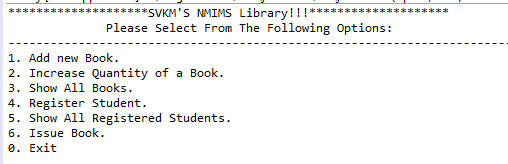
}

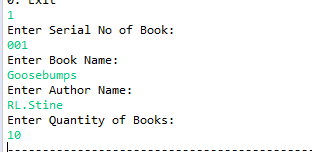
**return** **null**;

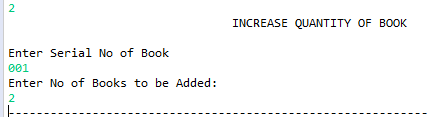
}

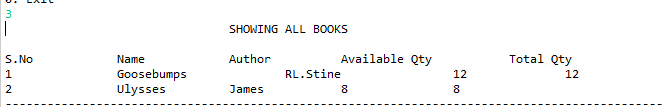
}

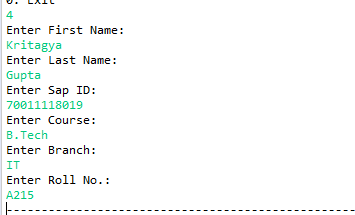
**OUTPUT SCREEN**



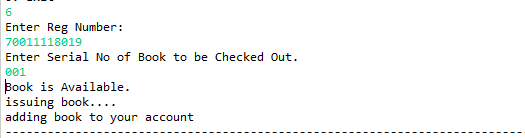














**USER MANUAL**

This is a JAVA based Application, requirements to run this application are:

1. Desktop/Laptop

2. OS (LINUX/Windows/MACOS)

3. JAVA Compiler

Steps to install Java compiler:

In Windows-

Step 0: Un-Install Older Version(s) of JDK/JRE

I recommend that you install only the latest JDK. Although you can install multiple versions of JDK/JRE concurrently, it is messy.

If you have previously installed older version(s) of JDK/JRE, un-install ALL of them. Go to "Control Panel" ⇒ (optional) "Programs" ⇒ "Programs and Features" ⇒ Un-install ALL programs begin with "Java", such as "Java SE Development Kit ...", "Java SE Runtime ...", "Java X Update ...", and etc.

Step 1: Download JDK

Goto http://www.oracle.com/technetwork/java/javase/downloads/index.html.

Under "Java Platform, Standard Edition" ⇒ "Java SE 13.0.{x}", where {x} denotes a fast running security-update number ⇒ Click the "Oracle JDK Download" button.

Under "Java SE Development Kit 13.0.{x}" ⇒ Check "Accept License Agreement".

Choose the JDK for your operating system, i.e., "Windows". Download the "exe" installer (e.g., "jdk-13.0.{x}\_windows-x64\_bin.exe" - about 159MB).

Step 2: Install JDK

Run the downloaded installer (e.g., "jdk-13.0.{x}\_windows-x64\_bin.exe"), which installs both the JDK and JRE.

By default, JDK is installed in directory "C:\Program Files\Java\jdk-13.0.{x}", where {x} denotes the update number. Accept the defaults and follow the screen instructions to install JDK.

Use your "File Explorer", navigate to "C:\Program Files\Java" to inspect the sub-directories. Take note of your JDK installed directory jdk-13.0.{x}, in particular, the update number {x}, which you will need in the next step.

I shall refer to the JDK installed directory as <JAVA\_HOME>, hereafter, in this article.

Step 3: Include JDK's "bin" Directory in the PATH

Windows' Command Prompt (CMD) searches the current directory and the directories listed in the PATH environment variable (or system variable) for executable programs. JDK's programs (such as Java compiler "javac.exe" and Java runtime "java.exe") reside in the sub-directory "bin" of the JDK installed directory. You need to include JDK's "bin" in the PATH to run the JDK programs.

To edit the PATH environment variable in Windows 10:

Launch "Control Panel" ⇒ (Optional) "System and Security" ⇒ "System" ⇒ Click "Advanced system settings" on the left pane.

Switch to "Advanced" tab ⇒ Click "Environment Variables" button.

Under "System Variables" (the bottom pane), scroll down to select variable "Path" ⇒ Click "Edit...".

For Newer Windows 10:

You shall see a TABLE listing all the existing PATH entries (if not, goto next step). Click "New" ⇒ Click "Browse" and navigate to your JDK's "bin" directory, i.e., "c:\Program Files\Java\jdk-13.0.{x}\bin", where {x} is your installation update number ⇒ Select "Move Up" to move this entry all the way to the TOP.

For Older Windows 10 (Time to change your computer!):

(CAUTION: Read this paragraph 3 times before doing this step! Don't push "Apply" or "OK" until you are 101% sure. There is no UNDO!!!)

(To be SAFE, copy the content of the "Variable value" to Notepad before changing it!!!)

In "Variable value" field, APPEND "c:\Program Files\Java\jdk-13.0.{x}\bin" (where {x} is your installation update number) IN FRONT of all the existing directories, followed by a semi-colon (;) to separate the JDK's bin directory from the rest of the existing

Directories. DO NOT DELETE any existing entries; otherwise, some existing applications may not run.

Variable name: PATH

Variable value: c:\Program Files\Java\jdk-13.0.{x}\bin;[do not delete exiting entries...]

Note: If you have started CMD, you need to re-start for the new environment settings to take effect.

Step 4: Verify the JDK Installation

Launch a CMD via one of the following means:

Click "Search" button ⇒ Type "cmd" ⇒ Choose "Command Prompt", or

Right-click "Start" button ⇒ run... ⇒ enter "cmd", or

Click "Start" button ⇒ Windows System ⇒ Command Prompt

Issue the following commands to verify your JDK installation:

Issue "path" command to list the contents of the PATH environment variable. Check to make sure that your JDK's "bin" is listed in the PATH.

PATH=c:\Program Files\Java\jdk-13.0.{x}\bin;other entries...

Issue the following commands to verify that JDK/JRE are properly installed and display their version:

// Display the JDK version

javac -version

javac 13.0.1

// Display the JRE version

java -version

java version "13.0.1" 2019-10-15

Java(TM) SE Runtime Environment (build 13.0.1+9)

Java HotSpot(TM) 64-Bit Server VM (build 13.0.1+9, mixed mode, sharing)

FOR MACOS-

Download the JDK .dmg file, jdk-10.interim.update.patch-macosx-x64.dmg.

Before the file can be downloaded, you must accept the license agreement.

From either the browser Downloads window or from the file browser, double-click the .dmg file to start it.

A Finder window appears that contains an icon of an open box and the name of the .pkg file.

Double-click the JDK 10.pkg icon to start the installation application.

The installation application displays the Introduction window.

Click Continue.

The Installation Type window appears.

Click Install.

A window appears that displays the message: Installer is trying to install new software. Enter your password to allow this.

Enter the Administrator user name and password and click Install Software.

The software is installed and a confirmation window is displayed.

FOR LINUX-

At the time of writing, the latest LTS version of Java is version 11.

Follow the steps below to install Java OpenJDK 11 on your Ubuntu system:

First, update the apt package index with:

sudo apt update

Once the package index is updated install the default Java OpenJDK package with:

sudo apt install default-jdk

Verify the installation, by running the following command which will print the Java version:

java -version

The output will look something like this:

openjdk version "11.0.2" 2019-01-15OpenJDK Runtime Environment (build 11.0.2+9-Ubuntu-3ubuntu118.04.3)OpenJDK 64-Bit Server VM (build 11.0.2+9-Ubuntu-3ubuntu118.04.3, mixed

Now Compile the test.java file and run it.

You have successfully entered the Application.

**CONCLUSION**

The **Library Management System** allows the user to store the book details and the person's details. This software allows storing the details of all the data related to **library**. The implementation of the **system** will reduce data entry time and provide readily calculated reports

**REFERENCES**

* <https://youtu.be/cv6GvRCluTs>
* <https://youtu.be/g0PrXoEKM2Y>
* <https://www.objectaid.com/home>
* <https://www.geeksforgeeks.org/equalsignorecase-in-java/>