
STUDENT MANAGEMENT SYSTEM

**Core Java and MySql
Project**

NAME: AADITYA KUMAR

GUIDED BY: INDRAKKA MALI

Batch

Code:2022-9100

Date:06-06-2023

CORE JAVA PROJECT

PROJECT TITLE: **STUDENT MANAGEMENT SYSTEM**

AIM:

- To achieve the Student Management Process without any interruptions.

LANGUAGES:

- JAVA and SQL

SOFTWARE REQUIREMENTS:

- Eclipse for java.
- Mysql for sql.

SYNOPSIS:

In this STUDENT MANAGEMENT SYSTEM there are some main operations .

MAIN OPERATIONS are:

- ✧ REGISTRATION → Student can perform this operation.
- ✧ LOGIN → Both Admin and Student can perform this operation .
- ✧ ADD COURSE → Admin can perform this operation.
- ✧ UPDATE AND DELETE COURSE → Admin can perform that operation
- ✧ DISPLAY STUDENT RECORDS → Both Admin and Student can perform this operation .
- ✧ VIEW COURSE → Both Admin and Student can perform
- ✧ UPDATE PROFILE → Student can perform this operation.

The control is given for both the admin and the Student. before perform any operation first admin must be login the system. Admin can perform six operations one is the fetching information about the registered student. Admin Can delete the registered student. Admin can add the new course for students. Admin can view course and delete course. admin can update the course. In that student side there is four main operations are given. The first one is registration operation it will make permanent account creation. Second one is login when ever you need to visit your page for that security purpose this module will help. Third one is student details this will take personal information's about that student. Student can Enroll for the new course which is added by the Admin. Student can view the enroll course and student can update the his/her profile.

INITIAL SETUP:

(IN JAVA):

Maven Project Name: StudentManagementSystemProject.

Package Names : com.student.view, com.student.db, com.student.services

Class Names :

1. MainJdbcApp(com.student.view)
2. DatabaseConnection(com.student.db)
2. AdminOperation(com.student.services)
3. StudentOperations(com.student.services)

(IN DATABASE):

Creating database in sql (query)

```
Mysql > create database Student;
```

```
Query OK, 1 row affected (0.31 sec)
```

Use Database in

sql(query)

```
mysql>use student;
```

```
Database changed
```

Tables Creation:

```
mysql> use student
Database changed
mysql> show tables;
+-----+
| Tables_in_student |
+-----+
| adminlogin         |
| course             |
| enrollcourse       |
| student_registration |
+-----+
4 rows in set (0.07 sec)
```

Database table creation:

Admin Login table structure:

```
mysql> desc adminlogin;
+-----+-----+-----+-----+-----+-----+
| Field      | Type          | Null | Key | Default | Extra          |
+-----+-----+-----+-----+-----+-----+
| Aid        | int           | NO   | PRI | NULL    | auto_increment |
| Email_id   | varchar(50)   | NO   | UNI | NULL    |                |
| Password   | varchar(50)   | YES  |     | NULL    |                |
+-----+-----+-----+-----+-----+-----+
3 rows in set (0.02 sec)
```

Student Registration table structure:

```
mysql> desc student_registration;
```

Field	Type	Null	Key	Default	Extra
Stud_id	int	NO	PRI	NULL	auto_increment
Email_id	varchar(50)	NO	UNI	NULL	
Password	varchar(50)	NO		NULL	
Full_Name	varchar(100)	NO		NULL	
Mobile_No	varchar(50)	NO	UNI	NULL	
age	int	YES		NULL	
Address	varchar(100)	YES		NULL	
Date_Of_Birth	date	YES		NULL	
Gender	varchar(50)	YES		NULL	

9 rows in set (0.00 sec)

Course table structure:

```
mysql> desc course;
```

Field	Type	Null	Key	Default	Extra
Cid	int	NO	PRI	NULL	
course_name	varchar(100)	NO		NULL	
Duration	time	YES		NULL	
course_fee	float	YES		NULL	

4 rows in set (0.00 sec)

Enroll Course table structure:

```
mysql> desc enrollcourse;
+-----+-----+-----+-----+-----+-----+
| Field | Type | Null | Key | Default | Extra |
+-----+-----+-----+-----+-----+-----+
| ceID  | int  | NO   | PRI | NULL    | auto_increment |
| Stud_id | int  | YES  | MUL | NULL    |                |
| Cid   | int  | YES  | MUL | NULL    |                |
| fees  | float | YES  |     | NULL    |                |
+-----+-----+-----+-----+-----+-----+
4 rows in set (0.00 sec)

mysql> _
```

Database Connection Code:

```
package com.student.db;

import java.sql.Connection;
import java.sql.DriverManager;

public class DatabaseConnection {
    //load the driver
        //make the connection
        //create a statement object
        //exeute sql commands

    public static String driver="com.mysql.cj.jdbc.Driver";
    public static String url="jdbc:mysql://localhost:3306/student";
    public static String un="root";
    public static String up="root";
    public static Connection conn;

    public static Connection getConnection() throws Exception {

        Class.forName(driver);
        conn=DriverManager.getConnection(url, un, up);
        if(conn==null) {
            System.out.println("Error in getting connection");
        }
        return conn;

    }
}
```

MainApp.Java Code:

Here I have created Switch case To Perform operation.

There is three option one is Admin Login ,Student Registration ,student Login

By choosing this option Admin And student Can perform the operation.

Here is code Of MainApp.Java.

```
package com.student.view;
```

```
import java.util.Scanner;
```

```
import com.student.services.AdminOperation;
```

```
import com.student.services.StudentOperations;
```

```
public class MainApp
```

```
{
```

```
public static void main(String[] args) throws Exception
```

```
{
```

```
    int ch,ch1;
```

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

```
    System.out.println();
```

```
    System.out.println("<<<<<<<<<<WELCOME TO STUDENT  
MANAGEMENT SYSTEM>>>>>>>>>");
```

```
    System.out.println();
```

```
    while(true) {
```

```
        System.out.println("Admin Login Or Student Login");
```

```
        System.out.println();
```

```
        System.out.println("1.ADMIN LOGIN \n2.STUDENT  
REGISTRATION \n3.STUDENT LOGIN");
```

```
System.out.println("=====");
```

```
    ch=sc.nextInt();
```

```
    switch(ch){
```

```
    case 1:if(AdminOperation.AdminLogin())
```

```
    {
```

```
        System.out.println("Login Successfully..!!");
```

```
        System.out.println();
```

```
        System.out.println("*****Welcome To Admin  
Dashboard*****");
```

```
        System.out.println();
```

```
        while(true)
```

```
        {
```



```

        System.out.println("1.VIEW STUDENT \n2.DELETE
STUDENT \n3.VIEW COURSE \n4.ADD COURSE \n5.DELETE COURSE
\n6.UPDATE COURSE");

```

```

System.out.println("=====");

```

```

        ch1=sc.nextInt();
        switch(ch1)
        {
            case 1:AdminOperation.viewStudents();
                break;
            case 2:AdminOperation.deleteStudent();
                break;
            case 3:AdminOperation.viewCourse();
                break;
            case 4:AdminOperation.addCourse();
                break;
            case 5:AdminOperation.deleteCourse();
                break;
            case 6:AdminOperation.updateCourse();
                break;
        }

```

```

        System.out.println();

```

```

        System.out.println("'Admin' Do you want to Perform
Other Operation Press 'y' for Yess AND Press 'n' for No");

```

```

        char choice=sc.next().charAt(0);
        if(choice=='n')
        {
            break;
        }

```

```

    }//while

```

```

    }
    else
    {

```

```

        System.out.println("Incorrect Login ");
    }

```

```

    break;

```

```

    case 2:if(StudentOperations.StudRegistration())
    {

```

```

        System.out.println("Successfully Registered..Please login");
    }

```

```

    else
    {

```

```

        System.out.println("Student Already Registered...Please
Login");
    }

    break;

    case 3:if(StudentOperations.Studlogin())
    {
        System.out.println("Login Successfully..!!");
        System.out.println();
        System.out.println("*****Welcome To Student
Dashboard*****");
        System.out.println();
        while(true)
        {
            System.out.println("1.VIEW PROFILE \n2.UPDATE
PROFILE \n3.ENROLL COURSE \n4.VIEW ENROLL COURSE");

System.out.println("=====");
            ch1=sc.nextInt();
            switch(ch1)
            {
                case 1:StudentOperations.ViewProfile();
                    break;
                case 2:StudentOperations.UpdateProfile();
                    break;
                case 3:StudentOperations.CourseEnroll();
                    break;
                case 4:StudentOperations.ViewEnrolledCourse();
                    break;
            }
            System.out.println();
            System.out.println("'Student' Do you want to Perform
Other Operation Press 'y' for continue AND 'n' for stop");
            char choice=sc.next().charAt(0);
            if(choice=='n')
            {
                break;
            }
        }
    }
    else
    {
        System.out.println("Incorrect Credentials..!! Or first Registered

```

```
your self");
        }
        break;

    }//switch
//      System.out.println("Do you Want To Login...Press 'y' for Yess AND
Press 'N' for No");
//      char choice=sc.next().charAt(0);
//      if(choice=='n') {
//          break;
//      }
    }//while

}

}

}
```

Output :-

MainJDBCApp (4) [Java Application] C:\Users\Aaditya\Downloads\Eclipse Batch 20

```
<<<<<<<<WELCOME TO STUDENT MANAGEMENT SYSTEM>>>>>>>>
```

Admin Login Or Student Login

- 1.ADMIN LOGIN
- 2.STUDENT REGISTRATION
- 3.STUDENT LOGIN

1

Enter Email Id:

Admin@gmail.com

Enter the Password:

Admin@123

Login Successfully..!!

```
*****Welcome To Admin Dashboard*****
```

- 1.VIEW STUDENT
- 2.DELETE STUDENT
- 3.VIEW COURSE
- 4.ADD COURSE
- 5.DELETE COURSE
- 6.UPDATE COURSE

=====

MainJDBCApp (4) [Java Application] C:\Users\Aaditya\Downloads\Eclipse Batch 2022-9100

```
<<<<<<<<WELCOME TO STUDENT MANAGEMENT SYSTEM>>>>>>>>
```

Admin Login Or Student Login

- 1.ADMIN LOGIN
- 2.STUDENT REGISTRATION
- 3.STUDENT LOGIN

2

Enter Email Id:

Nikhil@gmail.com

Enter the Password:

Nikhil@123

Enter the Name:

Nikhil

Enter the Mobile Number:

9238804565

Enter the Age:

22

Enter the Address:

Haryana

Enter the Date Of Birth:

2000-06-08

Enter the Gender:

Male

Successfully Registered..Please login

Admin Login Or Student Login

- 1.ADMIN LOGIN
- 2.STUDENT REGISTRATION
- 3.STUDENT LOGIN

=====

MainJDBCApp (4) [Java Application] C:\Users\Aaditya\Downloads\Eclipse Batch

<<<<<<<<WELCOME TO STUDENT MANAGEMENT SYSTEM>>>>>>>>>

Admin Login Or Student Login

- 1.ADMIN LOGIN
- 2.STUDENT REGISTRATION
- 3.STUDENT LOGIN

=====

3

Enter Email Id:

Nikhil@gmail.com

Enter the Password:

Nikhil@123

Login Successfully..!!

*****Welcome To Student Dashboard*****

- 1.VIEW PROFILE
- 2.UPDATE PROFILE
- 3.ENROLL COURSE
- 4.VIEW ENROLL COURSE

=====

Admin Operation :-

In this File I have added All the Methods Of Admin.first admin Login the system. Admin can view the Profile of student, admin can delete this student . admin can add some courses ,admin can delete course etc.

```
package com.student.services;

import java.sql.Connection;
import java.sql.PreparedStatement;
import java.sql.ResultSet;
import java.util.Scanner;

import com.student.db.DatabaseConnection;

public class AdminOperation
{
    static Connection con;
    static String course_name;
    static String Duration;
    static float course_fee;
    static String Email_id,Password;
    public static boolean AdminLogin() throws Exception {

        con=DatabaseConnection.getConnection();
        PreparedStatement pst;
        Scanner sc=new Scanner(System.in);
        System.out.println("Enter Email Id:");
        Email_id=sc.next();
        pst=con.prepareStatement("select * from AdminLogin where
Email_id=?");
        pst.setString(1,Email_id );
        ResultSet rs=pst.executeQuery();

        if(rs.next())
        {
            System.out.println("Enter the Password:");
            Password=sc.next();

            if(rs.getString(2).contentEquals(Email_id) &&
rs.getString(3).contentEquals(Password))
            {
                return true;
            }
        }
    }
}
```

```

//                System.out.println("Login succefully");
                }
                else
                {
                    return false;
                    //System.out.println("Password Wrong...!!");
                }
            }
            else

                //System.out.println("Email Id not present Wrong...!!");
                return false;
        }
        public static void viewStudents() throws Exception
        {
            con=DatabaseConnection.getConnection();
            PreparedStatement pst=con.prepareStatement("select * from
Student_Registration");
            ResultSet rs=pst.executeQuery();
            System.out.println("STUD_ID\tEMAIL
ID\tPASSWORD\tNAME\tMOBILE\tAGE\tADDRESS\tDOB\t GENDER");
            System.out.println(".....");
            .....");
            while(rs.next())
            {

                System.out.println(rs.getInt(1)+"\t"+rs.getString(2)+"\t"+rs.getString(3)+"\t"+rs.getString(4)+"\t"+rs.getString(5)+"\t"+rs.getInt(6)
                +"\t"+rs.getString(7)+"\t"+rs.getString(8)+"\t"+rs.getString(9));
            }

        }

        public static void viewCourse() throws Exception
        {
            con=DatabaseConnection.getConnection();
            PreparedStatement pst=con.prepareStatement("select * from course");
            ResultSet rs=pst.executeQuery();
            System.out.println("CID\tCOURSE NAME\tDuration\tFee");
            System.out.println(".....");

```

```

while(rs.next())
{

    System.out.println(rs.getInt(1)+"\t"+rs.getString(2)+"\t\t"+rs.getString(3)+"\t"+rs.getFloat(4));
}
}

public static void addCourse() throws Exception
{
    con=DatabaseConnection.getConnection();
    PreparedStatement pst;
    Scanner sc=new Scanner(System.in);
    System.out.print("Enter the course ID:");
    int cid=sc.nextInt();
    pst=con.prepareStatement("select * from course where cid=?");
    pst.setInt(1, cid);
    ResultSet rs=pst.executeQuery();
    if(!rs.next())
    {

        System.out.println("Enter the Course Name:");
        course_name=sc.next();
        System.out.println("Enter the Course Duration:");
        Duration=sc.next();
        System.out.println("Enter the Course fee:");
        course_fee=sc.nextFloat();

        pst=con.prepareStatement("insert into course values(?,?,?,?)");
        pst.setInt(1, cid);
        pst.setString(2, course_name);
        pst.setString(3, Duration);
        pst.setFloat(4, course_fee);
        int k=pst.executeUpdate();
        if(k>0)
        {
            System.out.println("Course Inserted Successfully!!");
        }

    }
    else
    {
        System.out.println("Course ID already Present");
    }
}

```



```

    }
}
public static void deleteCourse() throws Exception
{
    con=DatabaseConnection.getConnection();
    PreparedStatement pst;
    Scanner sc=new Scanner(System.in);
    System.out.println("Enter Course Id To delete the Course:");
    int cid=sc.nextInt();
    pst=con.prepareStatement("select * from course where cid=?");
    pst.setInt(1, cid);
    ResultSet rs=pst.executeQuery();
    if(rs.next())
    {
        pst=con.prepareStatement("delete from course where cid=?");
        pst.setInt(1, cid);
        int k=pst.executeUpdate();
        if(k>0)
        {
            System.out.println("Course Deleted Successfully...!!");
        }
    }
    else
    {
        System.out.println("Course Not Present");
    }
}

public static void deleteStudent() throws Exception
{
    con=DatabaseConnection.getConnection();
    PreparedStatement pst;
    Scanner sc=new Scanner(System.in);
    System.out.println("Enter Student Id To delete the Student:");
    int Stud_id=sc.nextInt();
    pst=con.prepareStatement("select * from Student_Registration where
Stud_id=?");
    pst.setInt(1, Stud_id);
    ResultSet rs=pst.executeQuery();
    if(rs.next())
    {
        pst=con.prepareStatement("delete from Student_Registration
where Stud_id=?");
        pst.setInt(1, Stud_id);
    }
}

```

```

        int k=pst.executeUpdate();
        if(k>0)
        {
            System.out.println("Student Record Deleted
Successfully...!!");
        }

    }
    else
    {
        System.out.println("Student Record Not Present");
    }
}
public static void updateCourse() throws Exception
{
    con=DatabaseConnection.getConnection();
    PreparedStatement pst;
    Scanner sc=new Scanner(System.in);
    System.out.println("Enter Course Id To Update the Course:");
    int cid=sc.nextInt();
    pst=con.prepareStatement("select * from course where cid=?");
    pst.setInt(1, cid);
    ResultSet rs=pst.executeQuery();
    if(rs.next())
    {
        while(true)
        {
            System.out.println("Which Feild do you want to
update");

            System.out.println("=====");
            System.out.println("1 UPDATE COURSE NAME
\n2 UPDATE COURSE DURATION \n3 UPDATE COURSE FEE");
            int ch=sc.nextInt();
            switch(ch)
            {
                case 1: System.out.println("enter the Course Name To
update");

                course_name=sc.next();
                String str="update course set course_name=? where
cid=?";

                pst=con.prepareStatement(str);
                pst.setString(1, course_name);

```

```

        pst.setInt(2, cid);
        int k=pst.executeUpdate();
        if(k>0)
        {
            System.out.println("Record Updated
Succesfully");
        }
        else
        {
            System.out.println("Not Updated...");
        }

        break;

        case 2: System.out.println("enter the Course Duration");
        Duration=sc.next();
        str="update course set Duration=? where cid=?";

        pst=con.prepareStatement(str);
        pst.setString(1, Duration);
        pst.setInt(2, cid);
        k=pst.executeUpdate();
        if(k>0)
        {
            System.out.println("Record Updated Succesfully");
        }
        else
        {
            System.out.println("Not Updated...");
        }

        break;

        case 3: System.out.println("enter the Course Fees");
        course_fee=sc.nextFloat();
        str="update course set course_fee=? where cid=?";

        pst=con.prepareStatement(str);
        pst.setFloat(1, course_fee);
        pst.setInt(2, cid);
        k=pst.executeUpdate();
        if(k>0)
        {

```

```

        System.out.println("Record Updated Succesfully");
    }
    else
    {
        System.out.println("Not Updated...");
    }

    break;
}
System.out.println();
System.out.println("Want to Update Other Records...Press 'y'
for Yess AND Press 'n' for No");
    char choice=sc.next().charAt(0);
    if(choice=='n')
    {
        break;
    }
}
else
{
    System.out.println("Course ID Not Present");
}
}

}

```

OUTPUT:

Admin Login:

Admin first login the system by entering email Id and Password.

```
MainJDBCApp (4) [Java Application] C:\Users\Aaditya\Downloads\Eclipse Batch 20
<<<<<<<<<WELCOME TO STUDENT MANAGEMENT SYSTEM>>>>>>>>>

Admin Login Or Student Login

1.ADMIN LOGIN
2.STUDENT REGISTRATION
3.STUDENT LOGIN
=====
1
Enter Email Id:
Admin@gmail.com
Enter the Password:
Admin@123
Login Successfully..!!

*****Welcome To Admin Dashboard*****

1.VIEW STUDENT
2.DELETE STUDENT
3.VIEW COURSE
4.ADD COURSE
5.DELETE COURSE
6.UPDATE COURSE
=====
```

View Registered Students:

After Login Admin can view the Registered students List:

```
MainJDBCApp (4) [Java Application] C:\Users\Aaditya\Downloads\Eclipse Batch 2022-9100\eclipse\plugins\org.eclipse.justj.openjdk.hotspot.jre.full.win32.x86_64_17.0.1.v21
```

```
<<<<<<<<<WELCOME TO STUDENT MANAGEMENT SYSTEM>>>>>>>>>>>>
```

```
Admin Login Or Student Login
```

- 1.ADMIN LOGIN
- 2.STUDENT REGISTRATION
- 3.STUDENT LOGIN

```
=====
1
Enter Email Id:
Admin@gmail.com
Enter the Password:
Admin@123
Login Successfully...!!
```

```
*****Welcome To Admin Dashboard*****
```

- 1.VIEW STUDENT
- 2.DELETE STUDENT
- 3.VIEW COURSE
- 4.ADD COURSE
- 5.DELETE COURSE
- 6.UPDATE COURSE

```
=====
1
STUD_ID      EMAIL_ID      PASSWORD      NAME      MOBILE      AGE      ADDRESS DOB      GENDER
-----
1            Akshay@gmail.com  Akshay@123    Akshay    9874566214   26      Delhi 1997-10-26    Male
2            Abhishek@gmail.com Abhishek@123  Abhishek  9522207845   24      Mumbai 1998-09-12      Male
3            Akhil@gmail.com  Akhil@123     Akhil     9815445278   24      Bihar 1999-04-06      Male
4            Nikhil@gmail.com  Nikhil@123    Nikhil    9238804565   22      Haryana 2000-06-08      Male
```

```
'Admin' Do you want to Perform Other Operation Press 'y' for Yess AND Press 'n' for No
```

Mysql Records:-

```
mysql> select * from student_registration;
+-----+-----+-----+-----+-----+-----+-----+-----+-----+
| Stud_id | Email_id | Password | Full_Name | Mobile_No | age | Address | Date_Of_Birth | Gender |
+-----+-----+-----+-----+-----+-----+-----+-----+-----+
| 1 | Akshay@gmail.com | Akshay@123 | Akshay | 9874566214 | 26 | Delhi | 1997-10-26 | Male |
| 2 | Abhishek@gmail.com | Abhishek@123 | Abhishek | 9522207845 | 24 | Mumbai | 1998-09-12 | Male |
| 3 | Akhil@gmail.com | Akhil@123 | Akhil | 9815445278 | 24 | Bihar | 1999-04-06 | Male |
| 4 | Nikhil@gmail.com | Nikhil@123 | Nikhil | 9238804565 | 22 | Haryana | 2000-06-08 | Male |
+-----+-----+-----+-----+-----+-----+-----+-----+-----+
4 rows in set (0.01 sec)
```

View Course :

After Login Admin can view the course List:

```
<<<<<<<<WELCOME TO STUDENT MANAGEMENT SYSTEM>>>>>>>>
```

Admin Login Or Student Login

- 1.ADMIN LOGIN
- 2.STUDENT REGISTRATION
- 3.STUDENT LOGIN

=====

1

Enter Email Id:

Admin@gmail.com

Enter the Password:

Admin@123

Login Successfully..!!

```
*****Welcome To Admin Dashboard*****
```

- 1.VIEW STUDENT
- 2.DELETE STUDENT
- 3.VIEW COURSE
- 4.ADD COURSE
- 5.DELETE COURSE
- 6.UPDATE COURSE

=====

3

CID	COURSE NAME	Duration	Fee
111	Java	48:00:00	1000.0
222	Python	48:00:00	1000.0
333	MySQL	48:00:00	1000.0
444	AWS	56:00:00	2000.0

```
'Admin' Do you want to Perform Other Operation Press 'y' for Yess AND Press 'n' for No
```

Admin Add new course:

After Login Admin can Add the new course in the database:

Admin add course name, course duration And course Fee. So that Student can Enroll For that course.

```
<<<<<<<<<WELCOME TO STUDENT MANAGEMENT SYSTEM>>>>>>>>>>>>
```

```
Admin Login Or Student Login
```

- 1.ADMIN LOGIN
- 2.STUDENT REGISTRATION
- 3.STUDENT LOGIN

```
=====
```

```
1
```

```
Enter Email Id:
```

```
Admin@gmail.com
```

```
Enter the Password:
```

```
Admin@123
```

```
Login Successfully...!!
```

```
*****Welcome To Admin Dashboard*****
```

- 1.VIEW STUDENT
- 2.DELETE STUDENT
- 3.VIEW COURSE
- 4.ADD COURSE
- 5.DELETE COURSE
- 6.UPDATE COURSE

```
=====
```

```
4
```

```
Enter the course ID:555
```

```
Enter the Course Name:
```

```
Testing
```

```
Enter the Course Duration:
```

```
48:00:00
```

```
Enter the Course fee:
```

```
1000
```

```
Course Inserted Successfully!!
```

```
'Admin' Do you want to Perform Other Operation Press 'y' for Yess AND Press 'n' for No
```


Mysql Course Table Records:-

```
mysql> desc course;
```

Field	Type	Null	Key	Default	Extra
Cid	int	NO	PRI	NULL	
course_name	varchar(100)	NO		NULL	
Duration	time	YES		NULL	
course_fee	float	YES		NULL	

```
4 rows in set (0.00 sec)
```

```
mysql> select * from course;
```

Cid	course_name	Duration	course_fee
111	Java	48:00:00	1000
222	Python	48:00:00	1000
333	MySQL	48:00:00	1000
444	AWS	56:00:00	2000
555	Testing	48:00:00	1000

```
5 rows in set (0.01 sec)
```

Admin Delete course:

Admin can delete the course which is added by admin itself.

```
<<<<<<<<<WELCOME TO STUDENT MANAGEMENT SYSTEM>>>>>>>>>>>>>
```

```
Admin Login Or Student Login
```

```
1.ADMIN LOGIN
2.STUDENT REGISTRATION
3.STUDENT LOGIN
```

```
=====
```

```
1
```

```
Enter Email Id:
```

```
Admin@gmail.com
```

```
Enter the Password:
```

```
Admin@123
```

```
Login Successfully..!!
```

```
*****Welcome To Admin Dashboard*****
```

```
1.VIEW STUDENT
2.DELETE STUDENT
3.VIEW COURSE
4.ADD COURSE
5.DELETE COURSE
6.UPDATE COURSE
```

```
=====
```

```
5
```

```
Enter Course Id To delete the Course:
```

```
555
```

```
Course Deleted Successfully...!!
```

```
'Admin' Do you want to Perform Other Operation Press 'y' for Yess AND Press 'n' for No
```

```
mysql> select * from course;
+-----+-----+-----+-----+
| Cid | course_name | Duration | course_fee |
+-----+-----+-----+-----+
| 111 | Java        | 48:00:00 | 1000        |
| 222 | Python      | 48:00:00 | 1000        |
| 333 | MySQL       | 48:00:00 | 1000        |
| 444 | AWS         | 56:00:00 | 2000        |
+-----+-----+-----+-----+
4 rows in set (0.00 sec)
```

Admin Delete Registered Students:

Admin can delete the records of registered students:

```
<<<<<<<<<WELCOME TO STUDENT MANAGEMENT SYSTEM>>>>>>>>>>>>>
```

```
Admin Login Or Student Login
```

- 1.ADMIN LOGIN
- 2.STUDENT REGISTRATION
- 3.STUDENT LOGIN

```
=====
```

```
1
```

```
Enter Email Id:
```

```
Admin@gmail.com
```

```
Enter the Password:
```

```
Admin@123
```

```
Login Successfully..!!
```

```
*****Welcome To Admin Dashboard*****
```

- 1.VIEW STUDENT
- 2.DELETE STUDENT
- 3.VIEW COURSE
- 4.ADD COURSE
- 5.DELETE COURSE
- 6.UPDATE COURSE

```
=====
```

```
2
```

```
Enter Student Id To delete the Student:
```

```
3
```

```
Student Record Deleted Successfully...!!
```

```
'Admin' Do you want to Perform Other Operation Press 'y' for Yess AND Press 'n' for No
```

Admin Update the Course:

Admin Can Update the course. Admin update the course name, course duration , course fee.

```
=====
6
Enter Course Id To Update the Course:
444
Which Feild do you want to update
=====
1 UPDATE COURSE NAME
2 UPDATE COURSE DURATION
3 UPDATE COURSE FEE
1
enter the Course Name To update
AWSCertification
Record Updated Succesfully

Want to Update Other Records...Press 'y' for Yess AND Press 'n' for No
y
Which Feild do you want to update
=====
1 UPDATE COURSE NAME
2 UPDATE COURSE DURATION
3 UPDATE COURSE FEE
2
enter the Course Duration
36:00:00
Record Updated Succesfully

Want to Update Other Records...Press 'y' for Yess AND Press 'n' for No
y
Which Feild do you want to update
=====
1 UPDATE COURSE NAME
2 UPDATE COURSE DURATION
3 UPDATE COURSE FEE
3
enter the Course Fees
2000
Record Updated Succesfully
```

Student Operation :-

In this File I have added All the Methods Of Student.first admin Login the system. Student can view the Profile of student, Student can Update the profile. Student can view the Course Can Enroll the course etc.

```
package com.student.services;

import java.sql.*;
import java.util.Scanner;
import com.student.db.DatabaseConnection;

public class StudentOperations {

    static Connection con;
    static String
    Email_id,Password,Full_Name,Mobile_No,Address,Gender,Date_Of_Birth;
    static int age,Stud_id;

    public static boolean StudRegistration() throws Exception
    {
        con=DatabaseConnection.getConnection();
        PreparedStatement pst;
        Scanner sc=new Scanner(System.in);
        System.out.println("Enter Email Id:");
        Email_id=sc.next();
        pst=con.prepareStatement("select * from Student_Registration where
Email_id=?");
        pst.setString(1,Email_id );

        ResultSet rs=pst.executeQuery();
        if(!rs.next())
        {
            System.out.println("Enter the Password:");
            Password=sc.next();
            System.out.println("Enter the Name:");
            Full_Name=sc.next();
            System.out.println("Enter the Mobile Number:");
            Mobile_No=sc.next();
            System.out.println("Enter the Age:");
            age=sc.nextInt();
            System.out.println("Enter the Address:");
            Address=sc.next();
            System.out.println("Enter the Date Of Birth:");
            Date_Of_Birth=sc.next();
        }
    }
}
```

```

        System.out.println("Enter the Gender:");
        Gender=sc.next();
        pst=con.prepareStatement("insert into
Student_Registration(Email_id>Password,Full_Name,Mobile_No,age,Address,Date_Of_Birth, Gender) values(?,?,?,?,?,?,?) ");
        pst.setString(1, Email_id);
        pst.setString(2, Password);
        pst.setString(3, Full_Name);
        pst.setString(4, Mobile_No);
        pst.setInt(5, age);
        pst.setString(6, Address);
        pst.setString(7, Date_Of_Birth);
        pst.setString(8, Gender);

        int k=pst.executeUpdate();
        if(k>0)
        {
//            System.out.println("Record Inserted successfully");
            return true;
        }
        else
        {
            return false;
        }
    }
    else
        // System.out.println("Student Email Already Presents");

        return false;

}

public static boolean Studlogin() throws Exception
{
    con=DatabaseConnection.getConnection();
    PreparedStatement pst;
    Scanner sc=new Scanner(System.in);
    System.out.println("Enter Email Id:");
    Email_id=sc.next();
    pst=con.prepareStatement("select * from Student_Registration where
Email_id=?");
    pst.setString(1,Email_id );

```

```

ResultSet rs=pst.executeQuery();

    if(rs.next())
    {
        System.out.println("Enter the Password:");
        Password=sc.next();

        if(rs.getString(2).contentEquals(Email_id) &&
rs.getString(3).contentEquals(Password))
        {
            return true;
            //System.out.println("Login succefully");
        }
        else
        {
            return false;
            //System.out.println("Password Wrong...First Register Your
self");
        }
    }
    else

        //System.out.println("Email sId not present Wrong...First Register
Your self");

        return false;
}

public static void CourseEnroll() throws SQLException
{

    Scanner sc=new Scanner(System.in);
    PreparedStatement pst;
    System.out.println("Which Course Do You Want To Do");
    pst=con.prepareStatement("select * from course ");
    ResultSet rs=pst.executeQuery();
    System.out.println("ID\tCName\tDuration\tFee");
    System.out.println("-----");
    String Email_id=StudentOperations.Email_id;
    while(rs.next())
    {

        System.out.println(rs.getInt(1)+"\t"+rs.getString(2)+"\t"+rs.getString(3)+"\t"

```

```

+rs.getFloat(4));

    }
    while(true)
    {
        System.out.println("Enter Your choice:");
        int ch1=sc.nextInt();
        switch(ch1)
        {
            case 111:
                pst=con.prepareStatement("select Stud_id from
Student_Registration where Email_id=?");

                pst.setString(1, Email_id);
                ResultSet rs1=pst.executeQuery();

                while(rs1.next())
                {
                    Stud_id=rs1.getInt(1);
                }
                int Cid=111;
                System.out.println("Pay course fee");
                float fees=sc.nextFloat();
                pst=con.prepareStatement("insert into
EnrollCourse(Stud_id,Cid,fees) values(?,?,?)");
                pst.setInt(1, Stud_id);
                pst.setInt(2, Cid);
                pst.setFloat(3, fees);
                int k=pst.executeUpdate();
                if(k>0)
                {
                    System.out.println("Course Enroll Successfully.....!!");
                }
                else
                {
                    System.out.println("Not Enroll");
                }
                break;

            case 222:
                pst=con.prepareStatement("select Stud_id from
Student_Registration where Email_id=?");

                pst.setString(1, Email_id);

```



```

        rs1=pst.executeQuery();

        while(rs1.next())
        {
            Stud_id=rs1.getInt(1);
        }
        Cid=222;
        System.out.println("Pay course fee");
        fees=sc.nextFloat();
        pst=con.prepareStatement("insert into
EnrollCourse(Stud_id,Cid,fees) values(?,?,?)");
        pst.setInt(1, Stud_id);
        pst.setInt(2, Cid);
        pst.setFloat(3, fees);
        k=pst.executeUpdate();
        if(k>0)
        {
            System.out.println("Course Enroll Successfully.....!!");
        }
        else
        {
            System.out.println("Not Enroll");
        }
        break;

        case 333:
            pst=con.prepareStatement("select Stud_id from Student_Registration
where Email_id=?");

            pst.setString(1, Email_id);
            rs1=pst.executeQuery();

            while(rs1.next())
            {
                Stud_id=rs1.getInt(1);
            }
            Cid=333;
            System.out.println("Pay course fee");
            fees=sc.nextFloat();
            pst=con.prepareStatement("insert into
EnrollCourse(Stud_id,Cid,fees) values(?,?,?)");
            pst.setInt(1, Stud_id);
            pst.setInt(2, Cid);
            pst.setFloat(3, fees);

```

```

        k=pst.executeUpdate();
        if(k>0)
        {
            System.out.println("Course Enroll Successfully.....!!");
        }
        else
        {
            System.out.println("Not Enroll");
        }

        break;

        case 444:
            pst=con.prepareStatement("select Stud_id from Student_Registration
            where Email_id=?");

            pst.setString(1, Email_id);
            rs1=pst.executeQuery();

            while(rs1.next())
            {
                Stud_id=rs1.getInt(1);
            }
            Cid=444;
            System.out.println("Pay course fee");
            fees=sc.nextFloat();
            pst=con.prepareStatement("insert into
            EnrollCourse(Stud_id,Cid,fees) values(?,?,?)");
            pst.setInt(1, Stud_id);
            pst.setInt(2, Cid);
            pst.setFloat(3, fees);
            k=pst.executeUpdate();
            if(k>0)
            {
                System.out.println("Course Enroll Successfully.....!!");
            }
            else
            {
                System.out.println("Not Enroll");
            }

            break;

        case 555:

```

```
        pst=con.prepareStatement("select Stud_id from Student_Registration  
where Email_id=?");
```

```
        pst.setString(1, Email_id);  
        rs1=pst.executeQuery();
```

```
        while(rs1.next())  
        {  
            Stud_id=rs1.getInt(1);  
        }
```

```
        Cid=555;
```

```
        System.out.println("Pay course fee");  
        fees=sc.nextFloat();  
        pst=con.prepareStatement("insert into  
EnrollCourse(Stud_id,Cid,fees) values(?,?,?)");
```

```
        pst.setInt(1, Stud_id);  
        pst.setInt(2, Cid);  
        pst.setFloat(3, fees);
```

```
        k=pst.executeUpdate();  
        if(k>0)  
        {  
            System.out.println("Course Enroll Successfully.....!!");  
        }  
        else  
        {  
            System.out.println("Not Enroll");  
        }
```

```
        break;
```

```
        case 666:
```

```
        pst=con.prepareStatement("select Stud_id from Student_Registration  
where Email_id=?");
```

```
        pst.setString(1, Email_id);  
        rs1=pst.executeQuery();
```

```
        while(rs1.next())  
        {  
            Stud_id=rs1.getInt(1);  
        }
```

```
        Cid=666;
```

```
        System.out.println("Pay course fee");  
        fees=sc.nextFloat();
```

```

                                pst=con.prepareStatement("insert into
EnrollCourse(Stud_id,Cid,fees) values(?,?,?)");
                                pst.setInt(1, Stud_id);
                                pst.setInt(2, Cid);
                                pst.setFloat(3, fees);
                                k=pst.executeUpdate();
                                if(k>0)
                                {
                                    System.out.println("Course Enroll Successfully.....!!");
                                }
                                else
                                {
                                    System.out.println("Not Enroll");
                                }
                                break;

                                }
                                System.out.println();
                                System.out.println("Want to Enroll Other Courses....Press 'n'
for No and Press 'y' for Yes");
                                char choice=sc.next().charAt(0);
                                if(choice=='n')
                                {
                                    break;
                                }
                                }
                                }

```

```

public static void ViewEnrolledCourse() throws Exception
{
    PreparedStatement pst;
    String Email_id=StudentOperations.Email_id;
    pst=con.prepareStatement("select
Full_Name,Mobile_No,Email_id,course_name,Duration, course_fee from
Student_Registration s,course c,EnrollCourse e where Email_id=? AND
s.Stud_id=e.Stud_id AND c.Cid=e.Cid");
    pst.setString(1, Email_id);
    ResultSet rs=pst.executeQuery();
    System.out.println("FULL
NAME\tMOBILE_NO\tEMAIL_ID\t\tCOURSE_NAME\tDURATION\tCOURSE
_FEE");

```

```

System.out.println(" ..... "
);
while(rs.next())
{

    System.out.println(rs.getString("Full_Name")+"\t"+rs.getString("Mobile_
No")+"\t"+rs.getString("Email_id")+"\t"+rs.getString("course_name")+"\t"+rs.ge
tString("Duration")+"\t"+rs.getFloat("course_fee"));

}
}

```

```

public static void ViewProfile() throws Exception
{

```

```

    con=DatabaseConnection.getConnection();
    PreparedStatement pst;
    Scanner sc=new Scanner(System.in);
    String Email_id=StudentOperations.Email_id;
    pst=con.prepareStatement("select * from Student_Registration where
Email_id=?");
    pst.setString(1, Email_id);
    ResultSet rs=pst.executeQuery();
    System.out.println("ID\tEMAIL_ADDRESS\tPASSWORD\tNAME\tMO
BILE_NUMBER\tAGE\tADDRESS\t\tDATE_OF\t\tGENDER");
    System.out.println(" .....
.....");
    while(rs.next())
    {

        System.out.println(rs.getInt(1)+"\t"+rs.getString(2)+"\t"+rs.getString(3)+"\t"
+rs.getString(4)+"\t"+rs.getString(5)+"\t"+rs.getInt(6)+"\t"+rs.getString(7)+"\t"+r
s.getString(8)+"\t"+rs.getString(9));
    }
}

```

```

public static void UpdateProfile() throws Exception
{

```

```

    con=DatabaseConnection.getConnection();
    //create a statement object
        Statement st=con.createStatement();
        Scanner sc=new Scanner(System.in);
        String Email_id=StudentOperations.Email_id;
        int ch;
        PreparedStatement pst;

```

```

        pst=con.prepareStatement("select * from Student_Registration
where Email_id=?");
        pst.setString(1, Email_id);
        ResultSet rs=pst.executeQuery();
        if(rs.next())
        {
            while(true)
            {
                System.out.println("Which Feild do you want to
update");

                System.out.println();
                System.out.println("1 UPDATE NAME \n2
UPDATE PASSWORD \n3 UPDATE MOBILE NUMBER \n4 AGE \n5
ADDRESS \n6 DATE_Of_BIRTH \n7 GENDER");

                System.out.println("=====");

                ch=sc.nextInt();

                switch(ch)
                {
                    case 1:
                        System.out.println("enter the Name to
update");

                        Full_Name=sc.next();
                        String str="update Student_Registration set
Full_Name=? where Email_id=?";

                        //System.out.println(str);
                        pst=con.prepareStatement(str);
                        pst.setString(1, Full_Name);
                        pst.setString(2, Email_id);
                        int k=pst.executeUpdate();
                        if(k>0)
                        {
                            System.out.println("Record
Updated Succesfully");
                        }
                        else
                        {
                            System.out.println("Not
Updated...");
                        }
                        break;

//                case 2:

```

```

//                                System.out.println("enter the Email");
//                                String Email=sc.next();
//                                str="update Student_Registration set
Email_id=? where Email_id=?";
//
//                                // System.out.println(str);
//                                pst=con.prepareStatement(str);
//                                pst.setString(1, Email_id);
//                                pst.setInt(2, _id);
//                                k=pst.executeUpdate();
//                                if(k>0)
//                                {
//                                    System.out.println("Record
Updated Succesfully");
//                                }
//                                else
//                                {
//                                    System.out.println("Not
Updated...");
//                                }
//                                break;

case 2:
    System.out.println("enter the Password");
    Password=sc.next();
    str="update Student_Registration set

Password=? where Email_id=?";

    // System.out.println(str);
    pst=con.prepareStatement(str);
    pst.setString(1, Password);
    pst.setString(2, Email_id);
    k=pst.executeUpdate();
    if(k>0)
    {
        System.out.println("Record

Updated Succesfully");

    }
    else
    {
        System.out.println("Not

Updated...");

    }
    break;

```

```

case 3:
    System.out.println("enter the Mobile
number");
    Mobile_No=sc.next();
    str="update Student_Registration set
Mobile_No=? where Email_id=?";

    // System.out.println(str);
    pst=con.prepareStatement(str);
    pst.setString(1, Mobile_No);
    pst.setString(2, Email_id);
    k=pst.executeUpdate();
    if(k>0)
    {
        System.out.println("Record
Updated Succesfully");
    }
    else
    {
        System.out.println("Not
Updated...");
    }
    break;

case 4:
    System.out.println("enter the Age");
    age=sc.nextInt();
    str="update Student_Registration set age=?
where Email_id=?";

    // System.out.println(str);
    pst=con.prepareStatement(str);
    pst.setInt(1, age);
    pst.setString(2, Email_id);
    k=pst.executeUpdate();
    if(k>0)
    {
        System.out.println("Record
Updated Succesfully");
    }
    else
    {
        System.out.println("Not
Updated...");
    }

```



```

        break;

    case 5:
        System.out.println("enter the Address");
        Address=sc.next();
        str="update Student_Registration set
Address=? where Email_id=?";

        // System.out.println(str);
        pst=con.prepareStatement(str);
        pst.setString(1, Address);
        pst.setString(2, Email_id);
        k=pst.executeUpdate();
        if(k>0)
        {
            System.out.println("Record
Updated Succesfully");
        }
        else
        {
            System.out.println("Not
Updated...");
        }
        break;

    case 6:
        System.out.println("enter the
Date_Of_Birth");

        Date_Of_Birth=sc.next();
        str="update Student_Registration set
Date_Of_Birth=? where Email_id=?";

        // System.out.println(str);
        pst=con.prepareStatement(str);
        pst.setString(1,Date_Of_Birth);
        pst.setString(2, Email_id);
        k=pst.executeUpdate();
        if(k>0)
        {
            System.out.println("Record
Updated Succesfully");
        }
        else
        {
            System.out.println("Not
Updated...");
        }

```

```

        }
        break;

    case 7:
        System.out.println("enter the Gender");
        Gender=sc.next();
        str="update Student_Registration set
Gender=? where Email_id=?";

        // System.out.println(str);
        pst=con.prepareStatement(str);
        pst.setString(1,Gender);
        pst.setString(2, Email_id);
        k=pst.executeUpdate();
        if(k>0)
        {
            System.out.println("Record
Updated Succesfully");
        }
        else
        {
            System.out.println("Not
Updated...");
        }
        break;
    }
    System.out.println();
    System.out.println("Want to Update Other
Records...Press 'y' for Yess AND Press 'n' for No");
    char choice=sc.next().charAt(0);
    if(choice=='n')
    {
        break;
    }

}

}

else
{
    System.out.println("ID Not Present");
}

}}

```

Output:-

Student Login:

Student first login the system by entering email Id and Password.

```
<<<<<<<<<WELCOME TO STUDENT MANAGEMENT SYSTEM>>>>>>>>>
```

```
Admin Login Or Student Login
```

```
1.ADMIN LOGIN
```

```
2.STUDENT REGISTRATION
```

```
3.STUDENT LOGIN
```

```
=====
```

```
3
```

```
Enter Email Id:
```

```
Abhi@gmail.com
```

```
Enter the Password:
```

```
Abhi@123
```

```
Login Successfully..!!
```

```
*****Welcome To Student Dashboard*****
```

```
1.VIEW PROFILE
```

```
2.UPDATE PROFILE
```

```
3.ENROLL COURSE
```

```
4.VIEW ENROLL COURSE
```

```
=====
```

Student Registration Table Structure:-

```
mysql> desc student_registration;
```

Field	Type	Null	Key	Default	Extra
Stud_id	int	NO	PRI	NULL	auto_increment
Email_id	varchar(50)	NO	UNI	NULL	
Password	varchar(50)	NO		NULL	
Full_Name	varchar(100)	NO		NULL	
Mobile_No	varchar(50)	NO	UNI	NULL	
age	int	YES		NULL	
Address	varchar(100)	YES		NULL	
Date_Of_Birth	date	YES		NULL	
Gender	varchar(50)	YES		NULL	

9 rows in set (0.00 sec)

```
mysql> select * from student_registration;
```

Stud_id	Email_id	Password	Full_name	Mobile_No	age	Address	Date_Of_Birth	Gender
1	Akshay@gmail.com	Akshay@123	Akshay	9415877526	23	Delhi	1999-10-10	male
2	Nikhil@gmail.com	Nikhil@123	Nikhil	8566974523	23	Pune	1999-05-06	Male
3	Abhi@gmail.com	Abhi@123	Abhi	8965741236	26	Mumbai	1988-09-09	Male
4	Karan@gmail.com	Karan@123	Karan	9874563215	26	Haryana	1999-05-16	Male

4 rows in set (0.00 sec)

Student View Profile:

After login Student can view the Profile.

```
<<<<<<<<WELCOME TO STUDENT MANAGEMENT SYSTEM>>>>>>>>>>>>>>
```

Admin Login Or Student Login

- 1.ADMIN LOGIN
- 2.STUDENT REGISTRATION
- 3.STUDENT LOGIN

=====

3

Enter Email Id:

Abhi@gmail.com

Enter the Password:

Abhi@123

Login Successfully...!!

*****Welcome To Student Dashboard*****

- 1.VIEW PROFILE
- 2.UPDATE PROFILE
- 3.ENROLL COURSE
- 4.VIEW ENROLL COURSE

=====

1

ID	EMAIL_ADDRESS	PASSWORD	NAME	MOBILE_NUMBER	AGE	ADDRESS	DATE_OFB	GENDER
3	Abhi@gmail.com	Abhi@123	Abhi	8965741236	26	Mumbai	1988-09-09	Male

'Student' Do you want to Perform Other Operation Press 'y' for continue AND 'n' for stop

Student Registration table:-

```
mysql> select * from student_registration;
```

Stud_id	Email_id	Password	Full_name	Mobile_No	age	Address	Date_Of_Birth	Gender
1	Akshay@gmail.com	Akshay@123	Akshay	9415877526	23	Delhi	1999-10-10	male
2	Nikhil@gmail.com	Nikhil@123	Nikhil	8566974523	23	Pune	1999-05-06	Male
3	Abhi@gmail.com	Abhi@123	Abhi	8965741236	26	Mumbai	1988-09-09	Male
4	Karan@gmail.com	Karan@123	Karan	9874563215	26	Haryana	1999-05-16	Male

```
4 rows in set (0.00 sec)
```

Student Update Profile:

After login Student can update profile based on name, address, age, password, mobile no etc.

```
1.VIEW PROFILE
2.UPDATE PROFILE
3.ENROLL COURSE
4.VIEW ENROLL COURSE
=====
```

```
2
```

```
Which Feild do you want to update
```

```
1 UPDATE NAME
2 UPDATE PASSWORD
3 UPDATE MOBILE NUMBER
4 AGE
5 ADDRESS
6 DATE_OF_BIRTH
7 GENDER
=====
```

```
4
```

```
enter the Age
```

```
25
```

```
Record Updated Succesfully
```

```
Want to Update Other Records...Press 'y' for Yess AND Press 'n' for No
```

```
y
```

```
Which Feild do you want to update
```

```
1 UPDATE NAME
2 UPDATE PASSWORD
3 UPDATE MOBILE NUMBER
4 AGE
5 ADDRESS
6 DATE_OF_BIRTH
7 GENDER
=====
```

```
5
```

```
enter the Address
```

```
Chitrakoot
```

```
Record Updated Succesfully
```

Student Enroll course:

After login Student can Enroll the course which is added by only Admin.

```
<<<<<<<<<WELCOME TO STUDENT MANAGEMENT SYSTEM>>>>>>>>>
```

```
Admin Login Or Student Login
```

- 1.ADMIN LOGIN
- 2.STUDENT REGISTRATION
- 3.STUDENT LOGIN

```
=====
```

```
3
```

```
Enter Email Id:
```

```
Abhi@gmail.com
```

```
Enter the Password:
```

```
Abhi@123
```

```
Login Successfully..!!
```

```
*****Welcome To Student Dashboard*****
```

- 1.VIEW PROFILE
- 2.UPDATE PROFILE
- 3.ENROLL COURSE
- 4.VIEW ENROLL COURSE

```
=====
```

```
3
```

```
Which Course Do You Want To Do
```

ID	CName	Duration	Fee

111	Java	48:00:00	1000.0
222	python	36:00:00	2000.0
333	Aws	48:00:00	1000.0
444	Nodejs	48:00:00	2000.0

```
Enter Your choice:
```

```
444
```

```
Pay course fee
```

```
2000
```

```
Course Enroll Successfully.....!!
```

```
Want to Enroll Other Courses....Press 'n' for No and Press 'y' for Yes
```

Enroll Course table in Mysql:-

```
mysql> select * from enrollcourse;
+-----+-----+-----+-----+
| ceID | Stud_id | Cid  | fees |
+-----+-----+-----+-----+
| 1    | 1       | 111  | 1000 |
| 2    | 2       | 222  | 2000 |
| 3    | 3       | 333  | 1000 |
| 4    | 4       | 444  | 2000 |
| 5    | 3       | 444  | 2000 |
+-----+-----+-----+-----+
5 rows in set (0.00 sec)
```

```
mysql> select * from course;
+-----+-----+-----+-----+
| Cid | course_name | Duration | course_fee |
+-----+-----+-----+-----+
| 111 | Java        | 48:00:00 | 1000       |
| 222 | python      | 36:00:00 | 2000       |
| 333 | Aws         | 48:00:00 | 1000       |
| 444 | Nodejs      | 48:00:00 | 2000       |
+-----+-----+-----+-----+
4 rows in set (0.00 sec)
```


Student view Enroll course:

After login Student can Enroll the course which is added by only Admin. And can view the enroll course.

```
<<<<<<<<<WELCOME TO STUDENT MANAGEMENT SYSTEM>>>>>>>>>>>>
```

```
Admin Login Or Student Login
```

- 1.ADMIN LOGIN
- 2.STUDENT REGISTRATION
- 3.STUDENT LOGIN

```
=====
```

```
3
```

```
Enter Email Id:
```

```
Abhi@gmail.com
```

```
Enter the Password:
```

```
Abhi@123
```

```
Login Successfully..!!
```

```
*****Welcome To Student Dashboard*****
```

- 1.VIEW PROFILE
- 2.UPDATE PROFILE
- 3.ENROLL COURSE
- 4.VIEW ENROLL COURSE

```
=====
```

```
4
```

FULL NAME	MOBILE_NO	EMAIL_ID	COURSE_NAME	DURATION	COURSE_FEE
Abhi	8965741236	Abhi@gmail.com	Aws	48:00:00	1000.0
Abhi	8965741236	Abhi@gmail.com	Nodejs	48:00:00	2000.0

```
'Student' Do you want to Perform Other Operation Press 'y' for continue AND 'n' for stop
```

Course Table In Mysql:-

```
mysql> desc course;
```

Field	Type	Null	Key	Default	Extra
Cid	int	NO	PRI	NULL	
course_name	varchar(100)	NO		NULL	
Duration	time	YES		NULL	
course_fee	float	YES		NULL	

4 rows in set (0.00 sec)

```
mysql> select * from course;
```

Cid	course_name	Duration	course_fee
111	Java	48:00:00	1000
222	python	36:00:00	2000
333	Aws	48:00:00	1000
444	Nodejs	48:00:00	2000

4 rows in set (0.00 sec)

CONCLUSION:-

- ✓ By using core java and making a database connection, a student management system is made and executed.
- ✓ So with the help of admin and student all the operations can be performed and every piece of information like admin data, student data, course data, enrolled course data are stored in the mysql database.
- ✓ Finally in a secured manner, after registering and logging in, the student can enroll into a course without any interruptions.