**CORE JAVA PROJECT**

**PROJECT TITLE: Student Management System**

**AIM:**

That maintains, and manages all of the data and perform operations with easy and secured.

**LANGUAGES:**

JAVA and SQL

**SOFTWARE REQUIREMENTS:**

Eclipse for java.

MySQL for sql**.**

**SYNOPSIS:**

In this Student Management System 5 main operations are there.

That MAIN OPERATIONS are:

1.REGISTRATION --> Student can perform that operation.

2.LOGIN --> Admin can perform that operation.

3. UPDATE -->Admin and Student can perform that operation.

4.DELETE --> only Admin can perform that operation

5.DISPLAY -->Both the admin and Student perform the operation.

The control is given for both the admin and also user. Admin can perform two operations one is the fetching information about the student who all are the confirm admission and second one is admin can able the student how many students confirm.

**INITIAL SETUP:**

**(IN JAVA):**

**Mavan Project Name**: Student Management System Project.

**Package Name:** com.project

**Classes Name:** 1. ProjectMain

2.ProjectConnections

3.ProjectOperations

**(IN DATABASE):**

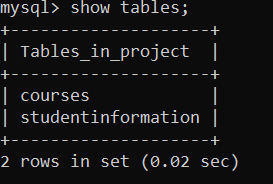
Creating database in sql (query)



**Use Databse :**



**Tables Creation:**



**Student Portal:**

**ADMIN AND USER MAIN METHOD CODE:**

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*CORE JAVA PROJECT\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

CSR CAPGEMINI TRAINING PROJECT

EDUBRIDGE INDIA PRIVATE LIMITED

PROJECT TITLE: SUDENT MANAGEMENT SYSTEM

UNDER THE GUIDENCE OF TRAINER MRS.INDRAKA MALI

@DONE BY Suresh Nanjappan

In STUDENT MANAGEMENT SYSTEM:

MAIN OPERATIONS:

1.REGISTRATION --> Student can perform that operation.

2.LOGIN --> Admin can perform that operation.

3. UPDATE -->Admin and Student can perform that operation.

4.DELETE --> only Admin can perform that operation

5.DISPLAY -->Both the admin and Student perform the operation

Student Main class is having two major operations

This will decide who is going to perform the operation

1. Admin

2. Student

case 1:

If it is admin choice means it gives two choices for admin

1. Admin Login --> it allows to login the admin account

2. Admin operations --> Doing Operations

Case 2:

If it is user choice means it gives two choices for user

1. User Registration --> it allows to registration for user

2. User Operations --> Doing Operation

package in.neha;

import java.sql.PreparedStatement;

import java.sql.ResultSet;

import java.sql.SQLException;

import java.util.Scanner;

public class ProjectMain {

public static void main(String[] args) throws SQLException {

String uname = "admin";

String upass = "admin@123";

int num;

char ch;

Scanner sc = new Scanner(System.in);

System.out.println(" WELCOME TO EDUBRIDGE INDIA PVT LTD");

System.out.println("1.ADMIN PORTAL");

System.out.println("2.STUDENT PORTAL");

System.out.println("... YOU ARE ...");

num=sc.nextInt();

switch(num) {

case 1: // Admin Login

System.out.println("For Admin Login Enter the USERNAME & USERPASSWORD");

System.out.println("ENTER ADMIN NAME");

String un = sc.next();

System.out.println("ENTER ADMIN PASSWORD");

String up = sc.next();

if(un.equals("admin") && up.equals("admin@123")) {

while(true) {

System.out.println("\*\*\*\*\*\*\*\*\*\* ADMIN PORTAL \*\*\*\*\*\*\*\*\*\*");

System.out.println(“\_\_\_\_\_HERE ADMIN OPERATIONS\_\_\_\_");

System.out.println("1. Show All Student Details");

System.out.println("2. Delete Student Records");

System.out.println("3. Update Student Details");

System.out.println(" Select Any ");

int adminch=sc.nextInt();

switch(adminch) {

case 1: // Display Student

ProjectOperations.displayAllStudent();

break;

case 2: // Delete Students

ProjectOperations.deleteStudent();

break;

case 3: // Update Students

ProjectOperations.updateStudent();

break;

default: System.out.println("Invalid Admin Option");

}

System.out.println("Do want to continue y/n");

ch=sc.next().charAt(0);

if(ch=='n' || ch=='N')

break;

}

}

else {

System.out.println("Invalid!!!!!!!!!!!!!");

System.out.println("YOU ARE NOT ADMIN");

}

break;

// STUDENT LOGIN

case 2:

System.out.println(" \*\*\*\*\*\*\*\*\*\* STUDENT PORTAL \*\*\*\*\*\*\*\*\*\* ");

System.out.println(" 1. You Are Already Registerd ");

System.out.println(" 2. You Are New User ");

System.out.println(" Choose Correct Option ");

int chr = sc.nextInt();

if(chr==1) {

System.out.println("Enter Student\_Id");

int Student\_Id= sc.nextInt();

String sel = "select \* from StudentInformation where Student\_Id"+Student\_Id;

PreparedStatement pst = null;

ResultSet rs = pst.executeQuery();

if (rs.next()) {

System.out.println(" 2. Show Your Record ");

System.out.println(" 3. Update Your Record ");

int studentch = sc.nextInt();

}

else {

System.out.println("Invalid Id!!!!!");

}

}

if(chr==2) {

System.out.println("HERE ARE COURSES");

System.out.println("Enroll Now");

System.out.println("\_\_\_\_\_\_\_\_\_\_COURSE LIST\_\_\_\_\_\_\_\_\_");

System.out.println(" FULL COUSE DETAILS ");

ProjectOperations.courseList();

System.out.println("DO U WANT TO ENROLL FOR THIS COURSE (Y/N)");

ch=sc.next().charAt(0);

if(ch == 'n' || ch == 'N');

break;

if(ch == 'y' || ch == 'Y')

System.out.println(" 1. Enroll for Course");

int studentch = sc.nextInt();

switch(studentch) {

case 1:// Insert Record

ProjectOperations.insetStudent();

break;

case 2: // Show particular Record

ProjectOperations.displayParticularStudent();

break;

case 3: // update only Your information

ProjectOperations.updateParticularStudent();

break;

default: System.out.println(" Invalid Student Operation ");

}

}

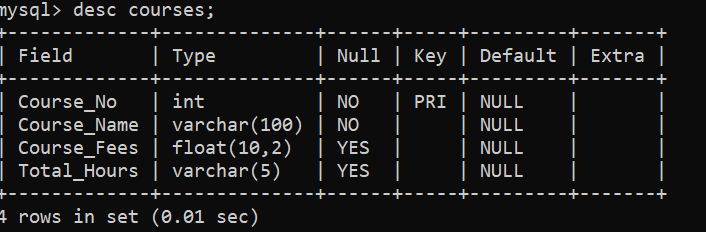
}

}

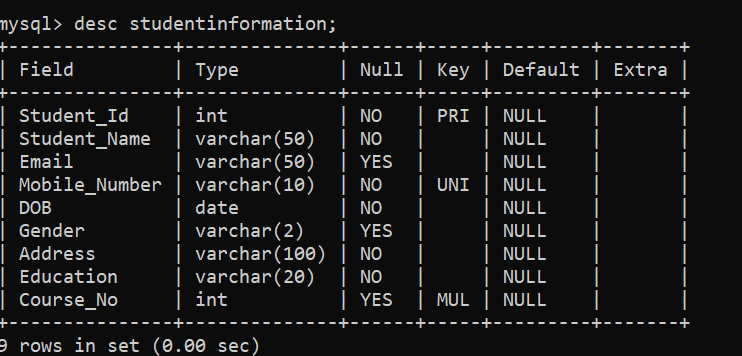
}

**Database table creation:**

**Course Table:**



**Student Information Table:**



**DATABASE CONNECTION CLASS :**

Data base connection class-> it makes a connection between the spring tool shoot and MySQL

1.Driver->That implements the java databases connectivity (JDBC) API

2.url->data base management system jdbc driver uses to connect to a database

3.username-> user name of the database

4.password->pass word for the databases

5.forname()-> method is loading the driver dynamically loads a java class at runtime

6.DriverManger->Is that class making connection to database by passing arguments as a url, username and password

package in.neha;

import java.sql.Connection;

import java.sql.DriverManager;

public class Project Connections {

private static Connection myconn;

private static String driver = "com.mysql.cj.jdbc.Driver";

private static String url = "jdbc:mysql://localhost:3306/project";

private static String un = "root";

private static String up = "root";

public static Connection getConnections() {

try {

Class.forName(driver);

myconn = DriverManager.getConnection(url,un,up);

if(myconn == null) {

System.out.println("Connection Error");

}

}

catch(Exception e) {

e.printStackTrace();

}

return myconn;

}

}

**REGISTRATION PROCESSES :**

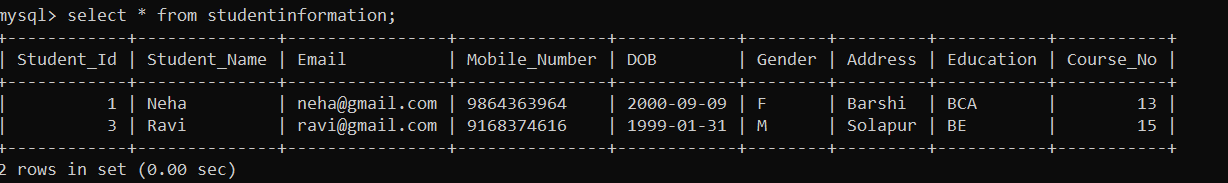
**REGISTRATION CODE :**

**FOR STUDENT:**

The Student registration is done by back-end process. Reason is only one its store permantely and secure and we can easily performs the operation and make some changes.

**Database output:**

**Student ragistration Record is inserted in database:**



**DATABASE OPERATION CLASS:**

package in.neha;

import java.sql.Connection;

import java.sql.Date;

import java.sql.PreparedStatement;

import java.sql.ResultSet;

import java.sql.SQLException;

import java.util.Scanner;

public class ProjectOperations {

private static Connection myconn;

private static PreparedStatement pst=null;

private static ResultSet rs;

//Course List

public static void courseList() throws SQLException {

myconn=ProjectConnections.getConnections();

String selsql="select \* from Courses";

pst=myconn.prepareStatement(selsql);

ResultSet rs = pst.executeQuery();

System.out.println("EDUBRIDGE COURSE LIST");

System.out.println("Course\_No \t Course\_Fees \t Total\_Hours \t Course\_Name");

System.out.println("\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_");

while(rs.next()) {

int no=rs.getInt("Course\_No");

String cn=rs.getString("Course\_Name");

int fee=rs.getInt("Course\_Fees");

String time = rs.getNString("Total\_Hours");

System.out.println(no+"\t\t"+fee+"\t\t"+time+"\t\t"+cn);

}

}

// For Admin

//Display

public static void displayAllStudent() throws SQLException {

myconn=ProjectConnections.getConnections;

String selsql="select \* from studentinformation";

pst=myconn.prepareStatement(selsql);

rs=pst.executeQuery();

System.out.println("Student\_Id \t Student\_Name \t Mobile\_Number \t DOB \t Gender \t Address \t Email \t Education \t Course\_No");

while(rs.next()) {

int id = rs.getInt("Student\_Id");

String sn = rs.getString("student\_Name");

String mail = rs.getString("Email");

String mono = rs.getString("Mobile\_Number");

String DOB = rs.getString("DOB");

String gen = rs.getString("Gender");

String add = rs.getString("Address");

String edu = rs.getString("Education");

int cno = rs.getInt("Course\_No"); System.out.println(id+"\t\t"+sn+"\t\t"+mono+"\t"+DOB+"\t"+gen+"\t"+add+"\t\t"+mail+"\t\t"+edu+"\t\t"+cno);

}

}

//Delete

public static void deleteStudent() throws SQLException {

myconn=ProjectConnections.getConnections();

int id;

Scanner sc=new Scanner(System.in);

System.out.println("Enter Student\_Id to delete record");

id=sc.nextInt();

String sel="select \* from StudentInformation where Student\_Id=?";

pst=myconn.prepareStatement(sel);

pst.setInt(1, id);

rs=pst.executeQuery();

if(rs.next()) {

String del="delete from StudentInformation where Student\_Id=?";

pst=myconn.prepareStatement(del);

pst.setInt(1,id);

int rv = pst.executeUpdate();

if(rv >0) {

System.out.println(" Record is Deleted ");

}

else {

System.out.println(" ERROR!!!!! ");

}

}

else {

System.out.println("Student\_Id "+id+" not exists")

}

}

//Update

public static void updateStudent() throws SQLException {

myconn=ProjectConnections.getConnections();

int id;

Scanner sc=new Scanner(System.in);

System.out.println("Enter Student\_id to update record");

id=sc.nextInt();

System.out.println("1.Change Your Mobile\_number");

String mo=sc.next();

System.out.println("Change Your Address");

String ad = sc.next();

String sel="select \* from studentinformation where Student\_Id=?";

pst=myconn.prepareStatement(sel);

pst.setInt(1,id);

rs=pst.executeQuery();

if(rs.next()) {

String upd="update studentinformation set Mobile\_number=?, Address=? where Student\_Id=?";

pst.syconn.prepareStatement(upd);

pst.setString(1, mo);

pst.setString(2,add);

pst.setStringetInt(3, id);

int rv=pst.executeUpdate();

if(rv>0) {

System.out.println("Record is Updated");

}

else {

System.out.println("ERRROR!!!!!!");

}

}

else {

System.out.println(id+" does not exists");

}

}

// For Student

//Display Particular Record

public static void displayParticularStudent() throws SQLException {

myconn=ProjectConnections.getConnections();

Scanner sc = new Scanner(System.in);

System.out.println("Enter Student\_id to See Your Record");

int id = sc.nextInt();

String sel="select \* from studentinformation where Student\_id=?";

pst=myconn.prepareStatement(sel);

rs=pst.executeQuery();

if(rs.next()) {

System.out.println("Student\_Id \t Student\_Name \t Email \t Mobile\_Number \t DOB \t Gender \t Address \t Education");

int idd = rs.getInt("Student\_Id");

String name = rs.getNString("Student\_Name");

String email = rs.getNString("Email");

String mono = rs.getNString("Mobile\_Number");

String DOB = rs.getNString("DOB");

String gender = rs.getNString("Gender");

String address = rs.getNString("Address");

String education = rs.getNString("Education");

System.out.println(idd+"\t"+name+"\t"+email+"\t"+mono+"\t"+gender+"\t"+address+"\t"+education);

}

}

//Update Particular Record

public static void updateParticularStudent() throws SQLException {

int id;

String name;

Scanner sc = new Scanner(System.in);

System.out.println("Enter Student\_id to update record");

id=sc.nextInt();

String sel = "select \* from studentinformation where Student\_Id=?";

pst=myconn.prepareStatement(sel);

pst.setInt(1,id);

rs=pst.executeQuery();

if(rs.next()) {

String upd="Update studentinformation set Email=?, MoNo=?, Address=? where Student\_Id=?";

pst = myconn.prepareStatement(upd);

pst.setInt(1, id);

int rs=pst.executeUpdate();

if(rs>0) {

System.out.println("Record is Updated");

}

else {

System.out.println("ERRROR!!!!!");

}

}

else {

System.out.println("Student\_Id"+id+" not exists");

}

}

// Insert Record

public static void insetStudent() throws SQLException {

myconn=ProjectConnections.getConnections();

int id;

String name;

String mail;

String mono;

String DOB ;

String gender;

String address;

String edu;

int no = 0;

Scanner sc=new Scanner(System.in);

System.out.println("Enter Your Student Id");

id = sc.nextInt();

System.out.println(“Enter Your Full Name");

name = sc.next();

System.out.println("Enter Your EMAIL ID");

mail = sc.next();

System.out.println("Enter Your Mobile Number");

mono = sc.next();

System.out.println("Your DOB (YYYY-MM-DD)");

DOB = sc.next();

System.out.println("Gender (M/F)");

gender = sc.next();

System.out.println("Enter Your Current Address");

address = sc.next();

System.out.println("Edcucation");

edu = sc.next();

System.out.println("Enter Course\_No Which One You Choose");

no = sc.nextInt();

String s="insert into StudentInformation values(?,?,?,?,?,?,?,?,?)";

pst=myconn.prepareStatement(s);

pst.setInt(1, id);

pst.setString(2, name);

pst.setString(3, mail);

pst.setString(4, mono);

pst.setString(5, DOB);

pst.setString(6, gender);

pst.setString(7, address);

pst.setString(8, edu);

pst.setInt(9,no);

int rv=pst.executeUpdate();

if(rv>0) {

System.out.println("HOORAY!!!!!!ADMISSION CONFIRMED");

}

else {

System.out.println("Student\_Id"+id+" not exists");

}

}

}