Green University of Bangladesh

Department of Computer Science and Engineering (CSE)
Faculty of Sciences and Engineering (FSE)
Semester: (Fall, Year: 2024), B.Sc. in CSE (Day)

University Management System

Course Title: Database System Lab
Course Code: CSE-210
Section: 223-D7

Student's Details

Name	ID
Md. Zehadul Islam	**222902069**

Submission Date: 12.06.2024
Course Teacher's Name: Umme Habiba

[For teachers use only: Don't write anything inside this box]

Project Report Status	
Marks:	**Signature:**
Comments:	**Date:**

Contents

Chapter 1

Introduction

1.1 Overview

A University Management System in Java using MySQL helps manage student and staff records, enter marks, handle examination details, update teacher and student details, manage courses, grades, student leave, teacher leave, and course fees. It also efficiently manages course admissions. The system includes functionalities such as registration, exam handling, and displaying details. It improves data organization and accessibility, making administrative tasks easier and faster for university staff and students. [1] [2] [3] [4]

1.2 Motivation

Building a University Management System in Java with MySQL simplifies administrative tasks for staff and students. By organizing student and staff records, managing exams, and handling course admissions, it streamlines processes, making them more efficient. With easy access to information and streamlined functionalities like registration and fee management, the system enhances productivity, ensuring smoother operations within the university environment.

1.3 Problem Definition

1.3.1 Problem Statement

The current university administration lacks an efficient system for managing student and staff records, handling exams, admissions, and fee management. This leads to disorganized data, delays in administrative tasks, and inefficiencies in managing academic_processes.

1.3.2 Complex Engineering Problem

A University Management System involves applying MySQL database expertise to design and implement a robust system. This necessitates understanding relational databases, normalization, and transaction management for seamless operations.

Table 1.1: Summary of the attributes touched by the mentioned projects.

Name of the P Attributess	**Explain how to address**
P1: Depth of knowledge required	For the University Management System project, thorough comprehension of relational databases, normalization, and transaction management is crucial. Proficiency in MySQL will be employed to ensure effective data storage and retrieval mechanisms.
P2: Range of conflicting requirements	Navigating a range of conflicting requirements is essential, balancing the diverse needs of stakeholders while maintaining system coherence and functionality in the University Management System project.
P3: Depth of analysis required	Thorough analysis is necessary to craft a normalized database structure, enhance query speed, and guarantee efficient storage of transactional data for optimal performance.
P4: Familiarity of issues	The primary challenge lies in crafting a database schema that is both secure and scalable. This entails safeguarding sensitive user data while optimizing performance through database normalization and indexing principles.
P5: Extent of applicable codes	The database queries and stored procedures will be expertly crafted, reusable, and highly efficient. Our codebase will strictly adhere to industry best practices in database development.
P6: Extent of stakeholder involvement and conflicting requirements	Engaging with stakeholders to grasp their data access requirements, ensuring the database schema satisfies security and performance criteria.
P7: Interdependence	Making sure the Java application works well with the database design, so the front and back parts fit together smoothly.

1.4 Design Goals/Objectives

The main aim of the project's database part is to create, set up, and make better a MySQL database for the Internet Banking System. The main targets are:

- Making a neat database layout to keep data consistent and avoid repetition.
- Adding safety measures to store sensitive details securely.
- Speeding up database searches and transactions for better performance.
- Following industry rules and the best ways to manage databases.

1.5 Application

The "University Management System" project offers practical solutions benefiting educational institutions, faculty and students through streamlined processes and efficient data management.

Admin Login

Access granted: Admin login verified, enabling privileged system control and management functionalities.

New student Information

The admin interface facilitates adding new student information, managing fees, storing student details, displaying results, and facilitating actions like semester drop/add and course enrollment. It ensures seamless administrative control over student-related processes in adherence to admission guidelines.

New Faculty Information

The admin interface allows for adding new faculty information, managing their details, assigning courses, and tracking their leave and performance, ensuring efficient faculty management within the university system.

Examination Information

Manage exam schedules, student marks, and result publication for efficient examination oversight.

Chapter 2

Design/Development/Implementation of the Project

2.1 Introduction

Welcome to our “University Management System”, a secure and convenient platform for all academic needs. This section explores the design, development, and implementation of the University Management System database project. The key to a secure, efficient, and reliable university management experience is the successful integration of MySQL as the database management system. In today’s fast-paced academic environment, we recognize the importance of providing seamless administrative services accessible anytime, anywhere. Our system manages student and faculty information, course enrollment, fee payments, and examination details efficiently. Experience streamlined administration, advanced security, and exceptional service. Join us today and simplify university management.

2.2 Project Details

2.2.1 Database Management

The database is crucial for ensuring the functionality and reliability of our University Management System. Database management tasks include maintaining student and staff records, academic history, and overall system efficiency**.** The Admin Panel work is:

- **Student Management:** The database stores student information, including details collected during the registration process. It also manages student results, handles semester drops, and tracks active semesters.
- **Faculty Management:** Faculty accounts, including activation or deactivation and other account-related tasks, are managed and updated in the database.
- **Performance Monitoring:** Database tables store metrics such as response time, resource utilization, and transaction throughput to identify and resolve performance bottlenecks..

- **Database Management:** The database administrator is responsible for tasks such as backups, optimization, and monitoring to ensure efficient system performance.

2.2.2 Data Integrity Measures

Ensuring data integrity is a fundamental aspect of database design. The University Management System employs the following measures:

- **Triggers:** Triggers are employed in the salary management system within the teacher table to automatically execute salary-related tasks.

- **Primary Key Constraints:** Used to uniquely identify records in tables, primary keys ensure each row has a distinct identifier, such as student IDs and teacher IDs.

- **Foreign Key Constraints:** Establish relationships between tables; for example, student and teacher IDs are used as foreign keys in the student leave and teacher leave tables, preventing orphaned records and enhancing database coherence.

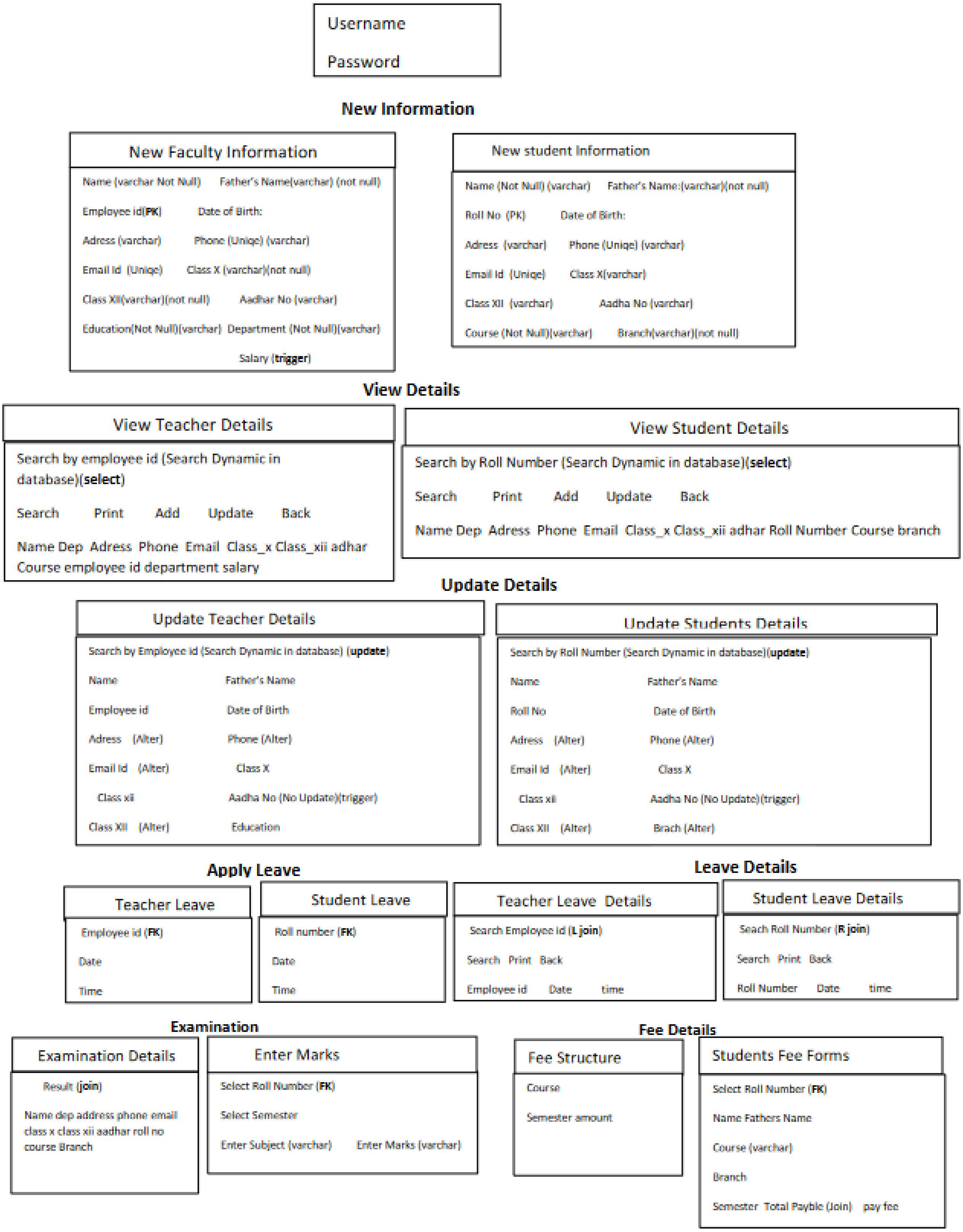
- **Join Operations:** Right joins are used to retrieve student leave details by connecting roll numbers in the leave table with the student table. Similarly, left joins are used to link teacher leave details with the teacher table. Results are also displayed using left joins to connect related tables.

- **Event-Based Mechanisms:** The system leverages event-based mechanisms to automatically execute SQL statements at specified intervals, aiding in tasks such as database maintenance and scheduled operations.

- **Auto-Increment Attributes:** Particularly useful for primary key fields, auto-increment attributes ensure that each new record is automatically assigned a unique identifier, streamlining data insertion.

2.2.3 Admin Section Functionality

Increment I have created a University management system project. After joining the admin section, he can do many functions. He can manage the project student, faculty, semester drop, admission work, semester fee. There are also some more function. Where you want notepad note calculator work about it can take exit from the project. I have given its Schema below:



Username

Password

New Information

New Faculty Information

Name (varchar Not Null) Father's Name(varchar) (not null)

Employee id(**PK**) Date of Birth:

Adress (varchar) Phone (Uniqe) (varchar)

Email Id (Uniqe) Class X (varchar)(not null)

Class XII(varchar)(not null) Aadhar No (varchar)

Education(Not Null)(varchar) Department (Not Null)(varchar)

Salary (**trigger**)

New student Information

Name (Not Null) (varchar) Father's Name:(varchar)(not null)

Roll No (PK) Date of Birth:

Adress (varchar) Phone (Uniqe) (varchar)

Email Id (Uniqe) Class X(varchar)

Class XII (varchar) Aadha No (varchar)

Course (Not Null)(varchar) Branch(varchar)(not null)

View Details

View Teacher Details

Search by employee id (Search Dynamic in database)(**select**)

Search Print Add Update Back

Name Dep Adress Phone Email Class_x Class_xii adhar Course employee id department salary

View Student Details

Search by Roll Number (Search Dynamic in database)(**select**)

Search Print Add Update Back

Name Dep Adress Phone Email Class_x Class_xii adhar Roll Number Course branch

Update Details

Update Teacher Details

Search by Employee id (Search Dynamic in database) (**update**)

Name Father's Name

Employee id Date of Birth

Adress (Alter) Phone (Alter)

Email Id (Alter) Class X

Class xii Aadha No (No Update)(trigger)

Class XII (Alter) Education

Update Students Details

Search by Roll Number (Search Dynamic in database)(**update**)

Name Father's Name

Roll No Date of Birth

Adress (Alter) Phone (Alter)

Email Id (Alter) Class X

Class xii Aadha No (No Update)(trigger)

Class XII (Alter) Brach (Alter)

Apply Leave

Teacher Leave

Employee id (**FK**)

Date

Time

Student Leave

Roll number (**FK**)

Date

Time

Leave Details

Teacher Leave Details

Search Employee id (**L join**)

Search Print Back

Employee id Date time

Student Leave Details

Seach Roll Number (**R join**)

Search Print Back

Roll Number Date time

Examination

Examination Details

Result (**join**)

Name dep address phone email class x class xii aadhar roll no course Branch

Enter Marks

Select Roll Number (**FK**)

Select Semester

Enter Subject (varchar) Enter Marks (varchar)

Fee Details

Fee Structure

Course

Semester amount

Students Fee Forms

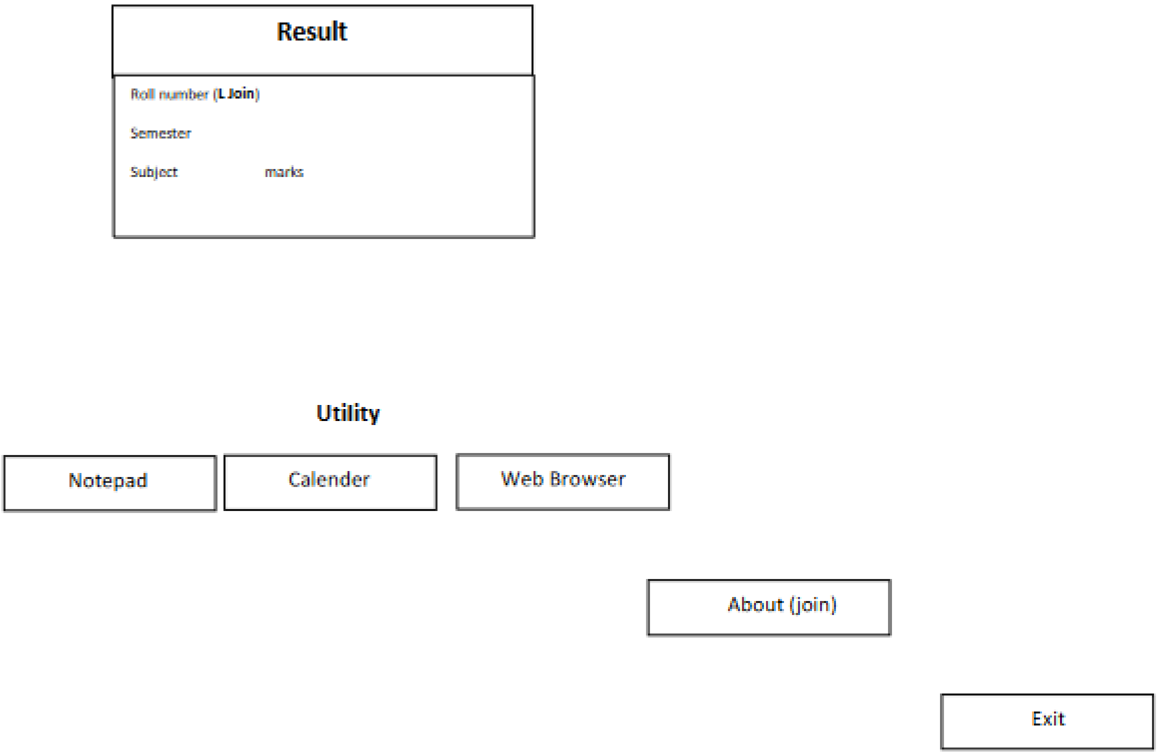
Select Roll Number (**FK**)

Name Fathers Name

Course (varchar)

Branch

Semester Total Payble (Join) pay fee



2.1 Implementation

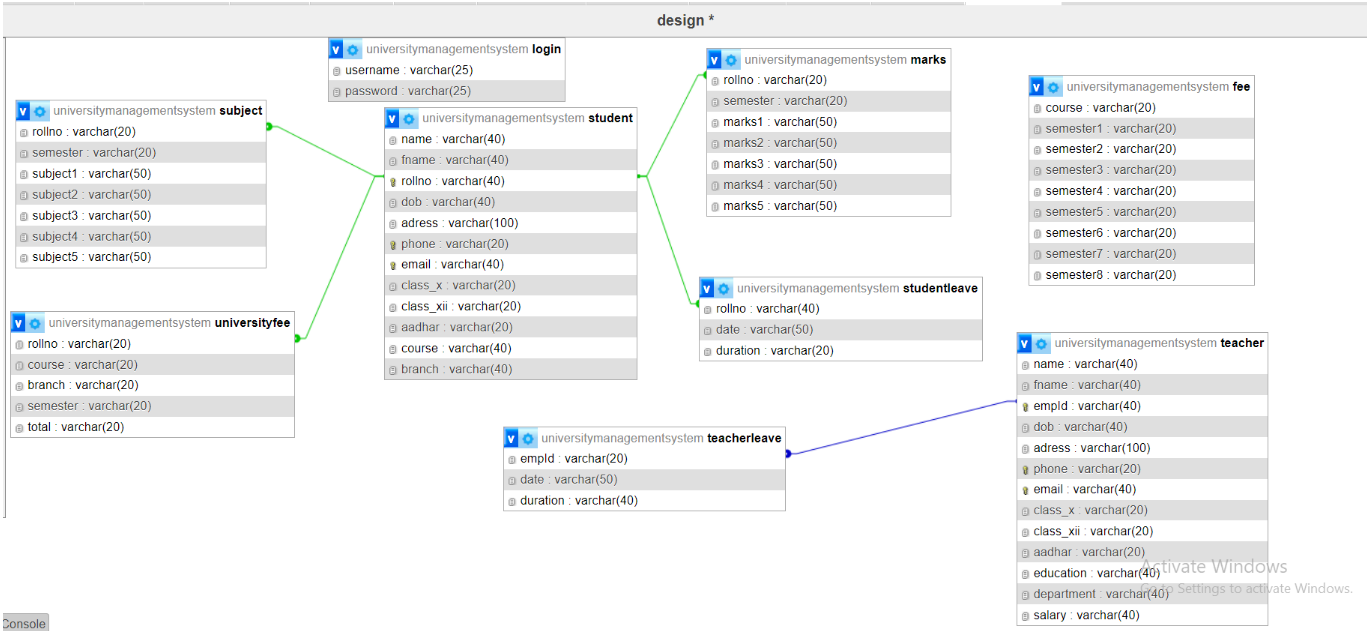
All the implementation details of our project are included in this section, along with many subsections.

2.1.1 Database Connectivity

Integration with MySQL is achieved through JDBC (Java Database Connectivity). The application establishes a connection pool to efficiently manage database connections and minimize overhead. [5]

```java
package university.management.system;

import java.sql.*;

public class Conn {

    Connection c;
    Statement s;

    Conn () {
        try {
            Class.forName("com.mysql.cj.jdbc.Driver");
            c = DriverManager.getConnection("jdbc:mysql://localhost:3307/universitymanagementsystem", "root", "");
            s = c.createStatement();

        } catch (Exception e) {
            e.printStackTrace();
        }
    }
}
```

ER Diagram :



universitymanagementsystem **login**
username : varchar(25)
password : varchar(25)

universitymanagementsystem **subject**
rollno : varchar(20)
semester : varchar(20)
subject1 : varchar(50)
subject2 : varchar(50)
subject3 : varchar(50)
subject4 : varchar(50)
subject5 : varchar(50)

universitymanagementsystem **student**
name : varchar(40)
fname : varchar(40)
rollno : varchar(40)
dob : varchar(40)
adress : varchar(100)
phone : varchar(20)
email : varchar(40)
class_x : varchar(20)
class_xii : varchar(20)
aadhar : varchar(20)
course : varchar(40)
branch : varchar(40)

universitymanagementsystem **universityfee**
rollno : varchar(20)
course : varchar(20)
branch : varchar(20)
semester : varchar(20)
total : varchar(20)

universitymanagementsystem **marks**
rollno : varchar(20)
semester : varchar(20)
marks1 : varchar(50)
marks2 : varchar(50)
marks3 : varchar(50)
marks4 : varchar(50)
marks5 : varchar(50)

universitymanagementsystem **studentleave**
rollno : varchar(40)
date : varchar(50)
duration : varchar(20)

universitymanagementsystem **fee**
course : varchar(20)
semester1 : varchar(20)
semester2 : varchar(20)
semester3 : varchar(20)
semester4 : varchar(20)
semester5 : varchar(20)
semester6 : varchar(20)
semester7 : varchar(20)
semester8 : varchar(20)

universitymanagementsystem **teacherleave**
empId : varchar(20)
date : varchar(50)
duration : varchar(40)

universitymanagementsystem **teacher**
name : varchar(40)
fname : varchar(40)
empId : varchar(40)
dob : varchar(40)
adress : varchar(100)
phone : varchar(20)
email : varchar(40)
class_x : varchar(20)
class_xii : varchar(20)
aadhar : varchar(20)
education : varchar(40)
department : varchar(40)
salary : varchar(40)

Let's highlight all the work between the relationships that have been created,

2.2 Algorithms

1. Initialize the system

- Create an Admin account with a unique username and password.
- Create necessary data structures, such as lists or databases, to store user accounts, transaction history, and other relevant information.

2. Admin Page

- Admin logs in using their username and password.
- Admin, If your password is wrong it will declare wrong
- If your admin password is correct then you can go to main page

3. New Information

- New information does two things. One Add student entry and Second Add teacher entry.
- Insert student details and teacher details into the database.
- Auto-Generate Unique Student and Teacher ID.

4. Apply Leave

- If a student drops a semester, it can be removed if running semester and displays updated status.
- Work updates: half-day or full-day off will be notified.
- Semester drop dates for students and leave dates for teachers will be recorded and maintained.

5. Examination

- Student marks will be recorded by subject and semester according to student roll number.
- Display will be student all details
- Roll based students can see the results.

6. Update Details

- Update details by selecting the student's roll number.
- Update details by selecting the employee roll number.
- Details can be viewed separately for students and faculty in the 'View Details' section.

7. Fee Details

- Database stores limited-semester courses like BTack, Bsc, Msc, and others. See the fee structure
- Students will select the semester according to the course
- Students submit course fees based on roll number; fees calculated automatically according to fee structure

Chapter 3

Performance Evaluation

3.1 Simulation Environment/ Simulation Procedure

For our "University management system" project, I need to install and configure a few components. Here's a step-by-step guide to the experimental setup and environment installation:

3.1.1 Software Installation

- Install MySQL as my database management system.
- Downloaded and included "Xampp" for Java-MySQL interaction.
- Installed Java Development Kit (JDK) from the official Oracle website.
- Chose and installed Apache NetBeans IDE for Java development from the official NetBeans website. https:netbeans.apache.org
- Considered using frameworks like Java Swing/AWT for advanced functionalities.

3.1.2 Project Configuration

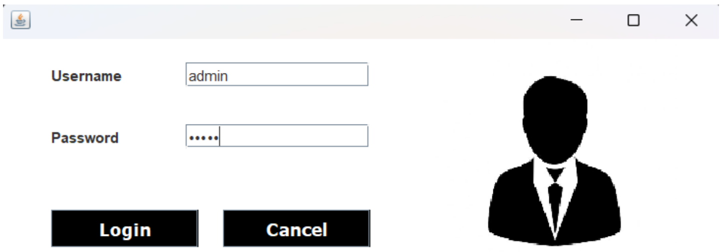
- **Set Up Database Connection**: Configured the "Xampp to establish a connection between the Java application and the MySQL database.
- **Design User Interface**: Utilized Java Swing / AWT frameworks to create an intuitive user interface for the "Internet Banking System".
- **Implement User Functionality**: Coded the functionalities required for user operations such as depositing money, withdrawing money, fund transfers, and ATM card services with OTP verification.
- **Create Admin Functionality:** Implemented the admin features allowing account approval, access to user details, and transaction history.

3.2 Results Analysis/Testing

In the "University Management System" project, main sector is working on student all details and teacher details. An Admin will have to add students and teachers, show details. Updating details. Working on student results. A university can be run very nicely be doing everything like collecting the student semester fee.

3.3 Admin Section

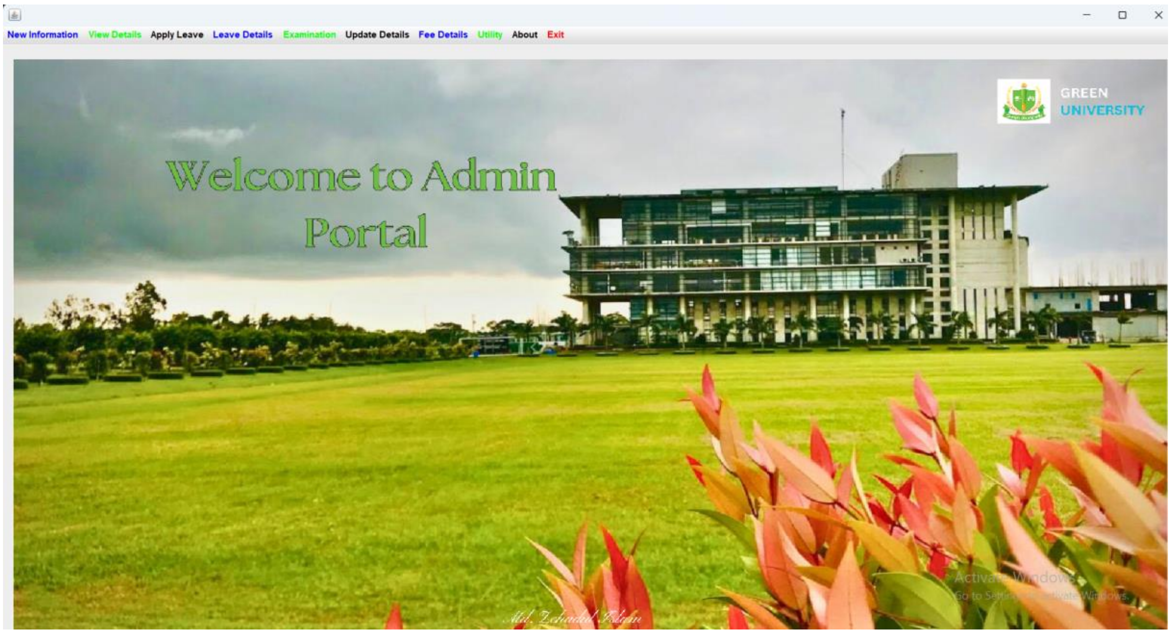
3.3.1 Admin Login



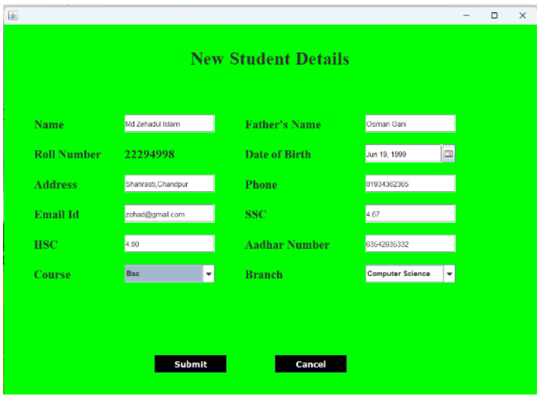
Username admin

Password •••••

3.3.2 Admin Homepage :



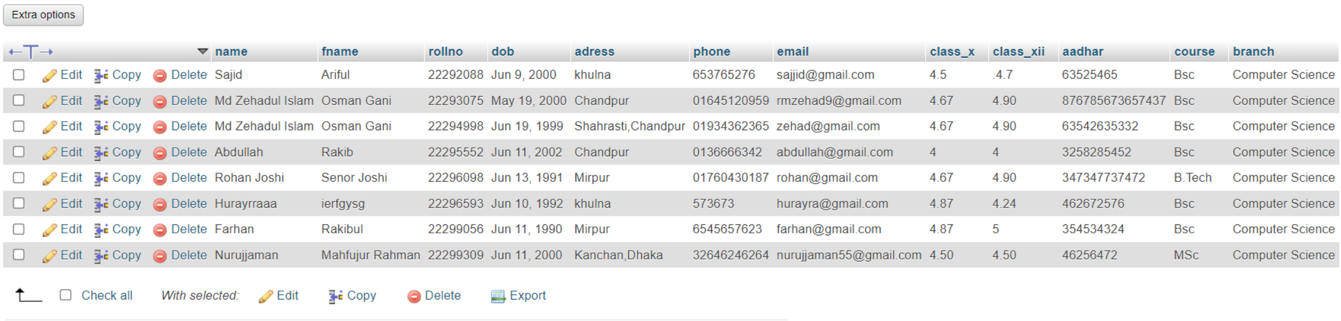
3.3.3 New Student Information



New Student Details

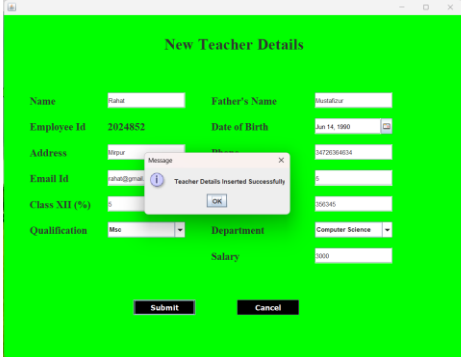
Name	Md Zehadul Islam	Father's Name	Osman Gani
Roll Number	22294998	Date of Birth	Jun 19, 1999
Address	Shahrasti,Chandpur	Phone	01934362365
Email Id	zehad@gmail.com	SSC	4.67
HSC	4.90	Aadhar Number	63542635332
Course	Bsc	Branch	Computer Science

Store in Database :



←T→ ▼	name	fname	rollno	dob	adress	phone	email	class_x	class_xii	aadhar	course	branch
□ Edit Copy Delete	Sajid	Ariful	22292088	Jun 9, 2000	khulna	653765276	sajjid@gmail.com	4.5	.4.7	63525465	Bsc	Computer Science
□ Edit Copy Delete	Md Zehadul Islam	Osman Gani	22293075	May 19, 2000	Chandpur	01645120959	rmzehad9@gmail.com	4.67	4.90	876785673657437	Bsc	Computer Science
□ Edit Copy Delete	Md Zehadul Islam	Osman Gani	22294998	Jun 19, 1999	Shahrasti,Chandpur	01934362365	zehad@gmail.com	4.67	4.90	63542635332	Bsc	Computer Science
□ Edit Copy Delete	Abdullah	Rakib	22295552	Jun 11, 2002	Chandpur	0136666342	abdullah@gmail.com	4	4	3258285452	Bsc	Computer Science
□ Edit Copy Delete	Rohan Joshi	Senor Joshi	22296098	Jun 13, 1991	Mirpur	01760430187	rohan@gmail.com	4.67	4.90	347347737472	B.Tech	Computer Science
□ Edit Copy Delete	Hurayrraaa	ierfgysg	22296593	Jun 10, 1992	khulna	573673	hurayra@gmail.com	4.87	4.24	462672576	Bsc	Computer Science
□ Edit Copy Delete	Farhan	Rakibul	22299056	Jun 11, 1990	Mirpur	6545657623	farhan@gmail.com	4.87	5	354534324	Bsc	Computer Science
□ Edit Copy Delete	Nurujjaman	Mahfujur Rahman	22299309	Jun 11, 2000	Kanchan,Dhaka	32646246264	nurujjaman55@gmail.com	4.50	4.50	46256472	MSc	Computer Science

3.3.4 New Faculty Information



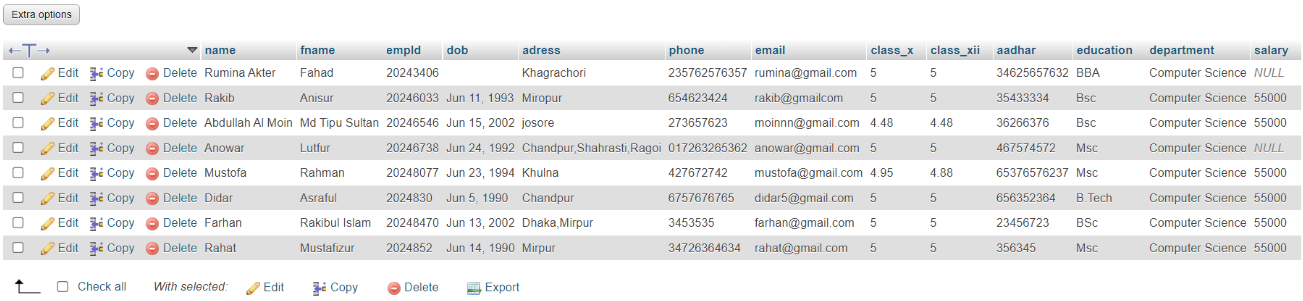
New Teacher Details

Name	Rahat	Father's Name	Mustafizur
Employee Id	2024852	Date of Birth	Jun 14, 1990
Address	Mirpur	Phone	34726364634
Email Id	rahat@gmail.		5
Class XII (%)	5		356345
Qualification	Msc	Department	Computer Science
		Salary	3000

Message

Teacher Details Inserted Successfully

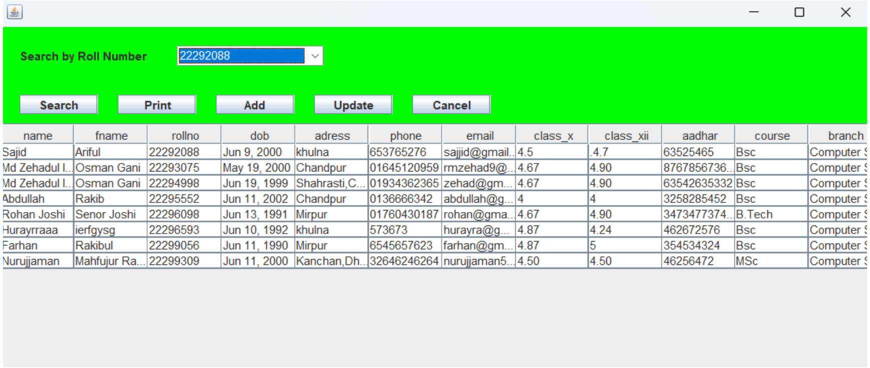
Store in Database,



←T→ ▼	name	fname	empId	dob	adress	phone	email	class_x	class_xii	aadhar	education	department	salary
□ Edit Copy Delete	Rumina Akter	Fahad	20243406		Khagrachori	235762576357	rumina@gmail.com	5	5	34625657632	BBA	Computer Science	NULL
□ Edit Copy Delete	Rakib	Anisur	20246033	Jun 11, 1993	Miropur	654623424	rakib@gmailcom	5	5	35433334	Bsc	Computer Science	55000
□ Edit Copy Delete	Abdullah Al Moin	Md Tipu Sultan	20246546	Jun 15, 2002	josore	273657623	moinnn@gmail.com	4.48	4.48	36266376	Bsc	Computer Science	55000
□ Edit Copy Delete	Anowar	Lutfur	20246738	Jun 24, 1992	Chandpur,Shahrasti,Ragoi	017263265362	anowar@gmail.com	5	5	467574572	Msc	Computer Science	NULL
□ Edit Copy Delete	Mustofa	Rahman	20248077	Jun 23, 1994	Khulna	427672742	mustofa@gmail.com	4.95	4.88	65376576237	Msc	Computer Science	55000
□ Edit Copy Delete	Didar	Asraful	2024830	Jun 5, 1990	Chandpur	6757676765	didar5@gmail.com	5	5	656352364	B.Tech	Computer Science	55000
□ Edit Copy Delete	Farhan	Rakibul Islam	20248470	Jun 13, 2002	Dhaka,Mirpur	3453535	farhan@gmail.com	5	5	23456723	BSc	Computer Science	55000
□ Edit Copy Delete	Rahat	Mustafizur	2024852	Jun 14, 1990	Mirpur	34726364634	rahat@gmail.com	5	5	356345	Msc	Computer Science	55000

3.4 View Details

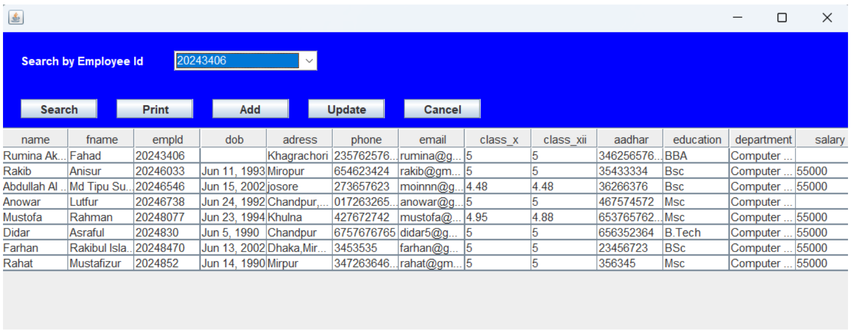
3.4.1 View Student Details



Search by Roll Number: 22292088

name	fname	rollno	dob	adress	phone	email	class_x	class_xii	aadhar	course	branch
Sajid	Ariful	22292088	Jun 9, 2000	khulna	653765276	sajjid@gmail...	4.5	.4.7	63525465	Bsc	Computer S
Md Zehadul I...	Osman Gani	22293075	May 19, 2000	Chandpur	01645120959	rmzehad9@...	4.67	4.90	8767856736...	Bsc	Computer S
Md Zehadul I...	Osman Gani	22294998	Jun 19, 1999	Shahrasti,C...	01934362365	zehad@gm...	4.67	4.90	63542635332	Bsc	Computer S
Abdullah	Rakib	22295552	Jun 11, 2002	Chandpur	0136666342	abdullah@g...	4	4	3258285452	Bsc	Computer S
Rohan Joshi	Senor Joshi	22296098	Jun 13, 1991	Mirpur	01760430187	rohan@gma...	4.67	4.90	3473477374...	B.Tech	Computer S
Hurayrraaa	ierfgysg	22296593	Jun 10, 1992	khulna	573673	hurayra@g...	4.87	4.24	462672576	Bsc	Computer S
Farhan	Rakibul	22299056	Jun 11, 1990	Mirpur	6545657623	farhan@gm...	4.87	5	354534324	Bsc	Computer S
Nurujjaman	Mahfujur Ra...	22299309	Jun 11, 2000	Kanchan,Dh...	32646246264	nurujjaman5...	4.50	4.50	46256472	MSc	Computer S

3.4.2 View Teacher Details

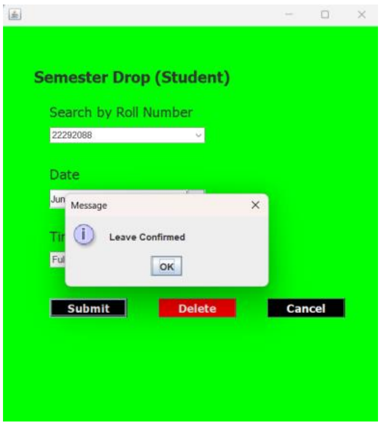


Search by Employee Id: 20243406

name	fname	empId	dob	adress	phone	email	class_x	class_xii	aadhar	education	department	salary
Rumina Ak...	Fahad	20243406		Khagrachori	235762576...	rumina@g...	5	5	346256576...	BBA	Computer ...	
Rakib	Anisur	20246033	Jun 11, 1993	Miropur	654623424	rakib@gm...	5	5	35433334	Bsc	Computer ...	55000
Abdullah Al ...	Md Tipu Su...	20246546	Jun 15, 2002	josore	273657623	moinnn@g...	4.48	4.48	36266376	Bsc	Computer ...	55000
Anowar	Lutfur	20246738	Jun 24, 1992	Chandpur,...	017263265...	anowar@g...	5	5	467574572	Msc	Computer ...	
Mustofa	Rahman	20248077	Jun 23, 1994	Khulna	427672742	mustofa@...	4.95	4.88	653765762...	Msc	Computer ...	55000
Didar	Asraful	2024830	Jun 5, 1990	Chandpur	6757676765	didar5@g...	5	5	656352364	B.Tech	Computer ...	55000
Farhan	Rakibul Isla...	20248470	Jun 13, 2002	Dhaka,Mir...	3453535	farhan@g...	5	5	23456723	BSc	Computer ...	55000
Rahat	Mustafizur	2024852	Jun 14, 1990	Mirpur	347263646...	rahat@gm...	5	5	356345	Msc	Computer ...	55000

3.5 Apply Leave

3.5.1 Student Leave



Semester Drop (Student)

Search by Roll Number: 22292088

Date: Jun

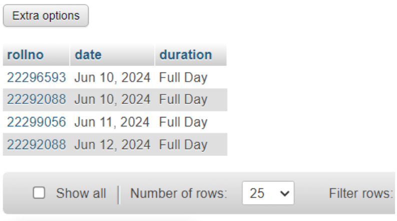
Ti

Ful

Message

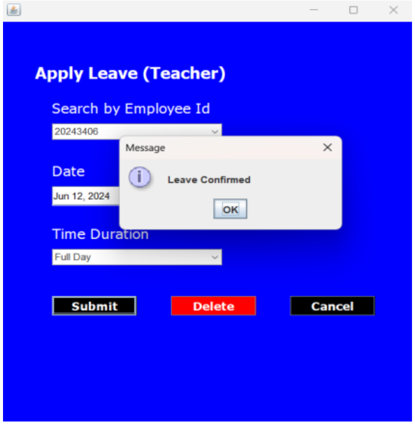
Leave Confirmed

Store Student Leave detail in Database



rollno	date	duration
22296593	Jun 10, 2024	Full Day
22292088	Jun 10, 2024	Full Day
22299056	Jun 11, 2024	Full Day
22292088	Jun 12, 2024	Full Day

3.5.2 Teacher Leave



Apply Leave (Teacher)

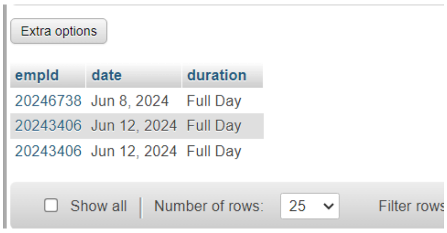
Search by Employee Id: 20243406

Date: Jun 12, 2024

Time Duration: Full Day

Message: Leave Confirmed

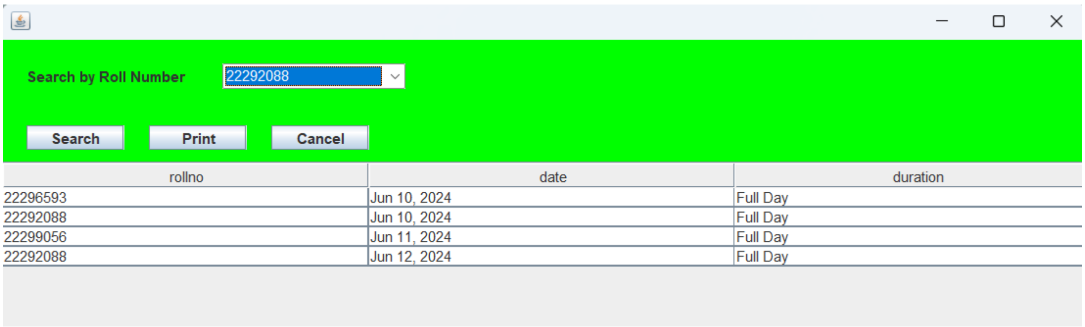
Store in database Teacher Leave Details



empId	date	duration
20246738	Jun 8, 2024	Full Day
20243406	Jun 12, 2024	Full Day
20243406	Jun 12, 2024	Full Day

3.6 Leave Details

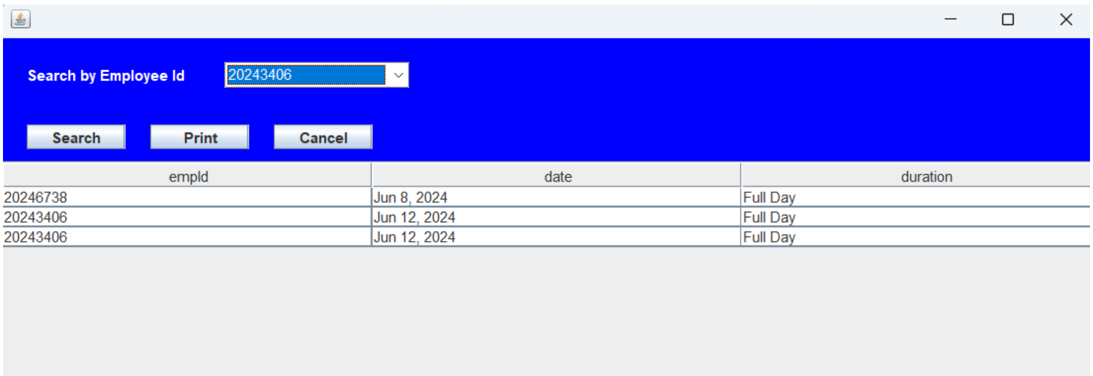
3.6.1 Student Leave Detail use Join Operation



Search by Roll Number: 22292088

rollno	date	duration
22296593	Jun 10, 2024	Full Day
22292088	Jun 10, 2024	Full Day
22299056	Jun 11, 2024	Full Day
22292088	Jun 12, 2024	Full Day

3.6.2 Teacher Leave Detail use Join Operation

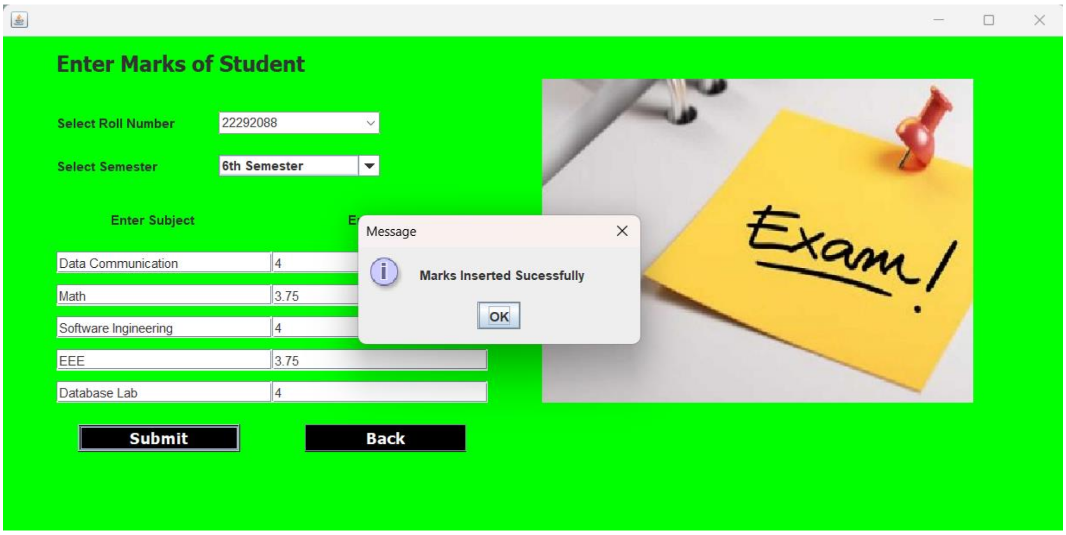


Search by Employee Id: 20243406

empId	date	duration
20246738	Jun 8, 2024	Full Day
20243406	Jun 12, 2024	Full Day
20243406	Jun 12, 2024	Full Day

3.7 Examination

3.7.1 Enter Student Mark



Enter Marks of Student

Select Roll Number: 22292088

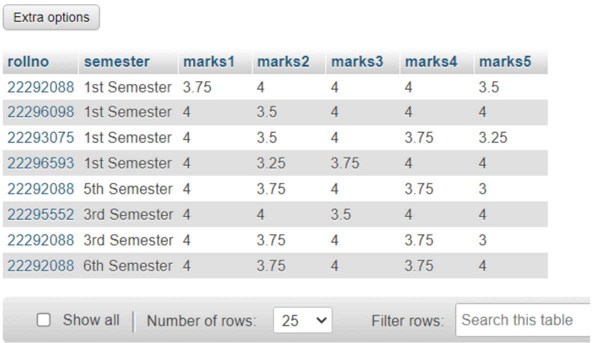
Select Semester: 6th Semester

Enter Subject	E
Data Communication	4
Math	3.75
Software Ingineering	4
EEE	3.75
Database Lab	4

Message

Marks Inserted Sucessfully

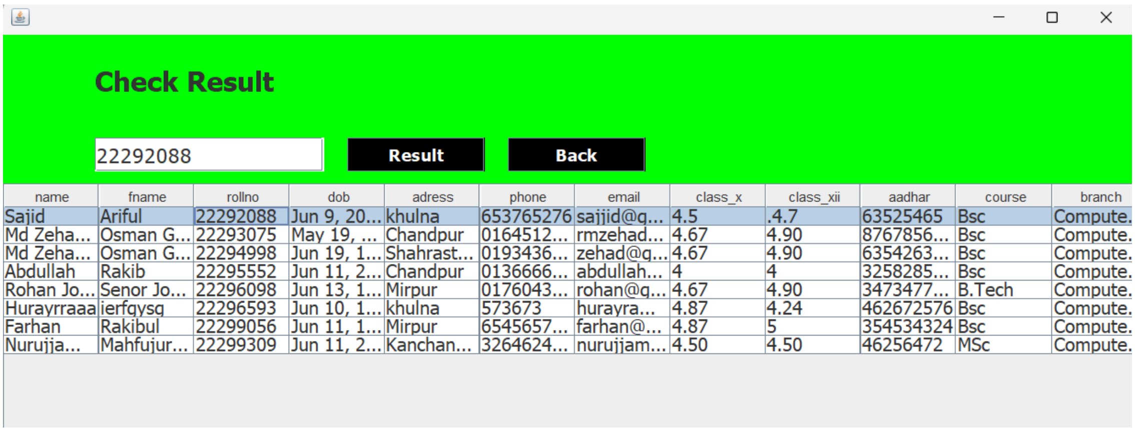
Student mark store in Database



rollno	semester	marks1	marks2	marks3	marks4	marks5
22292088	1st Semester	3.75	4	4	4	3.5
22296098	1st Semester	4	3.5	4	4	4
22293075	1st Semester	4	3.5	4	3.75	3.25
22296593	1st Semester	4	3.25	3.75	4	4
22292088	5th Semester	4	3.75	4	3.75	3
22295552	3rd Semester	4	4	3.5	4	4
22292088	3rd Semester	4	3.75	4	3.75	3
22292088	6th Semester	4	3.75	4	3.75	4

☐ Show all | Number of rows: 25 Filter rows: Search this table

3.7.2 Examination Result

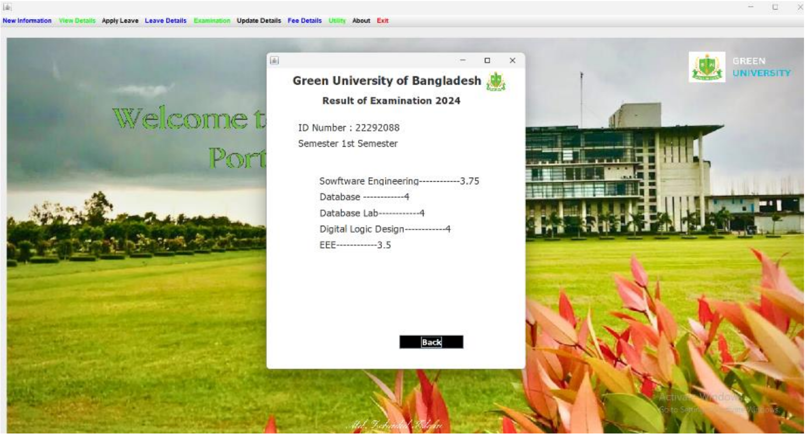


Check Result

22292088

name	fname	rollno	dob	adress	phone	email	class_x	class_xii	aadhar	course	branch
Sajid	Ariful	22292088	Jun 9, 20...	khulna	653765276	sajjid@g...	4.5	.4.7	63525465	Bsc	Compute.
Md Zeha...	Osman G...	22293075	May 19, ...	Chandpur	0164512...	rmzehad...	4.67	4.90	8767856...	Bsc	Compute.
Md Zeha...	Osman G...	22294998	Jun 19, 1...	Shahrast...	0193436...	zehad@g...	4.67	4.90	6354263...	Bsc	Compute.
Abdullah	Rakib	22295552	Jun 11, 2...	Chandpur	0136666...	abdullah...	4	4	3258285...	Bsc	Compute.
Rohan Jo...	Senor Jo...	22296098	Jun 13, 1...	Mirpur	0176043...	rohan@g...	4.67	4.90	3473477...	B.Tech	Compute.
Hurayrraaa	ierfgysg	22296593	Jun 10, 1...	khulna	573673	hurayra...	4.87	4.24	462672576	Bsc	Compute.
Farhan	Rakibul	22299056	Jun 11, 1...	Mirpur	6545657...	farhan@...	4.87	5	354534324	Bsc	Compute.
Nurujja...	Mahfujur...	22299309	Jun 11, 2...	Kanchan...	3264624...	nurujjam...	4.50	4.50	46256472	MSc	Compute.

After select the Result follow for Student roll number then show the Subject wise Student Result use Join Operation



Green University of Bangladesh

Result of Examination 2024

ID Number : 22292088

Semester 1st Semester

Sowftware Engineering------------3.75

Database ------------4

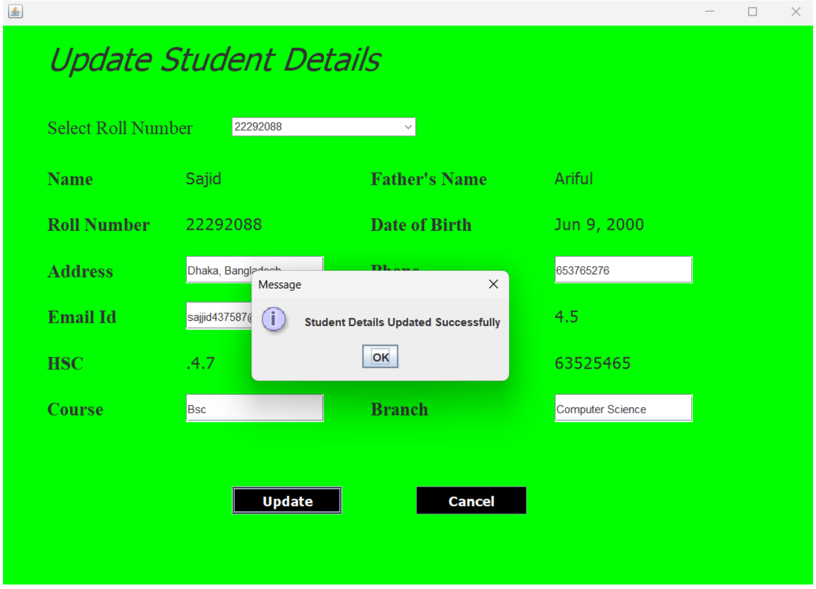
Database Lab------------4

Digital Logic Design------------4

EEE------------3.5

3.8 Update Details

3.8.1 Update Student Details



Update Student Details

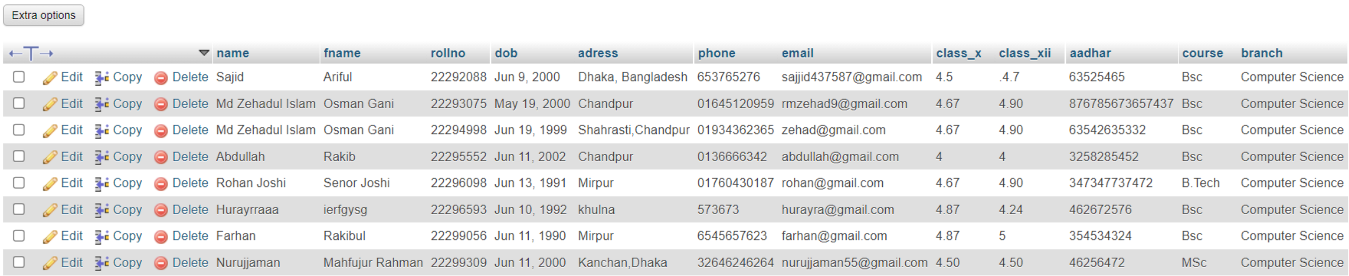
Select Roll Number 22292088

Name	Sajid	Father's Name	Ariful
Roll Number	22292088	Date of Birth	Jun 9, 2000
Address	Dhaka, Bangladesh	Phone	653765276
Email Id	sajjid437587(		4.5
HSC	.4.7		63525465
Course	Bsc	Branch	Computer Science

Message

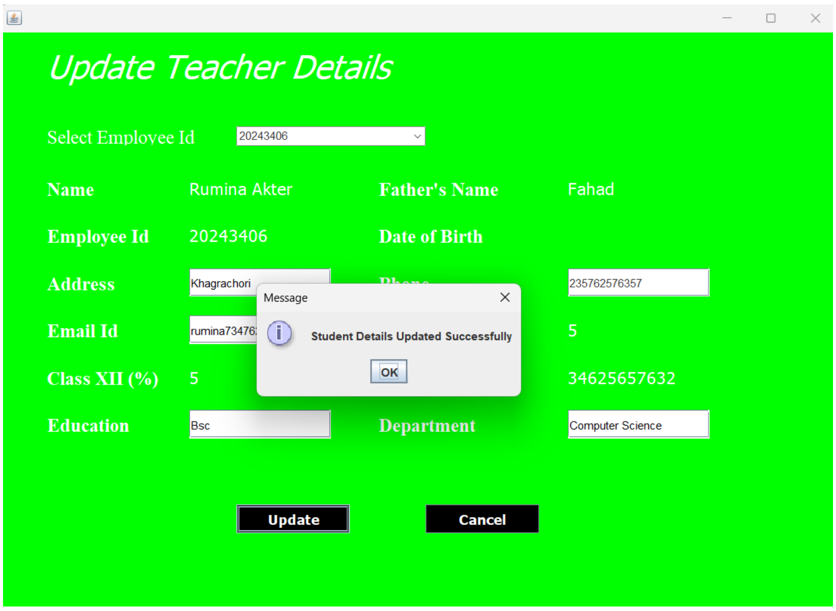
Student Details Updated Successfully

After update in Student details then store ine database



				name	fname	rollno	dob	adress	phone	email	class_x	class_xii	aadhar	course	branch
☐	Edit	Copy	Delete	Sajid	Ariful	22292088	Jun 9, 2000	Dhaka, Bangladesh	653765276	sajjid437587@gmail.com	4.5	.4.7	63525465	Bsc	Computer Science
☐	Edit	Copy	Delete	Md Zehadul Islam	Osman Gani	22293075	May 19, 2000	Chandpur	01645120959	rmzehad9@gmail.com	4.67	4.90	876785673657437	Bsc	Computer Science
☐	Edit	Copy	Delete	Md Zehadul Islam	Osman Gani	22294998	Jun 19, 1999	Shahrasti,Chandpur	01934362365	zehad@gmail.com	4.67	4.90	63542635332	Bsc	Computer Science
☐	Edit	Copy	Delete	Abdullah	Rakib	22295552	Jun 11, 2002	Chandpur	0136666342	abdullah@gmail.com	4	4	3258285452	Bsc	Computer Science
☐	Edit	Copy	Delete	Rohan Joshi	Senor Joshi	22296098	Jun 13, 1991	Mirpur	01760430187	rohan@gmail.com	4.67	4.90	347347737472	B.Tech	Computer Science
☐	Edit	Copy	Delete	Hurayrraaa	ierfgysg	22296593	Jun 10, 1992	khulna	573673	hurayra@gmail.com	4.87	4.24	462672576	Bsc	Computer Science
☐	Edit	Copy	Delete	Farhan	Rakibul	22299056	Jun 11, 1990	Mirpur	6545657623	farhan@gmail.com	4.87	5	354534324	Bsc	Computer Science
☐	Edit	Copy	Delete	Nurujjaman	Mahfujur Rahman	22299309	Jun 11, 2000	Kanchan,Dhaka	32646246264	nurujjaman55@gmail.com	4.50	4.50	46256472	MSc	Computer Science

3.8.2 Update Teacher Details



Update Teacher Details

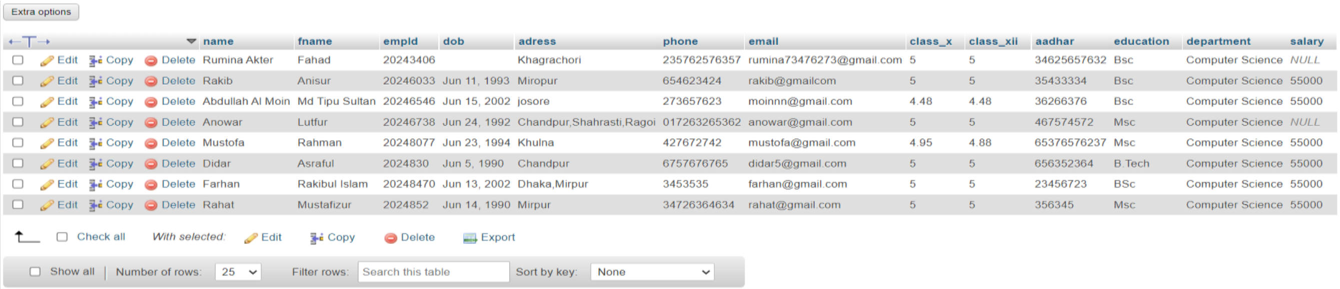
Select Employee Id: 20243406

Name	Rumina Akter	Father's Name	Fahad
Employee Id	20243406	Date of Birth	
Address	Khagrachori	Phone	235762576357
Email Id	rumina73476		5
Class XII (%)	5		34625657632
Education	Bsc	Department	Computer Science

Message

Student Details Updated Successfully

After update in Teacher details then store ine database

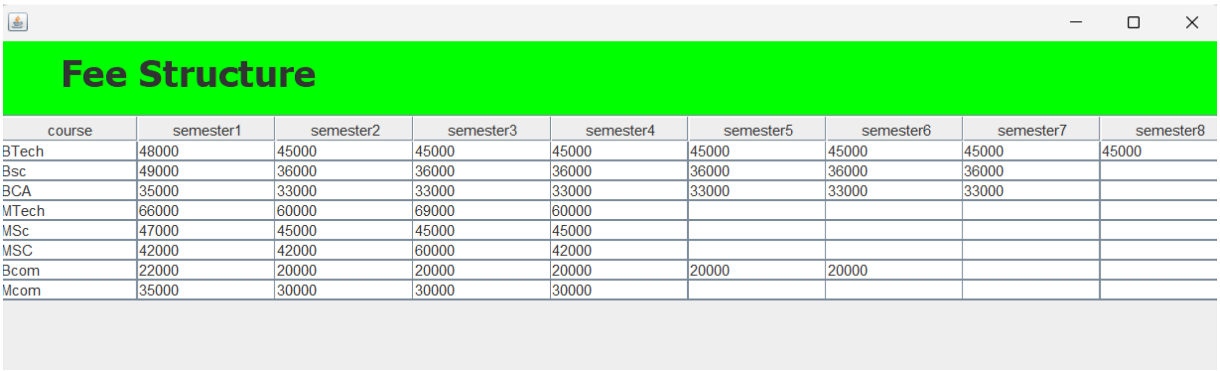


	name	fname	empId	dob	adress	phone	email	class_x	class_xii	aadhar	education	department	salary
□ Edit Copy Delete	Rumina Akter	Fahad	20243406		Khagrachori	235762576357	rumina73476273@gmail.com	5	5	34625657632	Bsc	Computer Science	NULL
□ Edit Copy Delete	Rakib	Anisur	20246033	Jun 11, 1993	Miropur	654623424	rakib@gmailcom	5	5	35433334	Bsc	Computer Science	55000
□ Edit Copy Delete	Abdullah Al Moin	Md Tipu Sultan	20246546	Jun 15, 2002	josore	273657623	moinnn@gmail.com	4.48	4.48	36266376	Bsc	Computer Science	55000
□ Edit Copy Delete	Anowar	Lutfur	20246738	Jun 24, 1992	Chandpur,Shahrasti,Ragoi	017263265362	anowar@gmail.com	5	5	467574572	Msc	Computer Science	NULL
□ Edit Copy Delete	Mustofa	Rahman	20248077	Jun 23, 1994	Khulna	427672742	mustofa@gmail.com	4.95	4.88	65376576237	Msc	Computer Science	55000
□ Edit Copy Delete	Didar	Asraful	2024830	Jun 5, 1990	Chandpur	6757676765	didar5@gmail.com	5	5	656352364	B.Tech	Computer Science	55000
□ Edit Copy Delete	Farhan	Rakibul Islam	20248470	Jun 13, 2002	Dhaka,Mirpur	3453535	farhan@gmail.com	5	5	23456723	BSc	Computer Science	55000
□ Edit Copy Delete	Rahat	Mustafizur	2024852	Jun 14, 1990	Mirpur	34726364634	rahat@gmail.com	5	5	356345	Msc	Computer Science	55000

□ Check all
With selected: Edit Copy Delete Export
□ Show all
Number of rows: 25
Filter rows: Search this table
Sort by key: None

3.9 Fee Details

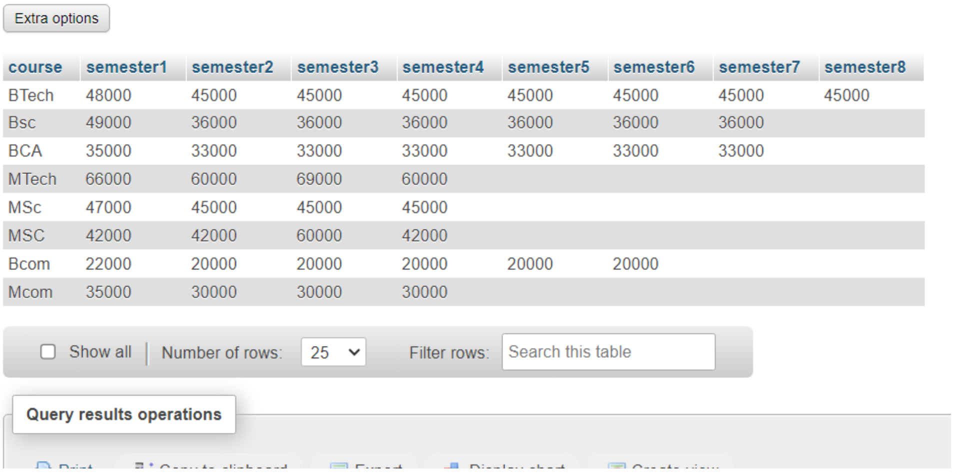
3.9.1 Fee Structure show



Fee Structure

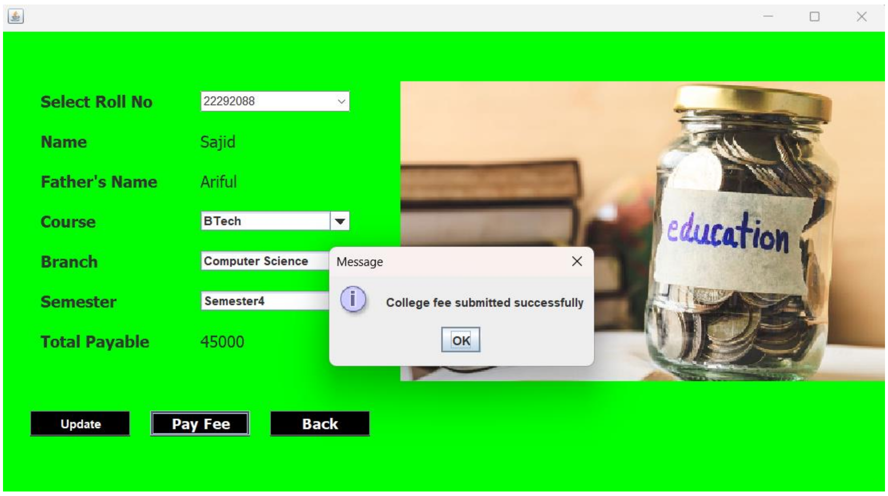
course	semester1	semester2	semester3	semester4	semester5	semester6	semester7	semester8
BTech	48000	45000	45000	45000	45000	45000	45000	45000
Bsc	49000	36000	36000	36000	36000	36000	36000	
BCA	35000	33000	33000	33000	33000	33000	33000	
MTech	66000	60000	69000	60000				
MSc	47000	45000	45000	45000				
MSC	42000	42000	60000	42000				
Bcom	22000	20000	20000	20000	20000	20000		
Mcom	35000	30000	30000	30000				

Fee Structure in Database:



course	semester1	semester2	semester3	semester4	semester5	semester6	semester7	semester8
BTech	48000	45000	45000	45000	45000	45000	45000	45000
Bsc	49000	36000	36000	36000	36000	36000	36000	
BCA	35000	33000	33000	33000	33000	33000	33000	
MTech	66000	60000	69000	60000				
MSc	47000	45000	45000	45000				
MSC	42000	42000	60000	42000				
Bcom	22000	20000	20000	20000	20000	20000		
Mcom	35000	30000	30000	30000				

3.9.2 Student fee form



Select Roll No 22292088

Name Sajid

Father's Name Ariful

Course BTech

Branch Computer Science

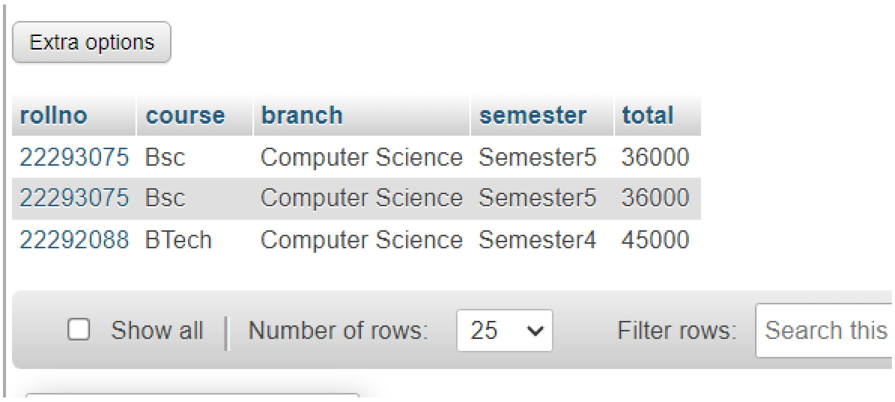
Semester Semester4

Total Payable 45000

Message

College fee submitted successfully

Show details after selecting student roll number and then select course, branch and semester then automatically generate semester fee value which collects the data of fee structure table in databse.

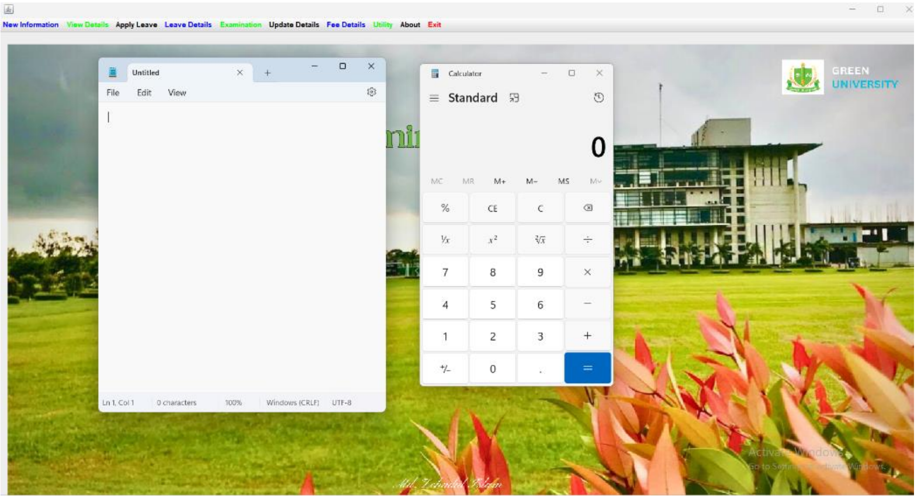


rollno	course	branch	semester	total
22293075	Bsc	Computer Science	Semester5	36000
22293075	Bsc	Computer Science	Semester5	36000
22292088	BTech	Computer Science	Semester4	45000

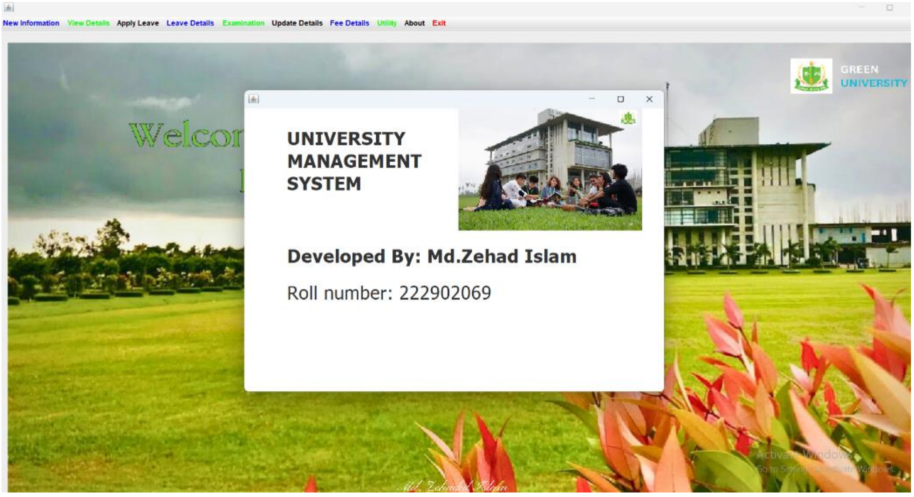
3.10 Utility

3.10.1 Notepad and Calculator

If the admin wants to, the notepad and calculator can be started from the portal and work.



3.11 About



UNIVERSITY MANAGEMENT SYSTEM

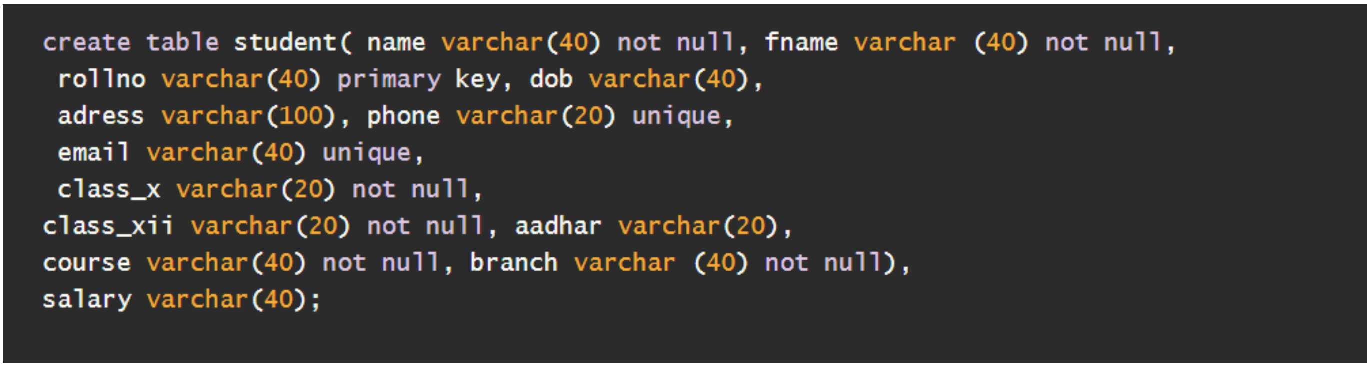
Developed By: Md.Zehad Islam

Roll number: 222902069

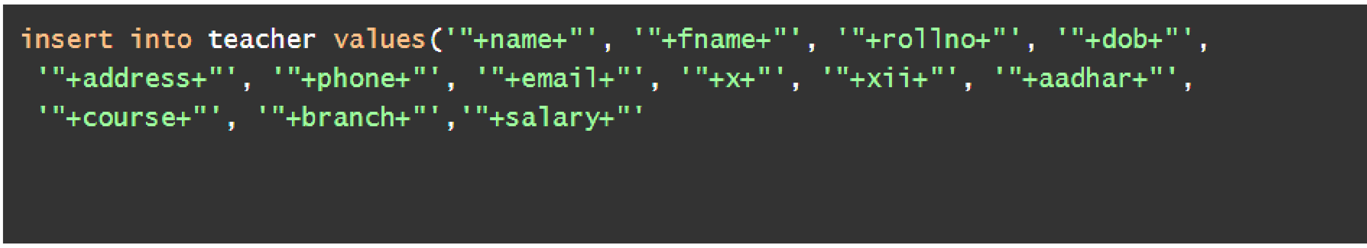
3.12 Database Queries

3.12.1 All Query That We Used in Our Project

1. **New student Information:**

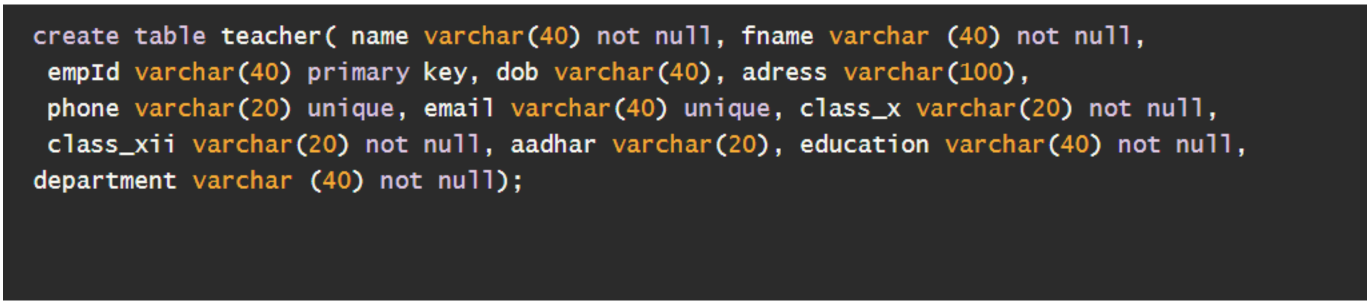


```
create table student( name varchar(40) not null, fname varchar (40) not null,
 rollno varchar(40) primary key, dob varchar(40),
 adress varchar(100), phone varchar(20) unique,
 email varchar(40) unique,
 class_x varchar(20) not null,
class_xii varchar(20) not null, aadhar varchar(20),
course varchar(40) not null, branch varchar (40) not null),
salary varchar(40);
```



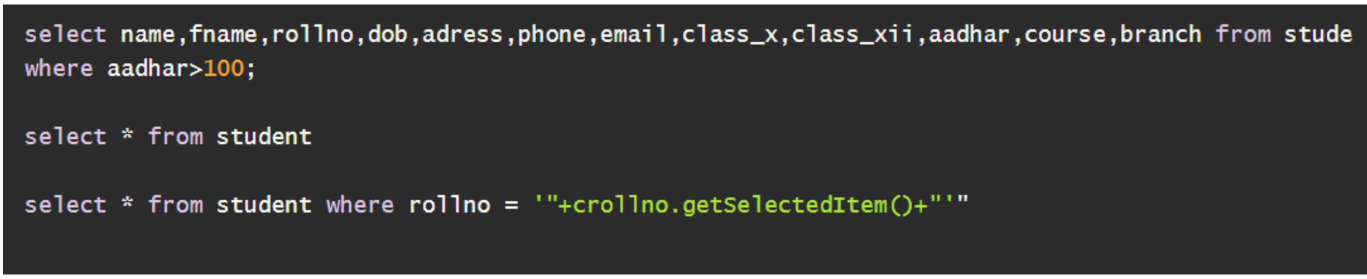
```
insert into teacher values('"+name+"', '"+fname+"', '"+rollno+"', '"+dob+"',
 '"+address+"', '"+phone+"', '"+email+"', '"+x+"', '"+xii+"', '"+aadhar+"',
 '"+course+"', '"+branch+"','"+salary+"'
```

2. **New Faculty Information:**



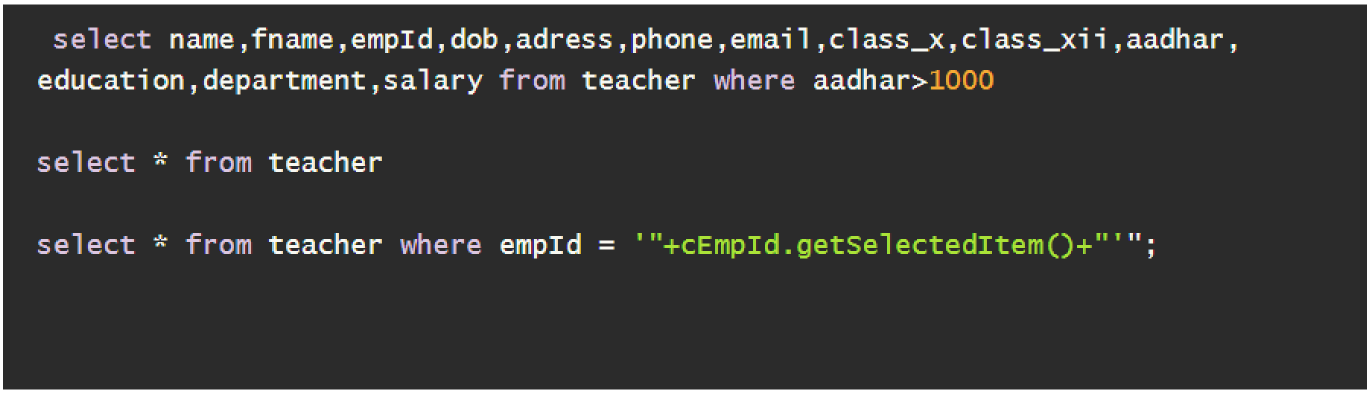
```
create table teacher( name varchar(40) not null, fname varchar (40) not null,
 empId varchar(40) primary key, dob varchar(40), adress varchar(100),
 phone varchar(20) unique, email varchar(40) unique, class_x varchar(20) not null,
 class_xii varchar(20) not null, aadhar varchar(20), education varchar(40) not null,
department varchar (40) not null);
```

3. **Student Details**



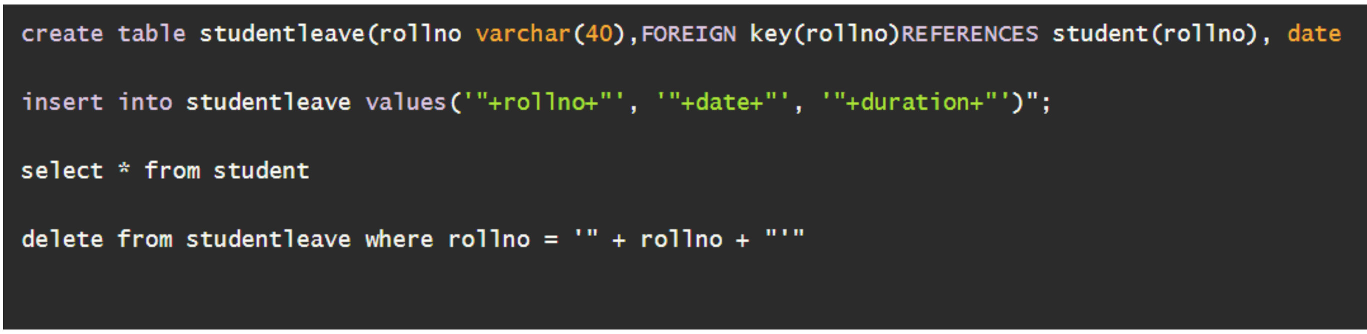
```
select name,fname,rollno,dob,adress,phone,email,class_x,class_xii,aadhar,course,branch from stude
where aadhar>100;

select * from student

select * from student where rollno = '"+crollno.getSelectedItem()+"'"
```

4. **Teacher Details**



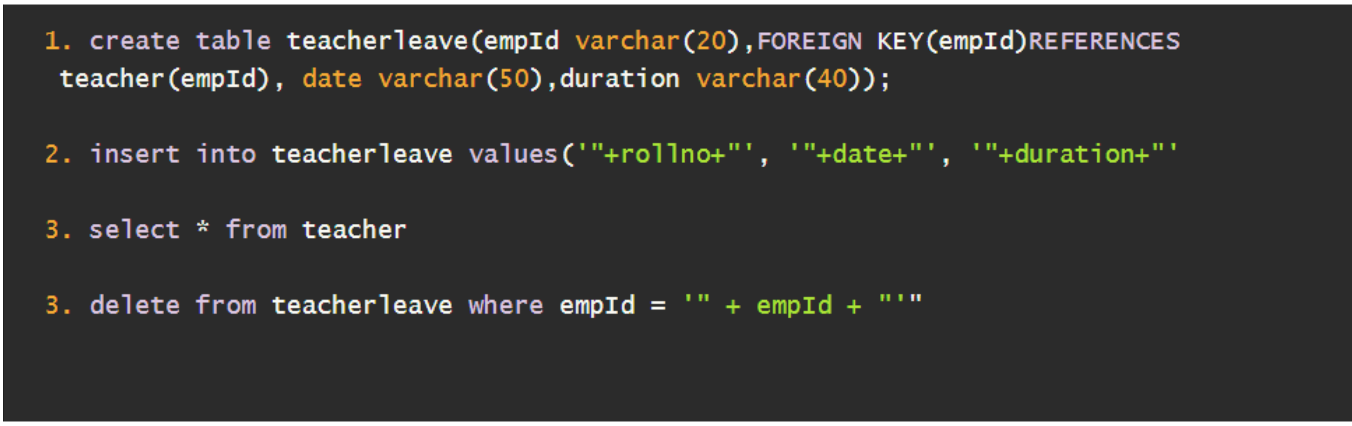
```
 select name,fname,empId,dob,adress,phone,email,class_x,class_xii,aadhar,
education,department,salary from teacher where aadhar>1000

select * from teacher

select * from teacher where empId = '"+cEmpId.getSelectedItem()+"'";
```

5. **Student Leave**



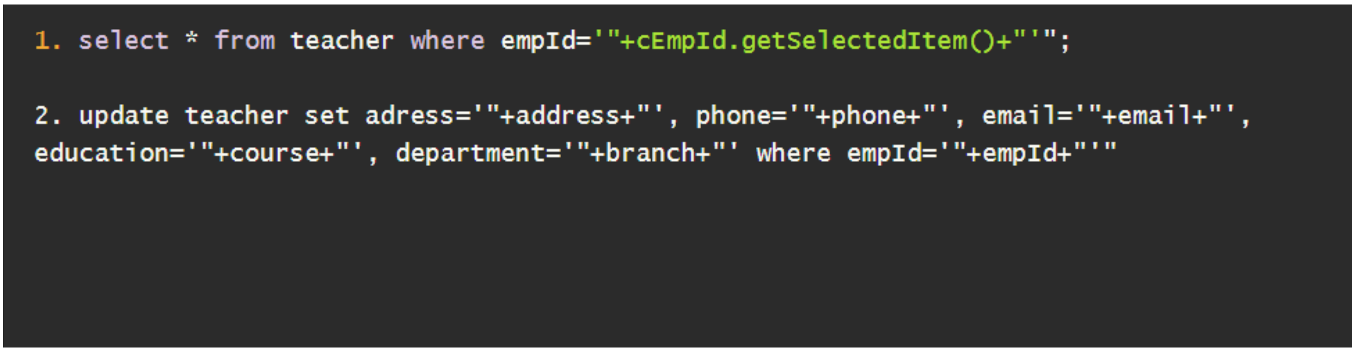
```
create table studentleave(rollno varchar(40),FOREIGN key(rollno)REFERENCES student(rollno), date

insert into studentleave values('"+rollno+"', '"+date+"', '"+duration+"')";

select * from student

delete from studentleave where rollno = '" + rollno + "'"
```

6. **Teacher Leave**



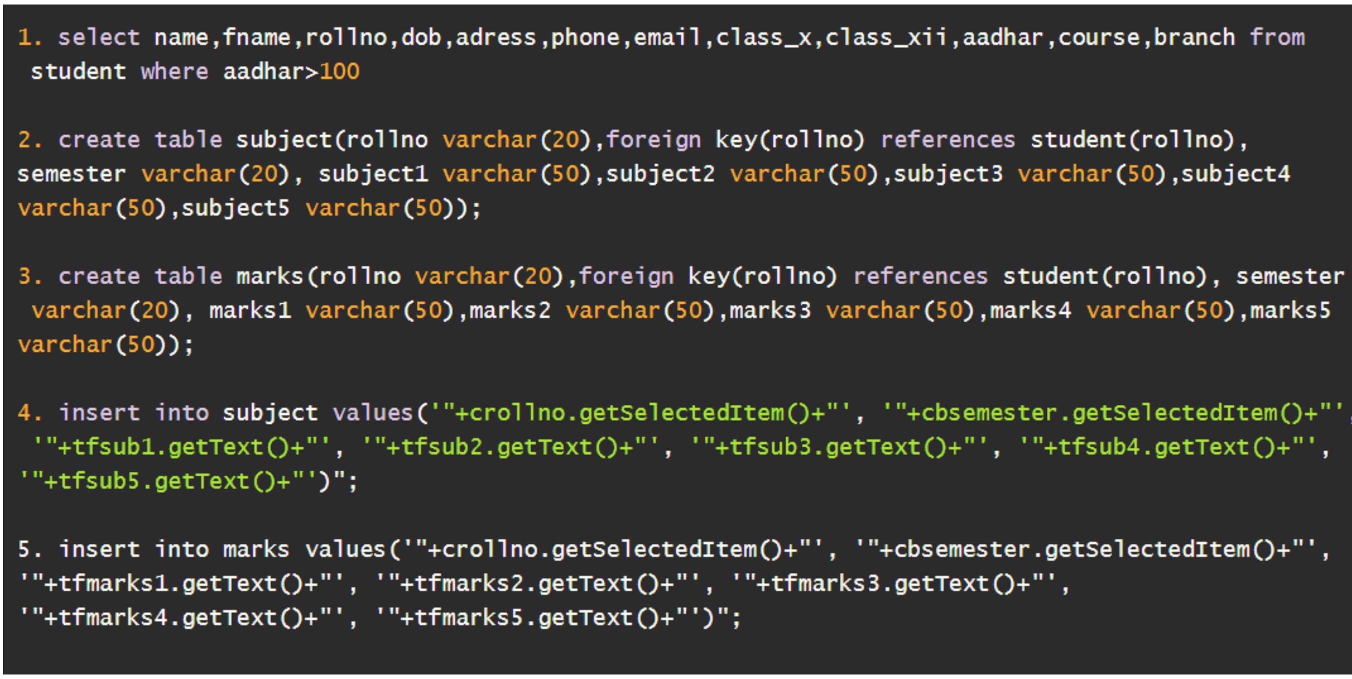
```
1. create table teacherleave(empId varchar(20),FOREIGN KEY(empId)REFERENCES
 teacher(empId), date varchar(50),duration varchar(40));

2. insert into teacherleave values('"+rollno+"', '"+date+"', '"+duration+"'

3. select * from teacher

3. delete from teacherleave where empId = '" + empId + "'"
```

7. **Update Teacher Details**



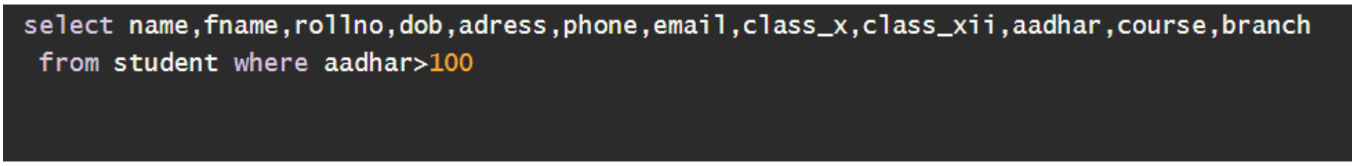
```
1. select * from teacher where empId='"+cEmpId.getSelectedItem()+"'";

2. update teacher set adress='"+address+"', phone='"+phone+"', email='"+email+"',
education='"+course+"', department='"+branch+"' where empId='"+empId+"'"
```

8. **Enter Marks**



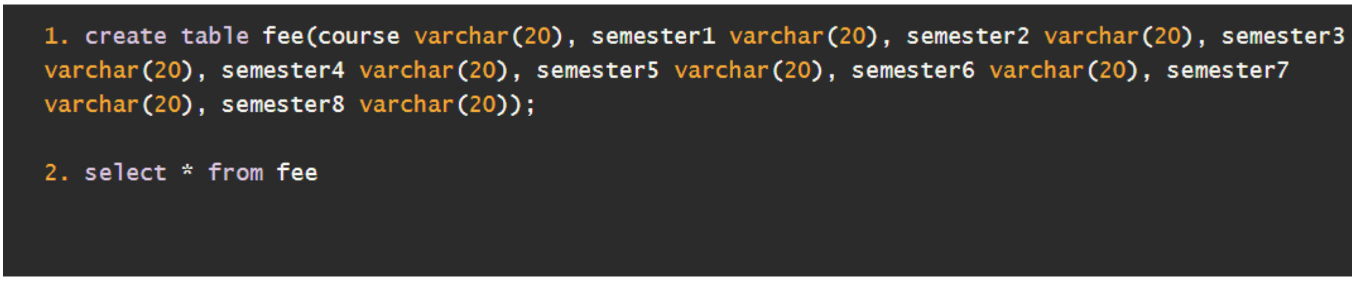
```
1. select name,fname,rollno,dob,adress,phone,email,class_x,class_xii,aadhar,course,branch from
 student where aadhar>100

2. create table subject(rollno varchar(20),foreign key(rollno) references student(rollno),
semester varchar(20), subject1 varchar(50),subject2 varchar(50),subject3 varchar(50),subject4
varchar(50),subject5 varchar(50));

3. create table marks(rollno varchar(20),foreign key(rollno) references student(rollno), semester
 varchar(20), marks1 varchar(50),marks2 varchar(50),marks3 varchar(50),marks4 varchar(50),marks5
varchar(50));

4. insert into subject values('"+crollno.getSelectedItem()+"', '"+cbsemester.getSelectedItem()+"'
 '"+tfsub1.getText()+"', '"+tfsub2.getText()+"', '"+tfsub3.getText()+"', '"+tfsub4.getText()+"',
'"+tfsub5.getText()+"')";

5. insert into marks values('"+crollno.getSelectedItem()+"', '"+cbsemester.getSelectedItem()+"',
'"+tfmarks1.getText()+"', '"+tfmarks2.getText()+"', '"+tfmarks3.getText()+"',
'"+tfmarks4.getText()+"', '"+tfmarks5.getText()+"')";
```

9. **Examination Details**



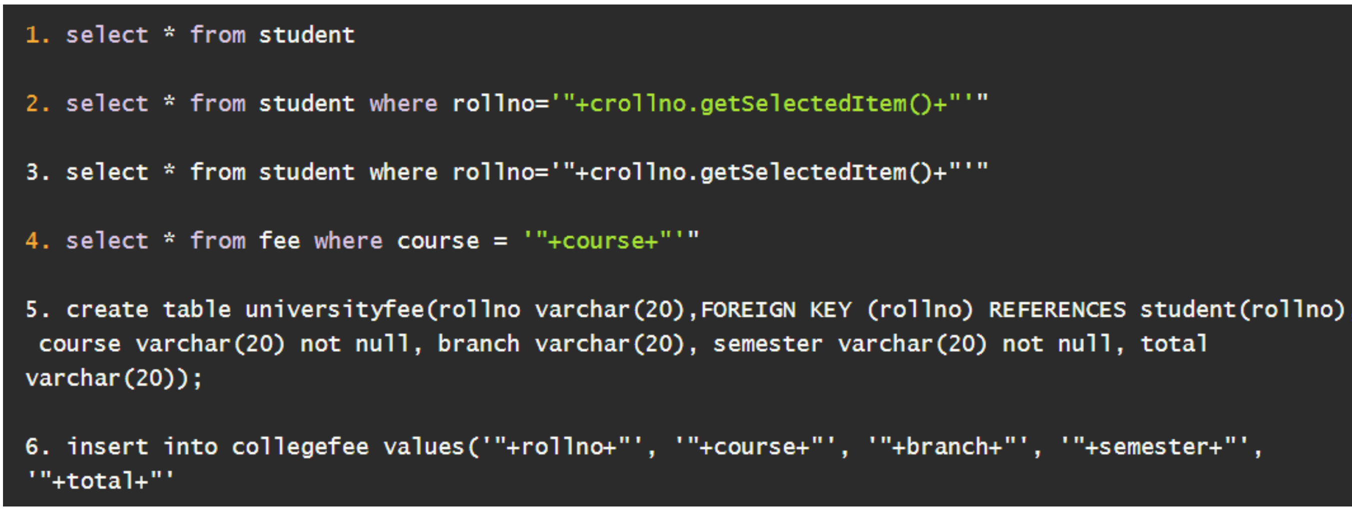
```
select name,fname,rollno,dob,adress,phone,email,class_x,class_xii,aadhar,course,branch
 from student where aadhar>100
```

10. **Fee Structure**



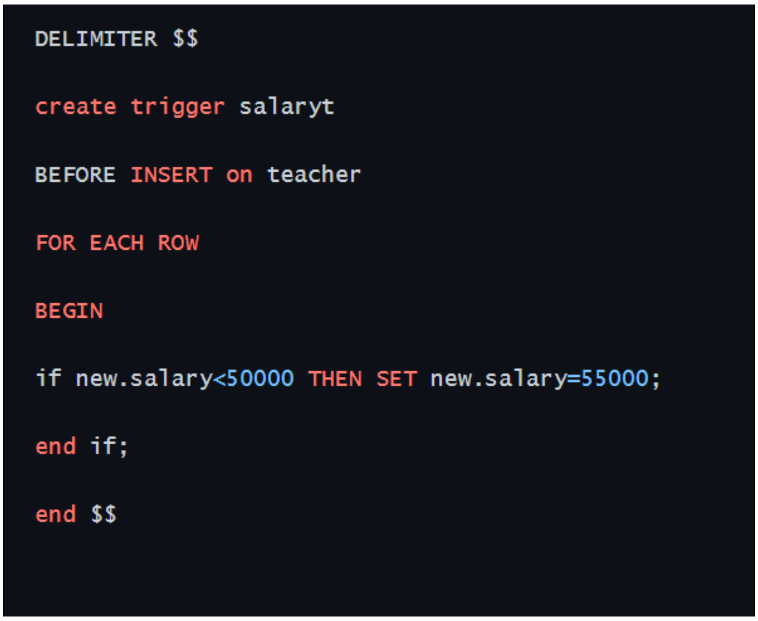
```
1. create table fee(course varchar(20), semester1 varchar(20), semester2 varchar(20), semester3
varchar(20), semester4 varchar(20), semester5 varchar(20), semester6 varchar(20), semester7
varchar(20), semester8 varchar(20));

2. select * from fee
```

11.**Student fee form**



```
1. select * from student

2. select * from student where rollno='"+crollno.getSelectedItem()+"'"

3. select * from student where rollno='"+crollno.getSelectedItem()+"'"

4. select * from fee where course = '"+course+"'"

5. create table universityfee(rollno varchar(20),FOREIGN KEY (rollno) REFERENCES student(rollno)
 course varchar(20) not null, branch varchar(20), semester varchar(20) not null, total
varchar(20));

6. insert into collegefee values('"+rollno+"', '"+course+"', '"+branch+"', '"+semester+"',
'"+total+"'
```

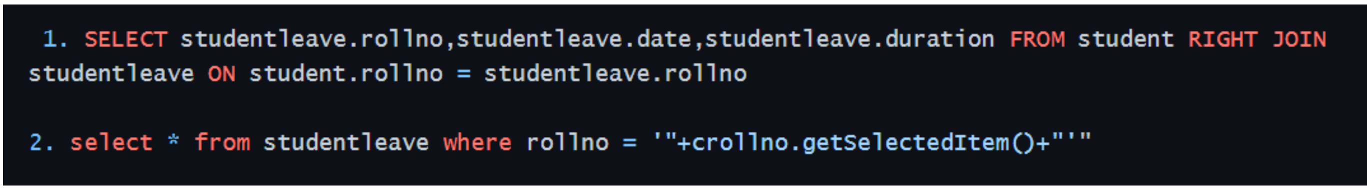
Trigger Operation :



```
DELIMITER $$

create trigger salaryt

BEFORE INSERT on teacher

FOR EACH ROW

BEGIN

if new.salary<50000 THEN SET new.salary=55000;

end if;

end $$
```

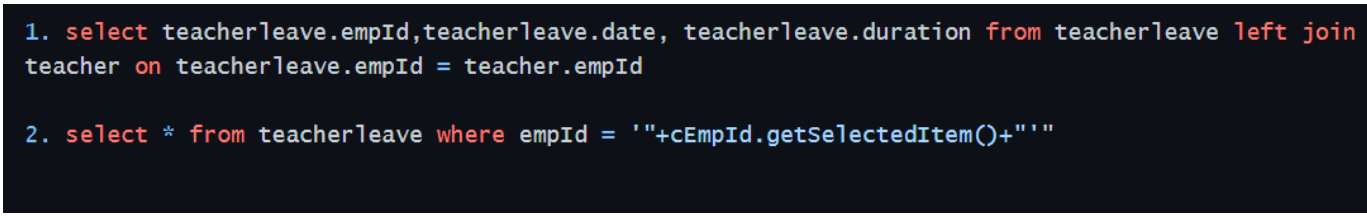
Join Operation That We Applied in My Java Code :

12.**Student Leave Details show Right Join**



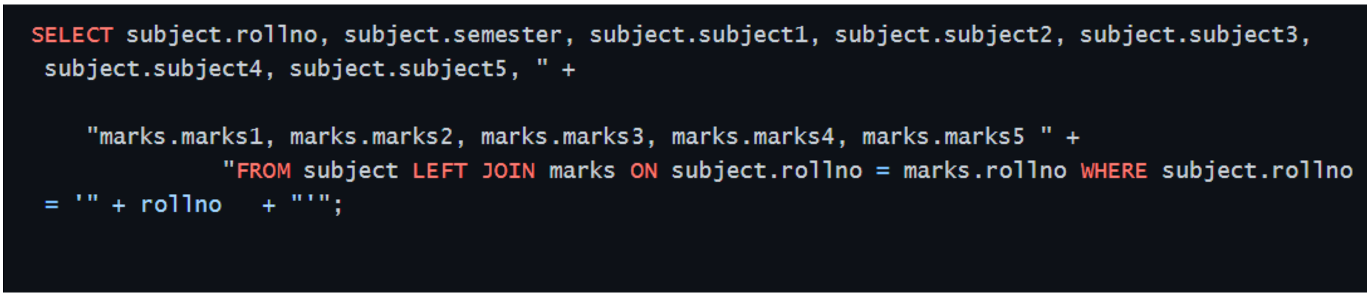
```
 1. SELECT studentleave.rollno,studentleave.date,studentleave.duration FROM student RIGHT JOIN
studentleave ON student.rollno = studentleave.rollno

2. select * from studentleave where rollno = '"+crollno.getSelectedItem()+"'"
```

13.**Teacher Leave Details show Left Join**



```
select teacherleave.empId,teacherleave.date, teacherleave.duration from teacherleave left join
teacher on teacherleave.empId = teacher.empId

select * from teacherleave where empId = '"+cEmpId.getSelectedItem()+"'"
```

14. **Student Mark show use Left Join**



```
SELECT subject.rollno, subject.semester, subject.subject1, subject.subject2, subject.subject3,
 subject.subject4, subject.subject5, " +

     "marks.marks1, marks.marks2, marks.marks3, marks.marks4, marks.marks5 " +
                "FROM subject LEFT JOIN marks ON subject.rollno = marks.rollno WHERE subject.rollno
 = '" + rollno   + "'";
```

3.13 Results Overall Discussion

The university management system project uses a database to manage all operations. Admins can log in to control various functions like adding students and teachers, viewing their details, managing leave requests, and handling examinations by entering and displaying student marks. Admins can also update student and teacher information, manage fee details including course fees and student fee forms, and access utilities like a notepad and calculator. Additionally, the system provides an "About" section for general information. If incorrect data is entered, the system will show errors. This project centralizes all admin tasks, ensuring efficient and error-free management using a database.

3.13.1 Complex Engineering Problem Discussion

The university management system project addresses complex engineering challenges by centralizing administrative tasks using a robust database. Key issues include ensuring data integrity, managing concurrent user access, and maintaining security for sensitive information. The system must handle various functionalities like adding and updating student and teacher records, managing leave and fee details, and processing examination results, all while providing real-time error detection for incorrect data entries. Efficient database design and user interface development are critical to ensure seamless and secure operations, this a comprehensive and challenging engineering problem.

Chapter 4

Conclusion

4.1 Discussion

The "University Management System" project aims to offer an efficient and user-friendly platform for managing university affairs. It includes features like student and teacher registration, course management, attendance tracking, exam scheduling, and result processing. The system, developed using Java technologies, prioritizes security and follows industry standards. The project report outlines its goals, design, and functions, demonstrating its capacity to streamline university processes and improve administrative efficiency.

4.2 Limitations

The University Management System in Java using MySQL, despite its comprehensive features, faces several limitations. It relies heavily on manual data entry, increasing the risk of human error. Scalability is another concern; as the number of students and faculty grows, performance might degrade without optimization. Additionally, security measures, though present, need constant updating to counter evolving threats. The system's dependency on MySQL may also limit flexibility with other databases. Finally, the initial setup and maintenance require significant technical expertise, which might not be readily available in all institutions.

4.3 Scope of Future Work

For future work and extension of the "University management System" project, several areas can be considered to enhance its functionality use Admin. Here are some potential plans for further development:

4.3.1 Enhanced Automation:

- Implement advanced automation to reduce manual data entry errors.

4.3.2 Scalability:

- Optimize the system to handle larger databases efficiently as the university grows..

4.3.3 Security Improvements:

- Continuously update security protocols to protect against new threats.

4.3.4 Database Flexibility:

- Incorporate compatibility with various database management systems beyond MySQL.

4.3.5 User Interface:

- Improve the user interface for better accessibility and user experience.

4.3.6 Mobile Application Developments :

- Develop mobile applications for easier access by faculty admin.

4.3.7 Advanced Analytics:

- Integrate analytics for data-driven decision-making and performance tracking.

References

[1] Learning MySQL" by Seyed M.M. Tahaghoghi and Hugh E. Williams (Publisher: O'Reilly Media)

[2] Learning MySQL. https://www.javatpoint.com/mysql-tutorial.

[3] "MySQL 8 Cookbook" by Karthik Appigatla and Chintan Mehta (Publisher: Packt)

[4] Using DataBase Software : Xampp - https://www.apachefriends.org/index.html