| **Faculty of Engineering & Technology** | | | |
| --- | --- | --- | --- |
| **Ramaiah University of Applied Sciences** | | | |
| **Department** | Computer Science and Engineering | **Programme** | B. Tech. CSE/AIML/ISE |
| **Semester/Batch** | 5th/2021 | | |
| **Course Code** | 20CSC302A | **Course Title** | Database Systems |
| **Course Leader(s)** | Dr. Narendra Babu, Mrs. Sahana P Shankar, Mrs. Supriya M S | | |

| **Assignment** | | | | | | | | |
| --- | --- | --- | --- | --- | --- | --- | --- | --- |
| Register No. | | | Bhoomika K | Name of Student | | 21ETAI410033 | | |
| **Sections** |  | **Marking Scheme** | | | **Max Marks** | | **First Examiner Marks** | **Second Examiner Marks** |
| **Part A** | **A.1** | Discuss any two database models used in the modern day enterprise computing applications with suitable examples. | | | 05 | |  |  |
|  | **Part-A Max Marks** | | | **05** | |  |  |
| **Part B** | **B2.1** | List of functional requirements | | | 02 | |  |  |
| **B2.2** | Implementation of relational database schema with  appropriate attributes, and constraints using SQL commands | | | 10 | |  |  |
| **B2.3** | Design and implementation of GUI | | | 05 | |  |  |
| **B2.4** | Connection of front end with the database and discussion  on the results | | | 03 | |  |  |
|  | **Part-B Max Marks** | | | **20** | |  |  |
|  | **Total Assignment Marks** | | | | **25** | |  |  |

| **Course Marks Tabulation** | | | | |
| --- | --- | --- | --- | --- |
| **Component-**  **1(B)Assignment** | **First**  **Examiner** | **Remarks** | **Second**  **Examiner** | **Remarks** |
| **A** |  |  |  |  |
| **B** |  |  |  |  |
| **Marks (out of 25)** |  |  |  |  |
| Signature of First Examiner Signature of Second Examiner | | | | |

# Please note:

1. Documental evidence for all the components/parts of the assessment such as the reports, photographs, laboratory exam / tool tests are required to be attached to the assignment report in a proper order.
2. The First Examiner is required to mark the comments in RED ink and the Second Examiner’s comments should be in GREEN ink.
3. The marks for all the questions of the assignment have to be written only in the **Component – CET B: Assignment** table.
4. If the variation between the marks awarded by the first examiner and the second examiner lies within +/- 3 marks, then the marks allotted by the first examiner is considered to be final. If the variation is more than +/- 3 marks then both the examiners should resolve the issue in consultation with the Chairman BoE.

# Assignment

**Instructions to students:**

* 1. The assignment consists of **2** questions: Part A –**1** Question, Part B- **1** Questions.
  2. Maximum marks is **25**.
  3. The assignment has to be neatly word processed as per the prescribed format.
  4. **Submission Date:** 12/03/2024

# Submission after the due date is not permitted.

* 1. **IMPORTANT**: It is essential that all the sources used in preparation of the assignment must be suitably referenced in the text.
  2. Marks will be awarded only to the sections and subsections clearly indicated as per the problem statement/exercise/question

# PART A 05 Marks

Enterprise computing applications are software applications designed to meet the complex and extensive requirements of large organizations or enterprises. These applications are critical for supporting various business processes, enhancing efficiency, and facilitating collaboration across different departments. Some the examples include Enterprise Resource Planning, Customer Relationship Management, Business Intelligence, Project Management Systems among others. You are required to generate a short report (no exceeding 300 Words) on the context which should address the following:

* 1. Discuss any two database models used in the modern day enterprise computing applications with suitable examples.

# PART B 20 Marks

Consider the **RUAS Student Management System** to manage the details of students in RUAS. The computerized system enables the users to access students’ data at any time and from any place. The system consists of the functionalities such as Student Details, Branch Details, Fee Payment, Exam Results and any other student related information needed by the university. It is required to undertake the following activities:

**B2.1** List of functional requirements

**B2.2** Implementation of relational database schema with appropriate attributes, and constraints using SQL commands

**B2.3** Design and implementation of GUI

**B2.4** Connection of front end with the database and discussion on the results

**Note:** Make appropriate assumptions to make the specification complete.



**ANSWERS:**

1)

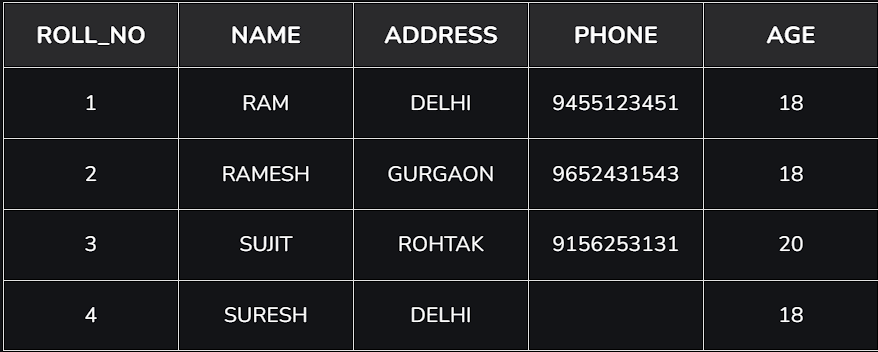
In modern enterprise computing applications, two prevalent database models are the relational database model and the Graph database model.

**Relational Database Model(RDBMS):**

A relational database stores related data in tables based on the relational model, where each row represents a record with a unique key. Columns hold attributes, facilitating relationships between data points. Tables, or relations, depict data and relationships. This model is record-based, widely adopted, and forms the foundation of many modern database systems.

The relational model offers several **advantages**, including simplicity, flexibility, security, data accuracy, integrity maintenance, and ease of applying operations. It provides a straightforward structure for organizing data into tables with relationships, supporting various operations like data definition, manipulation, and transaction management. However, it also has **limitations**, such as inefficiency with large databases, occasional difficulty in finding table relations, and slower query response times due to complex structures. Overall, the relational model represents data in rows and columns, ensuring distinct attribute representation and single-entity per row, forming the backbone of many database systems.

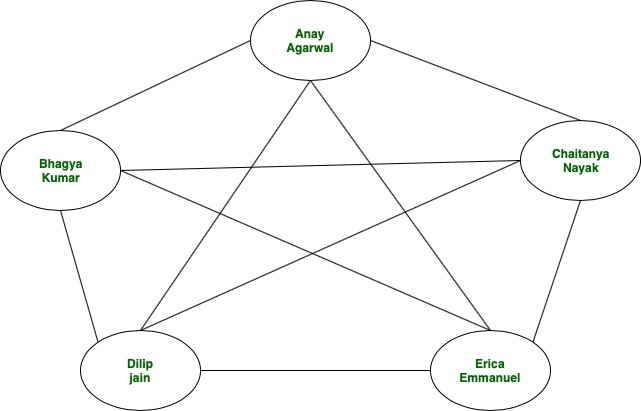
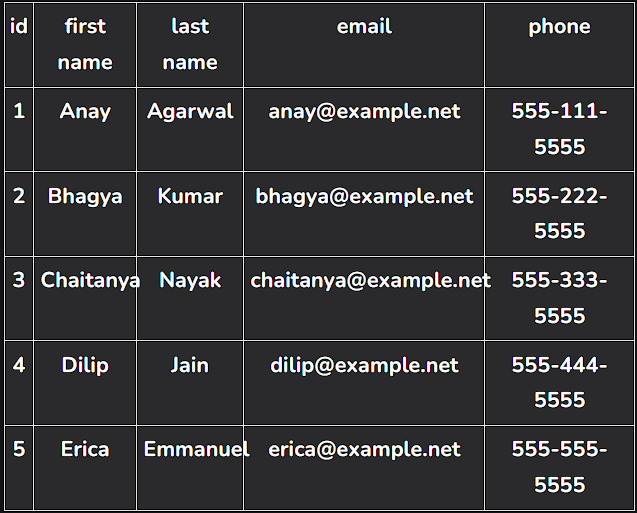
**Example** The relational model represents how data is stored in Relational Databases. A relational database consists of a collection of tables, each of which is assigned a unique name. Consider a relation STUDENT with attributes ROLL\_NO, NAME, ADDRESS, PHONE, and AGE shown in the table.



# Graph Database:

A graph database (GDB) is a database that uses graph structures for storing data. It uses nodes, edges, and properties instead of tables or documents to represent and store data. The edges represent relationships between the nodes. This helps in retrieving data more easily and, in many cases, with one operation. Graph databases are commonly referred to as a NoSQL.

**Example** We have a social network in which five friends are all connected. These friends are Anay, Bhagya, Chaitanya, Dilip, and Erica. A graph database that will store their personal information may look something like this:

Now, let’s analyse the time taken in this Relational database approach. This will be approximately log(N) times where N represents the number of tuples in friendship table or number of relations. Here, the database maintains the rows in the order of id’s. So, in general for ‘M’ no of queries, we have a time complexity of **M\*log(N)** Only if we had used a graph database approach, the total time complexity would have been O(N).

**Advantages:** Frequent schema changes, managing  volume of data, real-time query response time, and more intelligent data activation requirements are done by graph model.

**Disadvantages:** Note that graph databases aren’t always the best solution for an application. We will need to assess the needs of application before deciding the architecture.

**Limitations of Graph Databases:**Graph Databases may not be offering better choice over the NoSQL variations.If application needs to scale horizontally this may introduces poor performance.

B)

**List of Functional Requirements for RUAS Student Management System:**

- the user should have the ability to display the list of all the students that are included in the student managemment system

- the user should have the ability to search for any specific student based on any of the parameters like name, student id, result status, fee status, etc.

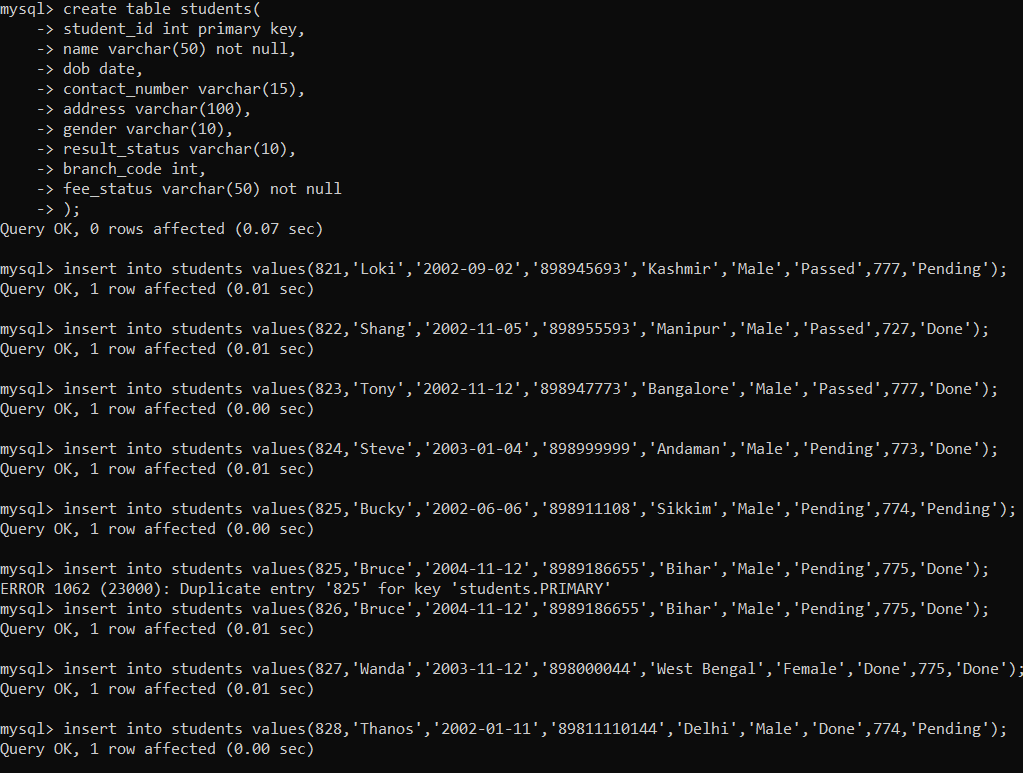
- the user should have the ability to access the database.

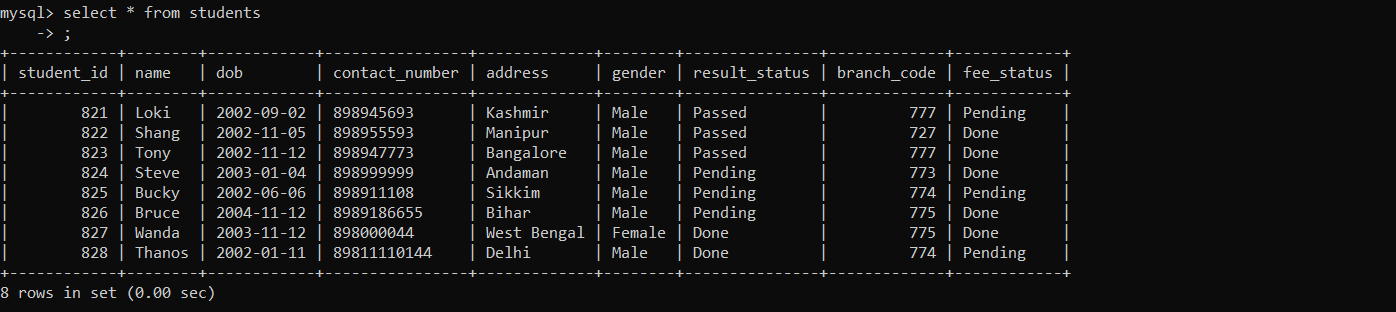
- the user should have the ability to allow adding, updating, and deleting student records.

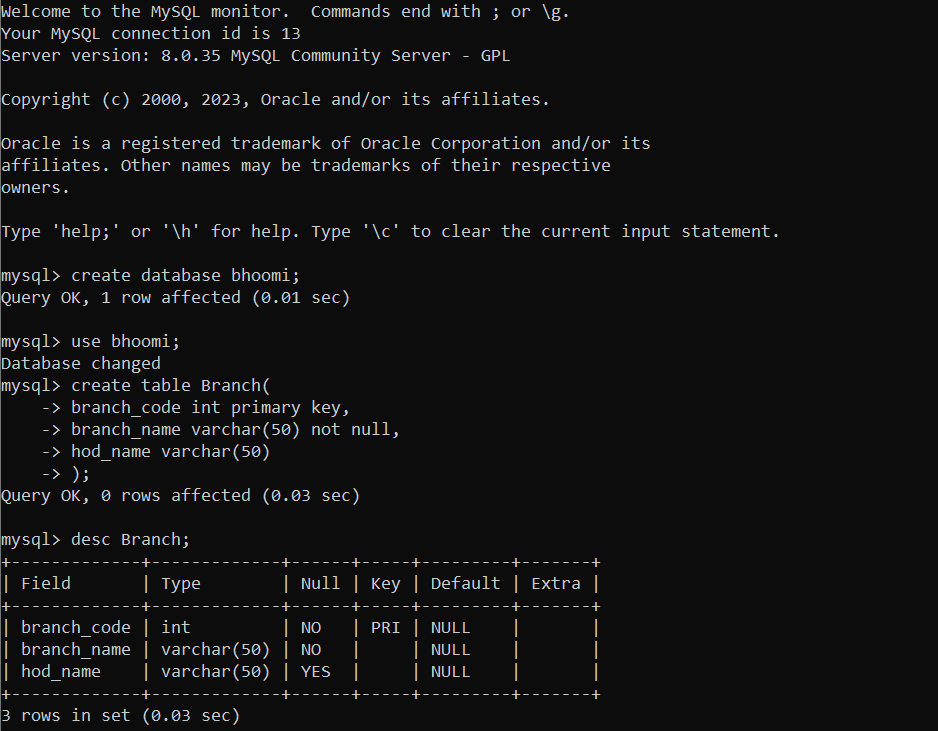
- the user should should have the ability to generate unique roll numbers for each student.

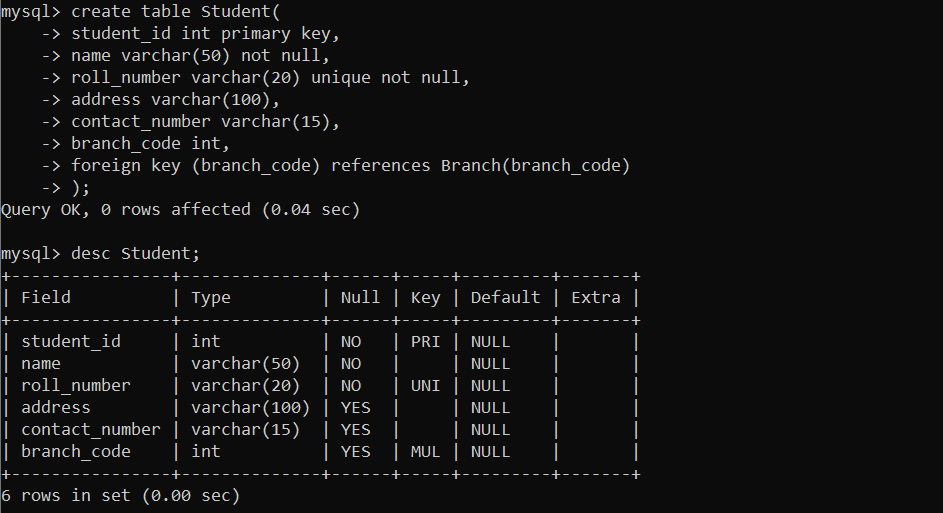
**Implementation of Relational Database with MySQL commands:**

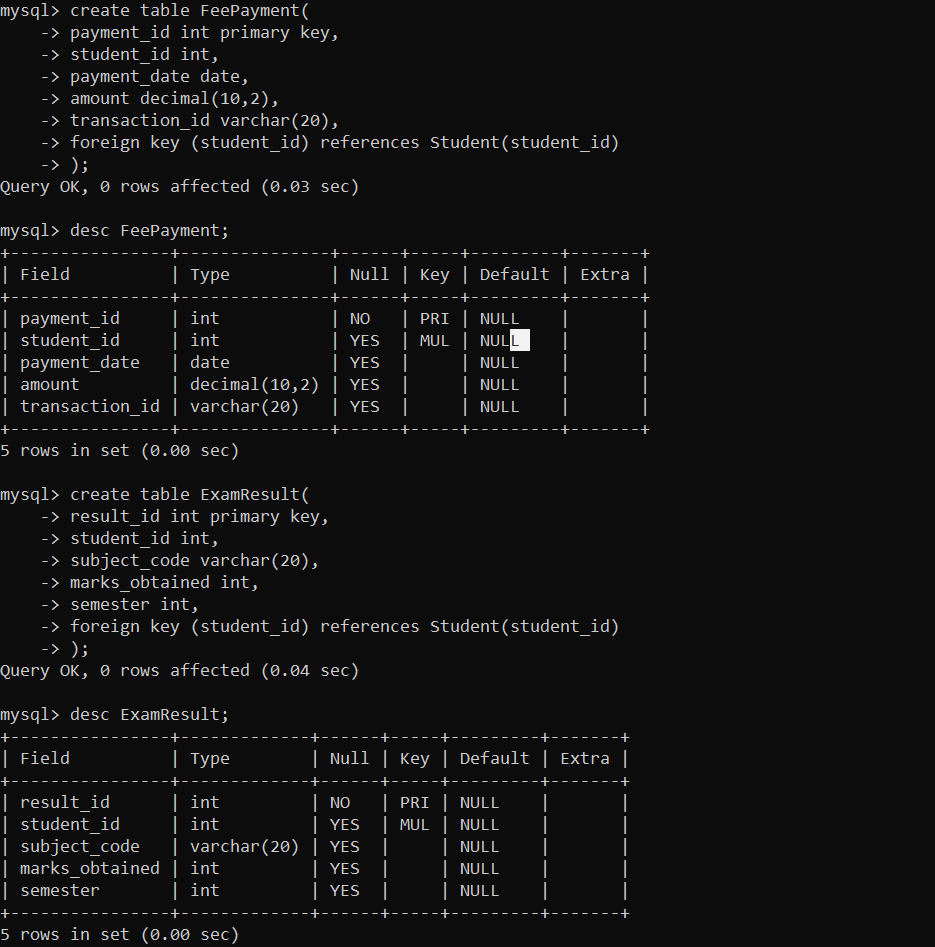
Creating tables:

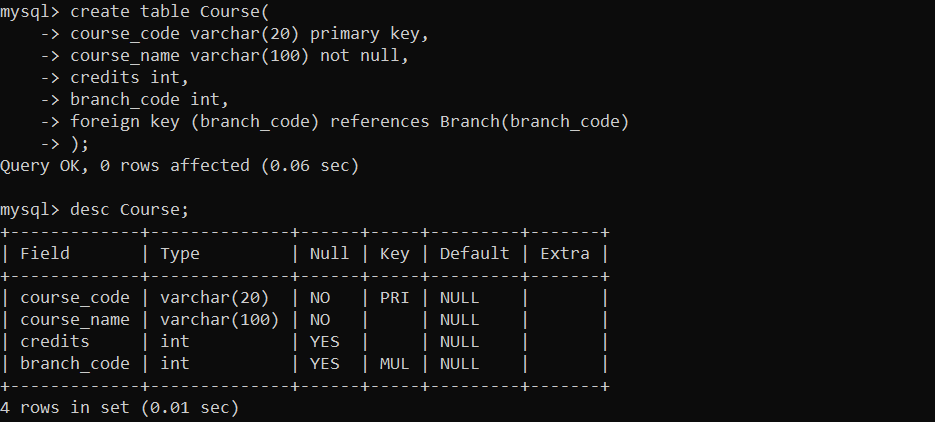


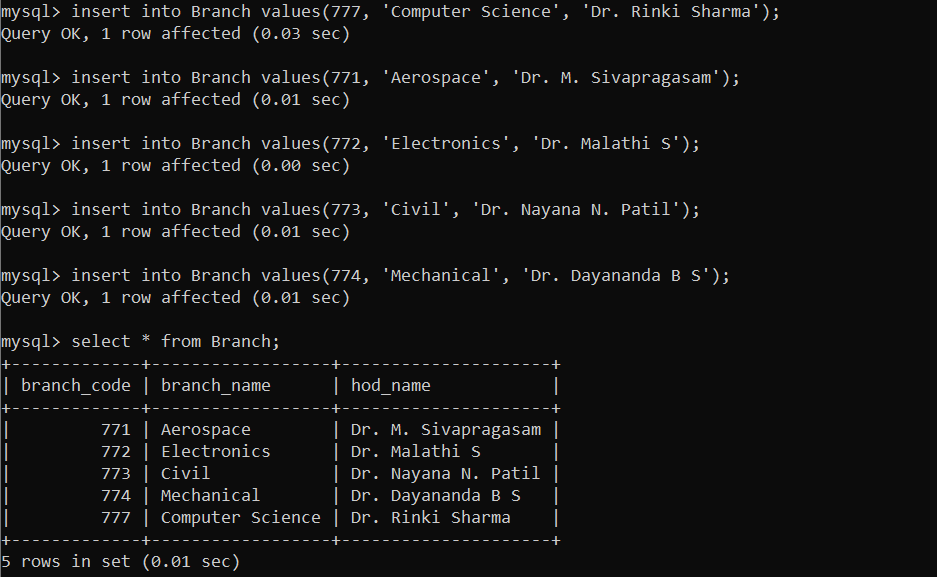




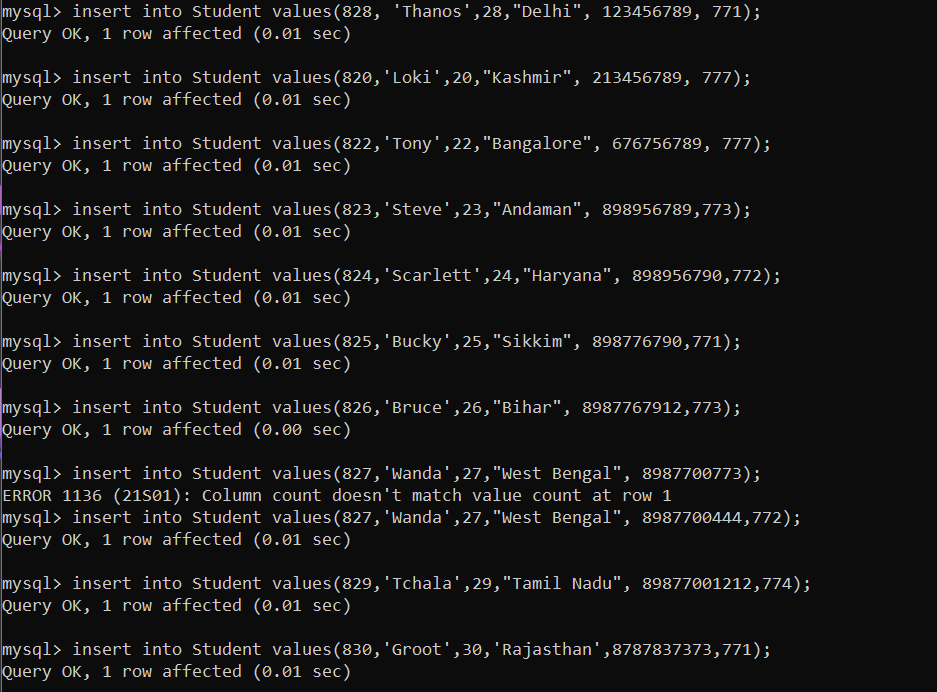


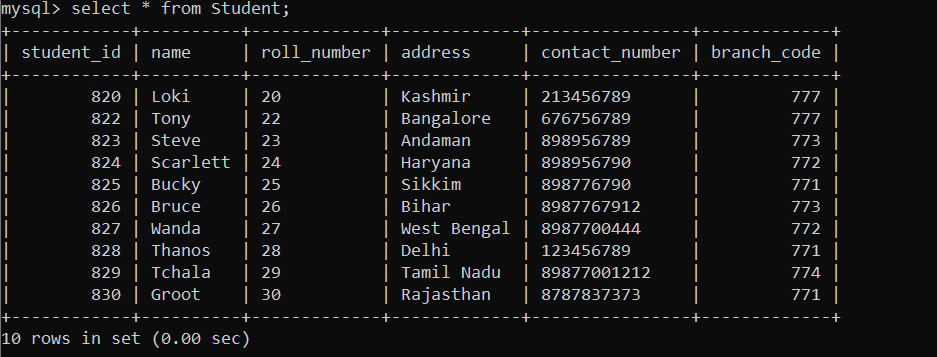




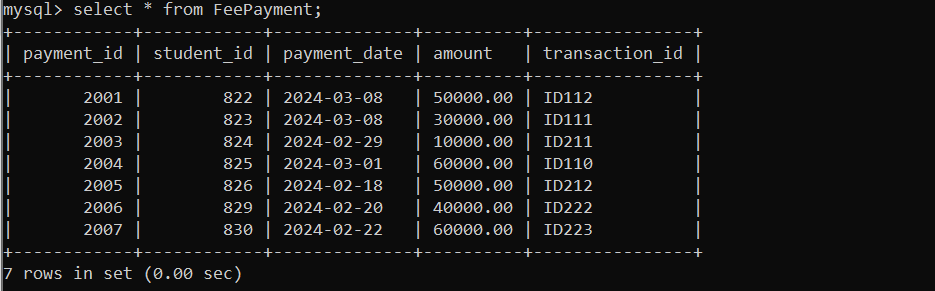


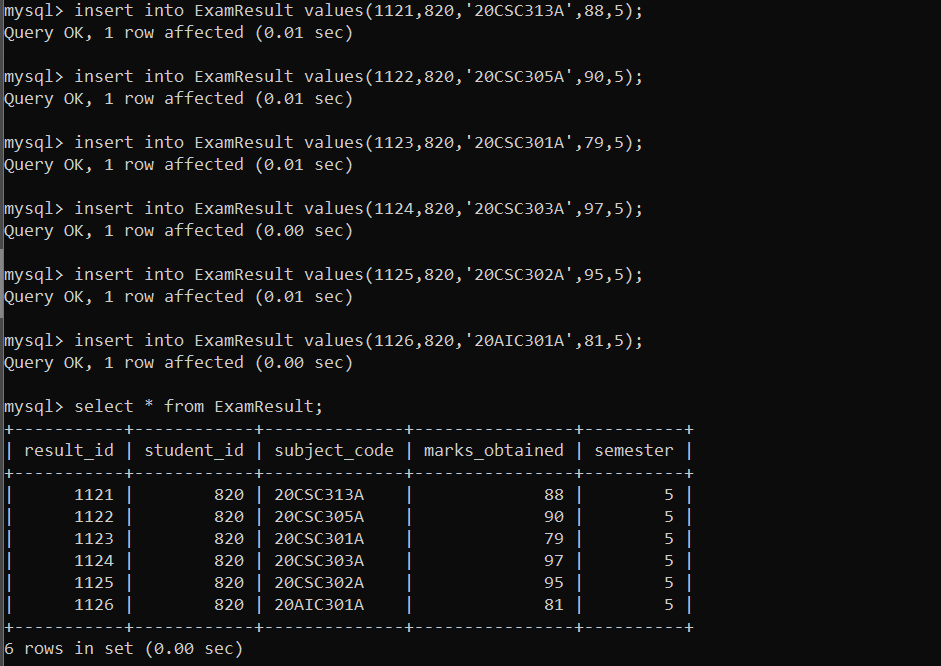
Inserting tuples:

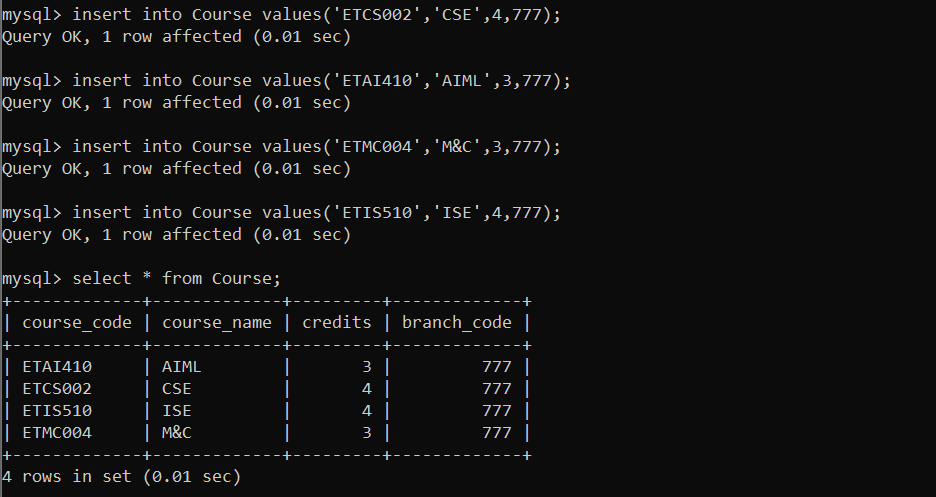




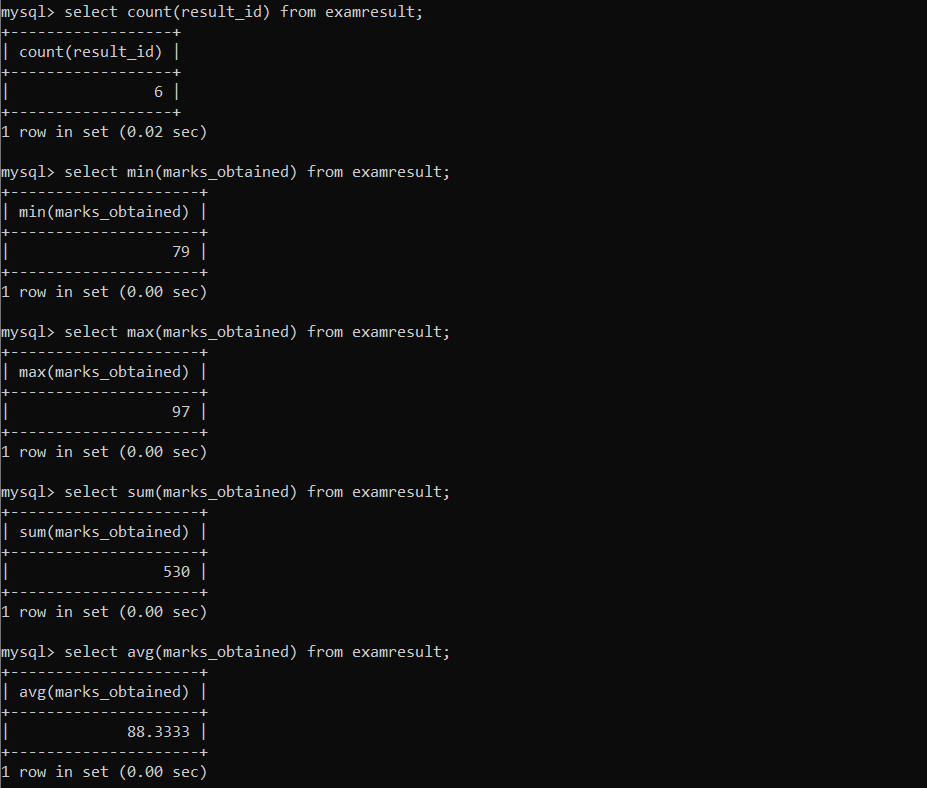


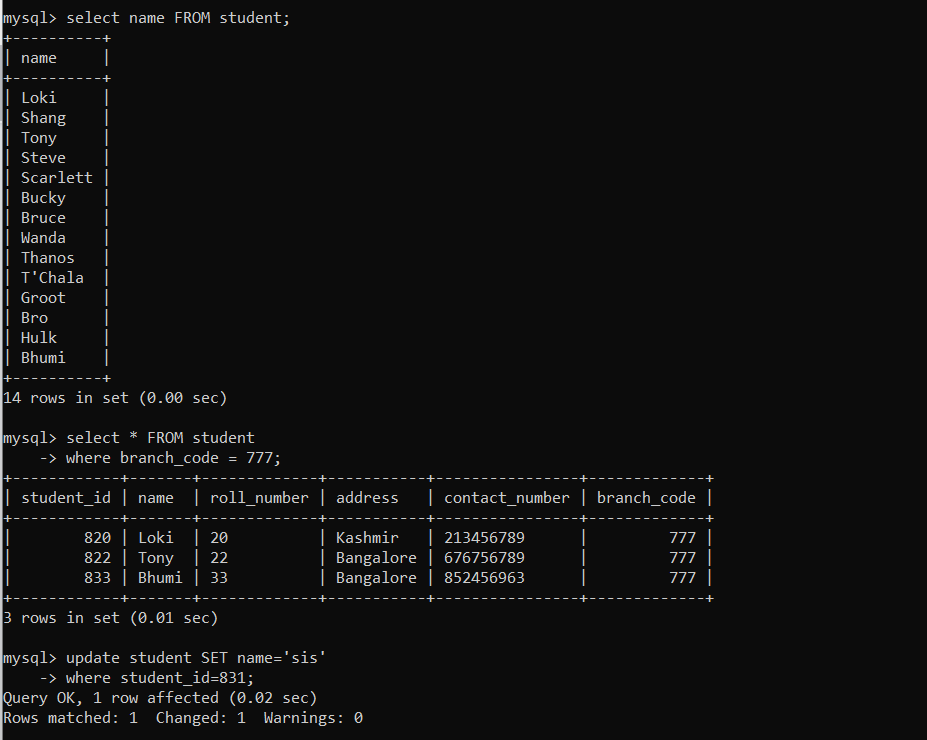


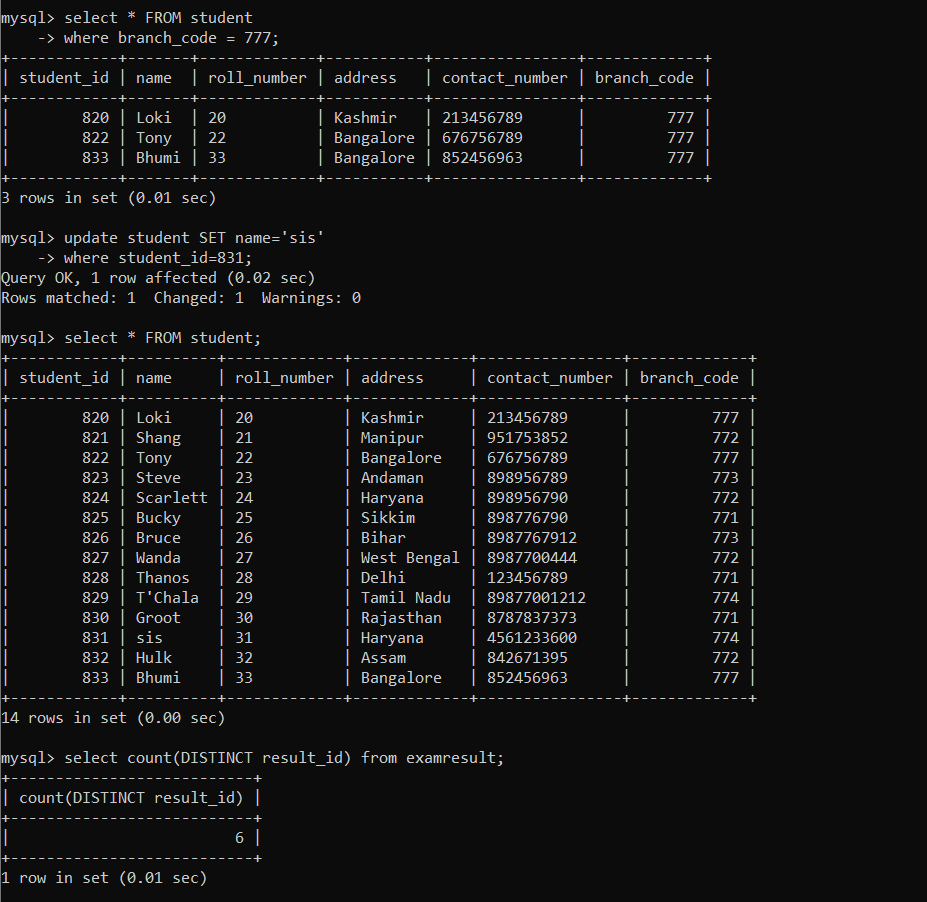


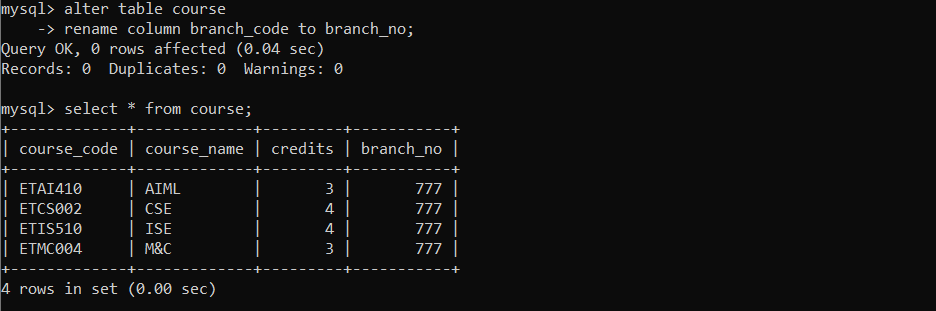


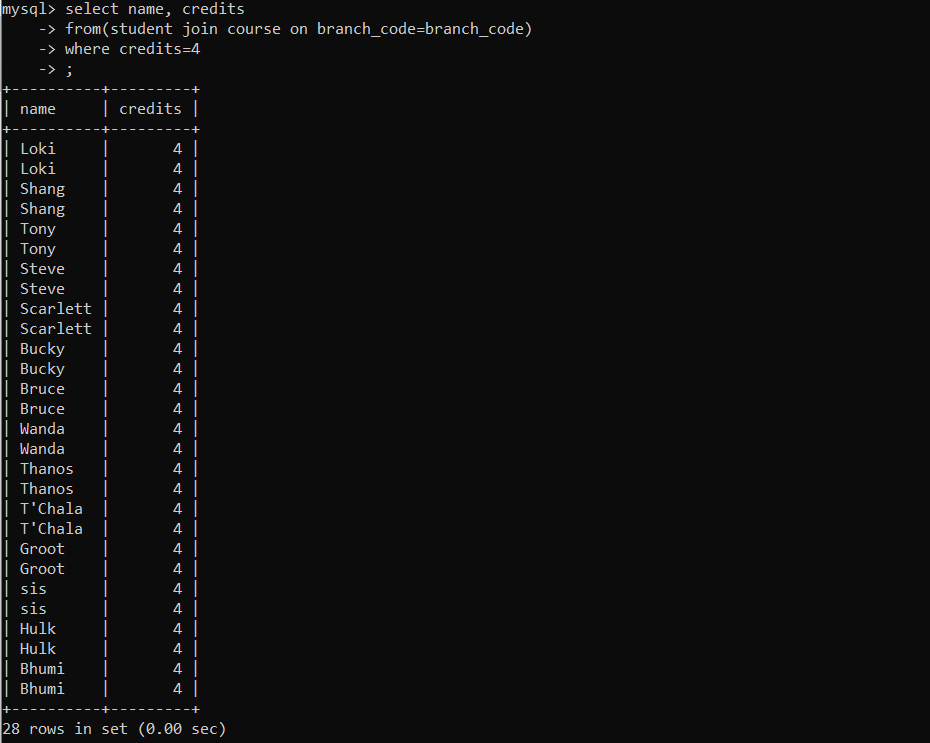
Performing different MySql Operations on tables:

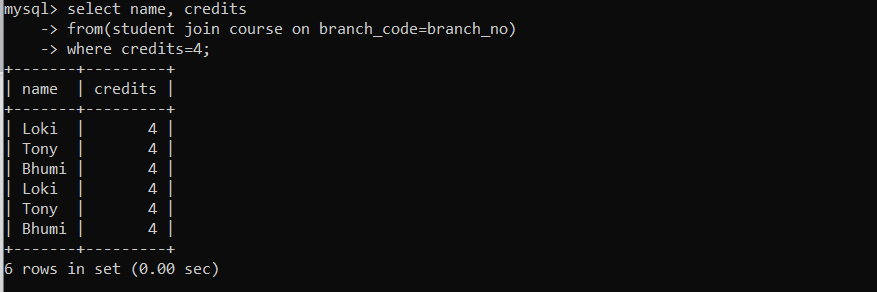




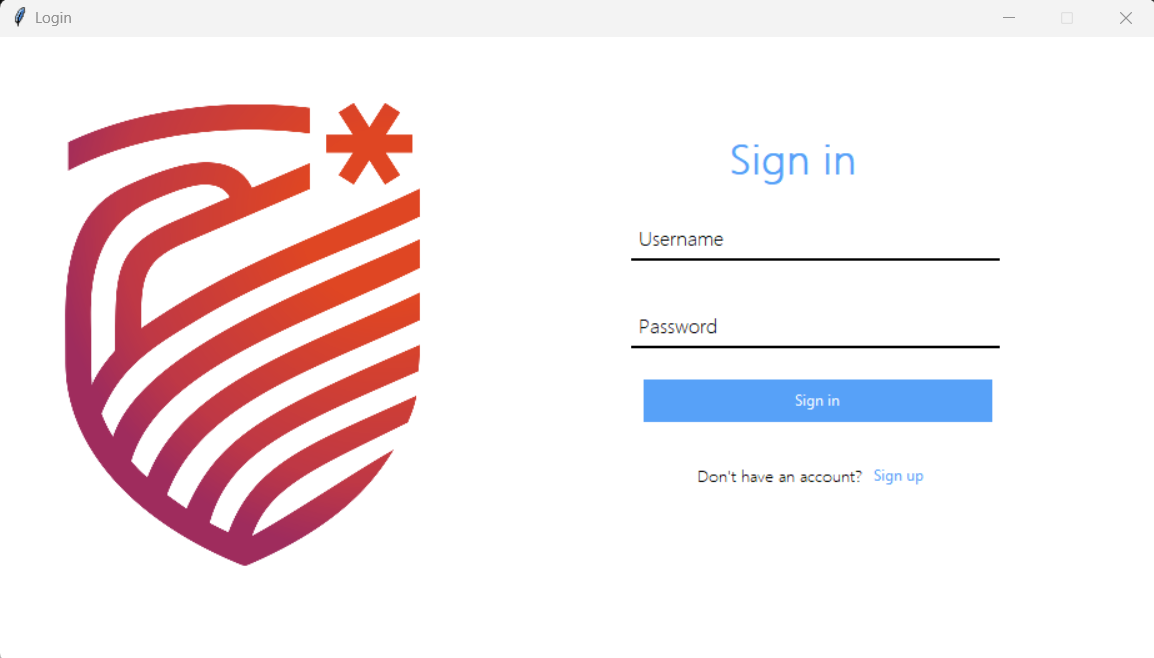


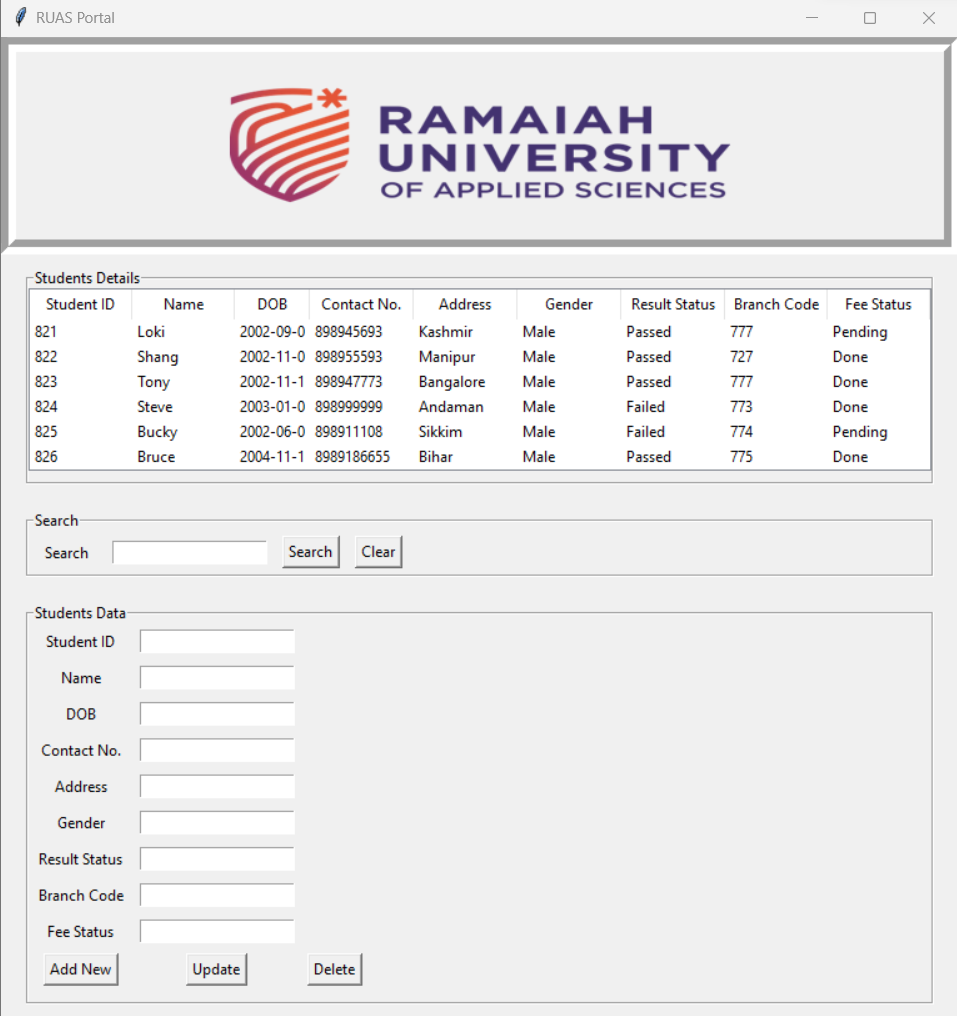






**Design and Implementation of GUI:**





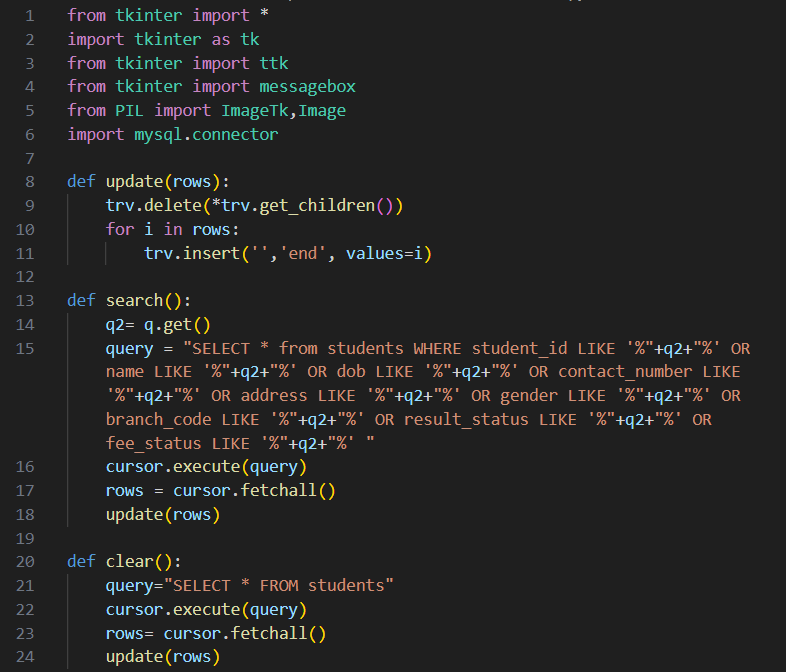
**Code for Login Page:**







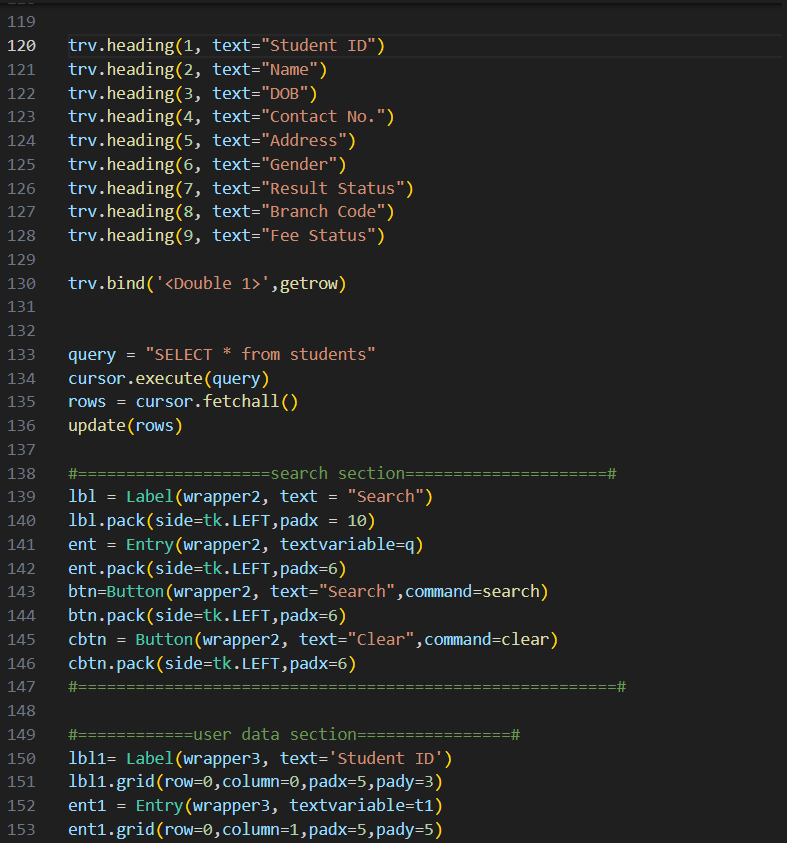
**Connection of Front end with Database:**

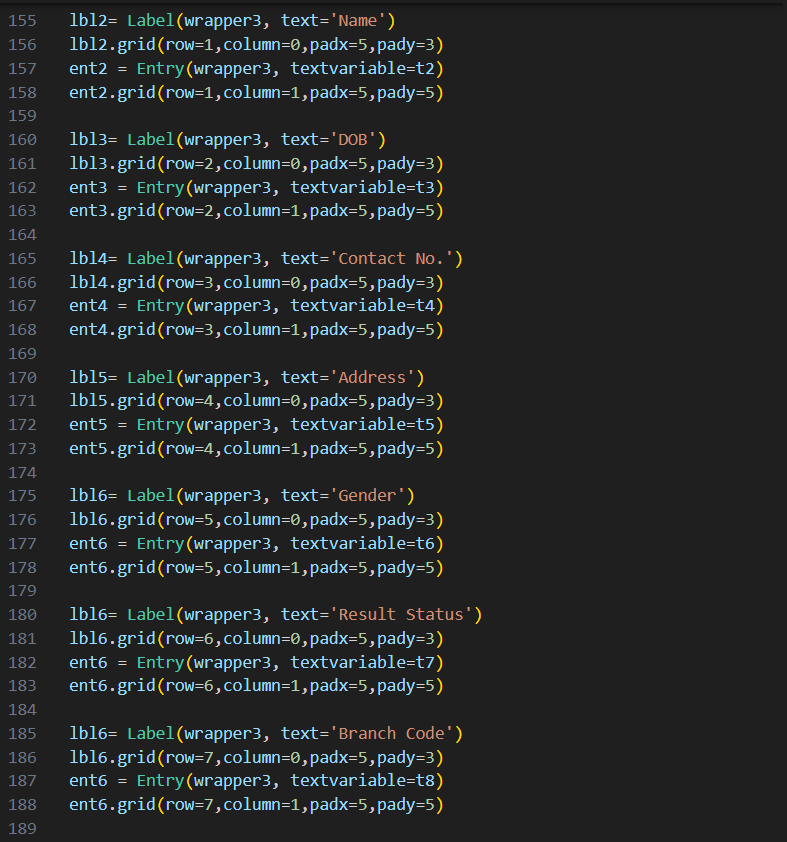


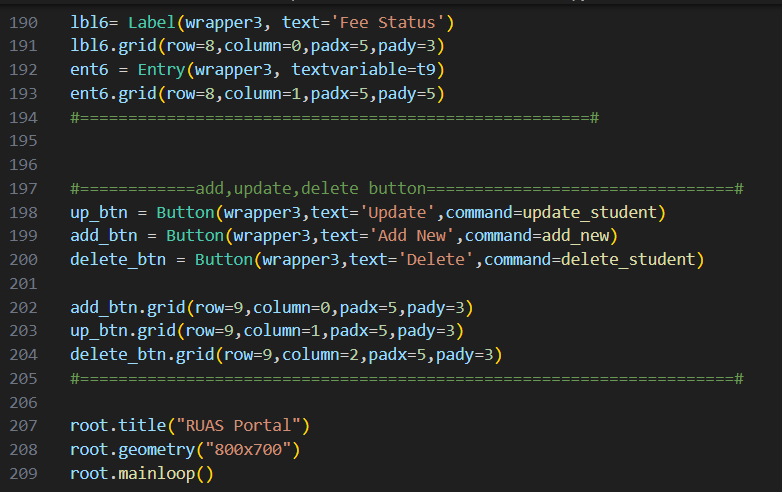












**Conclusion:**

A well-designed student database management system (DBMS) crafted with Tkinter and MySQL presents an effective solution for organizing, managing, and retrieving student information efficiently. Through the development process, we have successfully integrated Tkinter's user-friendly interface with MySQL's robust database management capabilities, resulting in a comprehensive system tailored to meet the needs of educational institutions.

By leveraging Tkinter's GUI toolkit, we have created an intuitive interface that allows users to interact with the database seamlessly. Through intuitive forms and controls, users can easily input, update, and retrieve student data with minimal effort. Additionally, Tkinter's versatility enables us to design a visually appealing interface that enhances user experience and promotes ease of navigation.

Furthermore, by harnessing the power of MySQL, we have established a secure and reliable database backend to store and manage student records. MySQL's scalability and performance ensure that the system can accommodate large volumes of data without compromising speed or efficiency. With features such as data integrity constraints, indexing, and relational database management, MySQL enables us to maintain the integrity and consistency of student data while facilitating efficient data retrieval and analysis.

In conclusion, the integration of Tkinter and MySQL has enabled us to develop a robust student database management system that streamlines administrative tasks, enhances data organization, and improves overall productivity within educational institutions. Whether it's managing student enrollment, tracking academic progress, or generating reports, our system provides a comprehensive solution that meets the diverse needs of educators, administrators, and students alike. Through continued refinement and adaptation, we are committed to evolving our system to ensure it remains a valuable asset in the educational landscape.

**References:**

[https://www.google.com/](http://www.google.com)

[https://geeksforgeeks.org/](http://www.google.com)

<https://www.wikipedia.org/>

[https://www.youtube.com/](https://www.wikipedia.org/)