# 

REPORT ON

Sumaiah Binta Musa || Istyaque Ahammed || Samundra Dhakal

190205 || 200240 || 200245

Dining Management System

KHULNA UNIVERSITY

**Contents:**

[**Introduction:** 2](#_Toc131685358)

[**Motivation:** 2](#_Toc131685359)

[**Interesting queries:** 2](#_Toc131685360)

[**Entity Relationship Diagram:** 3](#_Toc131685361)

[**Schema Design:** 4](#_Toc131685362)

[**Schema Diagram:** 5](#_Toc131685363)

[**Data Entry:** 6](#_Toc131685364)

[Student table: 6](#_Toc131685365)

[Payments table: 6](#_Toc131685366)

[Meals table: 7](#_Toc131685367)

[Manager table: 7](#_Toc131685368)

[Reports table: 7](#_Toc131685369)

[**Normalization:** 8](#_Toc131685370)

[Student Table: 8](#_Toc131685371)

[Payments table: 8](#_Toc131685372)

[Meals table: 8](#_Toc131685373)

[Manager table: 8](#_Toc131685374)

[Reports table: 9](#_Toc131685375)

[**Design:** 9](#_Toc131685376)

[Registration Form: 9](#_Toc131685377)

[Payment Form: 9](#_Toc131685378)

[Data: 10](#_Toc131685379)

[Interesting Query: 10](#_Toc131685380)

[Tools and Technologies: 11](#_Toc131685381)

# **Introduction:**

Database system for converting dining hall meal receipt and payment process from manual to online. Students can pre-order lunch and dinner with pre-payment facility using the system. Easy order management facility using database for dining managers and users.

# **Motivation:**

The subject of our project is Khulna University Hall Dining Management System.

There are total five halls in Khulna University. Three halls for boys and two for girls. Each hall has a separate dining room. Students can eat two meals in the dining hall. Dining is mainly available for lunch and dinner. Every day there are different types of menu of food. Moreover, the fest is given twice every month. Good food is provided on that day.

Every night the meal starts from 8.00 pm. Every day students have to pay a certain amount of money to take next day meal. 65 taka for two time meals (lunch and dinner). And 35 takas have to be paid for one meal (lunch or dinner). Most of the time student give 100 taka and then money change problem arises. Moreover sometimes many students stay outside of the hall. But they want to take the next day's meal. But it is not possible for him. Also, the dining manager has to face problem to keep track of everything. It is also difficult to calculate the amount of money every day. This is a huge waste of his time. He/she had to sit up to late-night to take the payment of everyone's meal.

To solve these problems we want to make online based dining management system.

Here we can get rid of all these problems. There will be an opportunity to place food orders online every day through online websites or apps. As a result, students will not be in trouble. Even if he/she is not in hall, he can order next day meal. The dining manager does not have to sit up to late-night. As a result, his/her time will be saved. Later there will be option to make payment through online payment gateway. There, the student's information and the amount of money will be calculated through the database system. So, the manager will not waste extra time. He/she will get the hard copy by downloading it the next day before market. Moreover, there will be no problem of money change as there is an option of online payment.

Therefore, dining management system is required for both students and dining manager.

# **Interesting queries:**

In our system there can be two types of query. One is from manager side and another is from user side. Those are mentioned below-

Manager queries:

1. Total amount of meals per day.
2. Total amount of money per day.
3. List of students of taking every day meal.
4. Which students take highest meal per month?
5. In which day of week highest number of students take meal.

User queries:

1. How much money he/she spend for meal per month.
2. How many days he/she takes meal in a particular month.

# **Entity Relationship Diagram:**

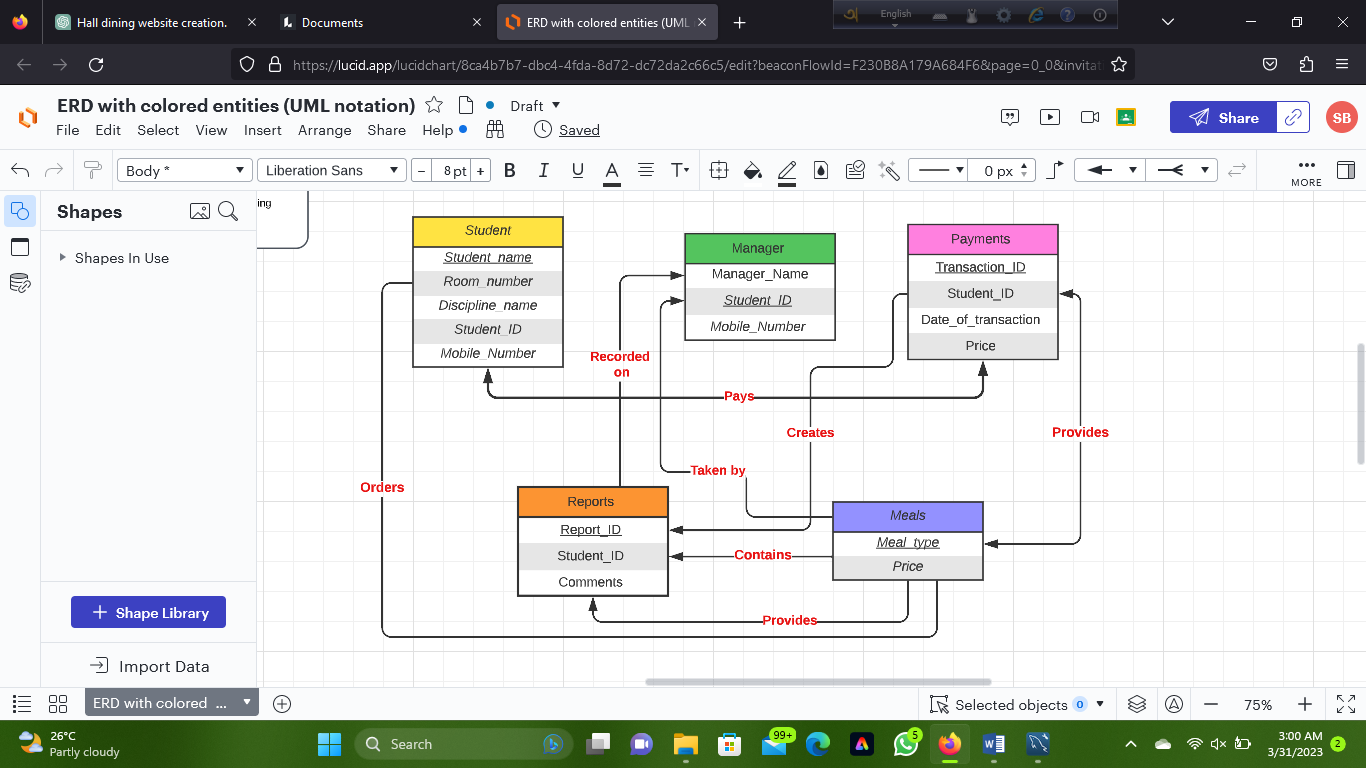


Fig: E-R Diagram

# **Schema Design:**

SQL Command:

create database Dining\_Management\_System;

Use Dining\_Management\_System;

create table Student

(Student\_name varchar(20),

Room\_number varchar(3),

Discipline\_name varchar(20),

Student\_ID numeric(6),

Mobile\_Number numeric(11),

primary key (Student\_ID) );

create table Meals

(Meal\_type varchar(6),

Price numeric(2,2),

primary key (Meal\_type) );

create table Payments

(Transaction\_ID varchar(15),

Student\_ID numeric(6),

Date\_of\_transaction date,

Price numeric(2,2),

primary key (Transaction\_ID) );

create table Manager

(Manager\_Name varchar(20),

Student\_ID numeric(6,0),

Moblie\_Number numeric(11),

primary key (Student\_ID)

);

create table Reports

(Report\_ID varchar(20),

Student\_ID numeric(6),

Comments varchar(20),

primary key (Report\_ID )

);

# **Schema Diagram:**

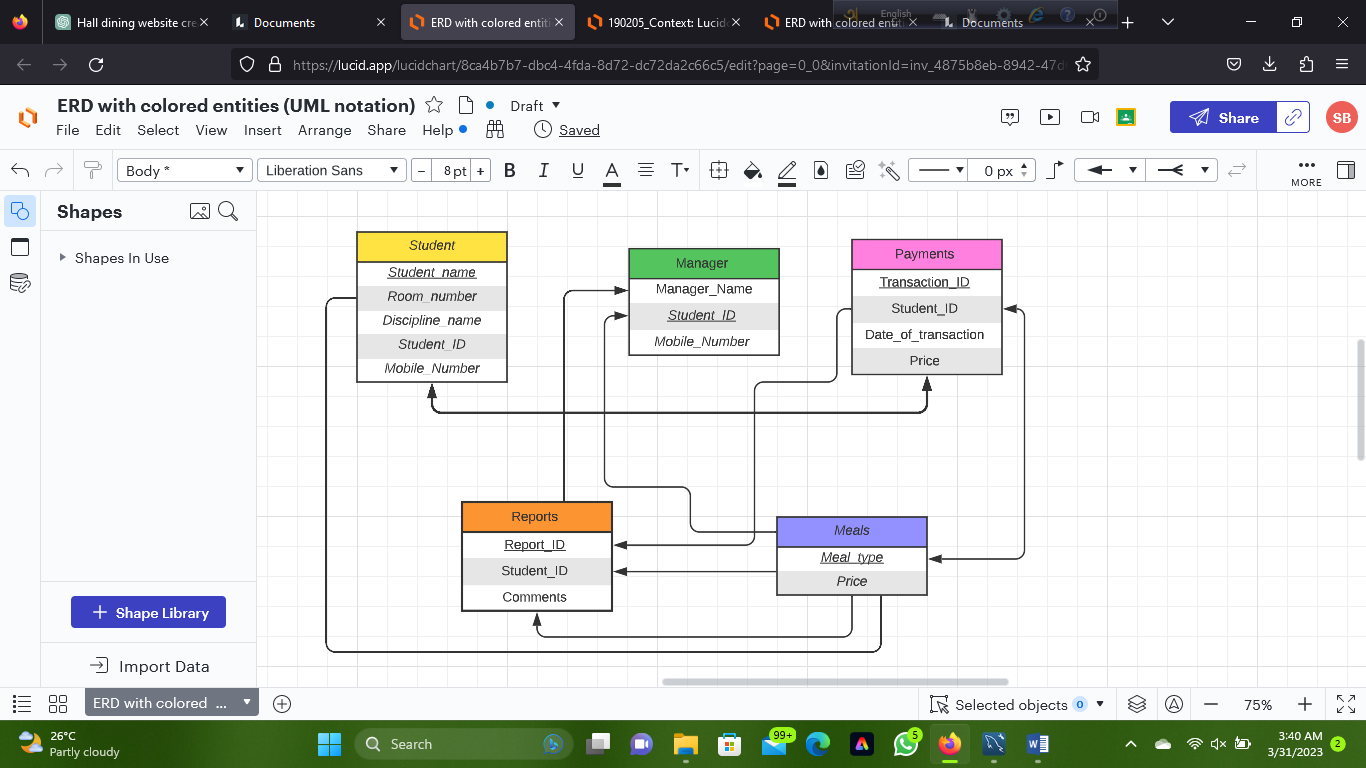


Fig: Schema Diagram

# **Data Entry:**

## Student table:

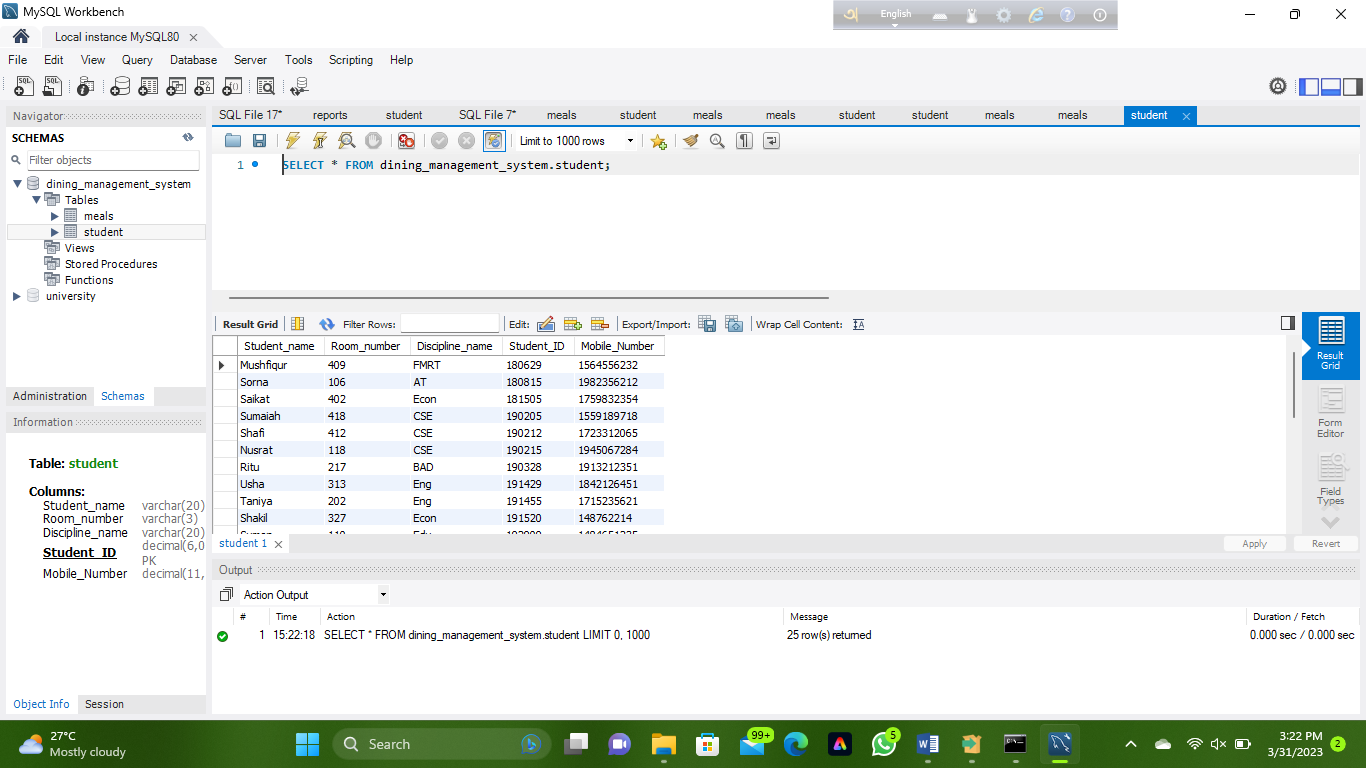


Fig: Student Table

## Payments table:

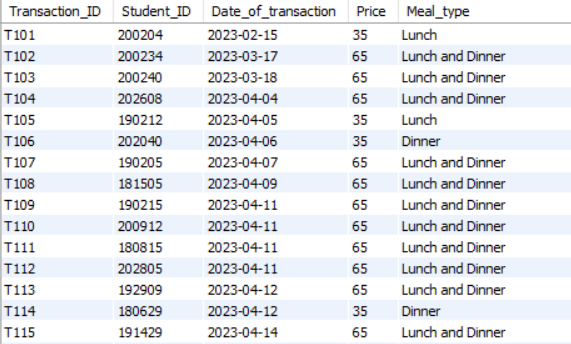


Fig: Payments Table

## Meals table:

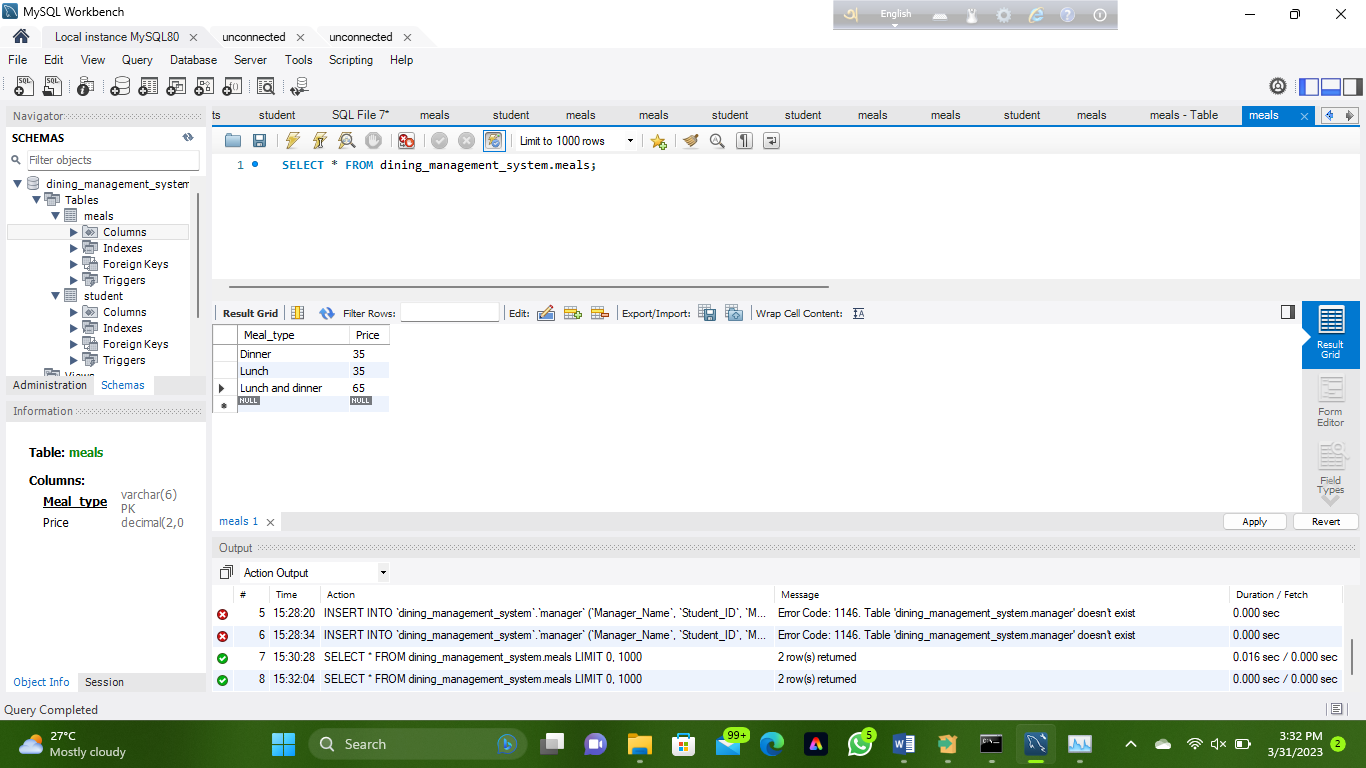


Fig: Meals Table

## Manager table:

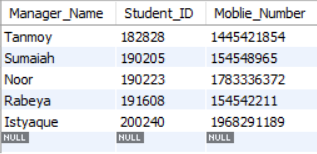


Fig: Manager Table

## Reports table:

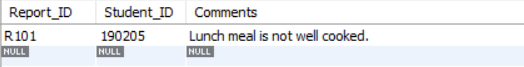


Fig: Reports Table

# **Normalization:**

## Student Table:

Student table satisfies the 1st Normal Form, because it has no repeating groups of data. Each column contains atomic values, which means that it cannot be further divided into sub-columns.

This table satisfies the 2nd Normal Form, because it is in 1NF and all non-key attributes are fully dependent on the primary key.

This table satisfies the 3rd Normal Form, because it is in 2NF and there are no non-key attributes that are dependent on another non-key attributes.

## Payments table:

Payments table satisfies the 1st Normal Form, because it has no repeating groups of data. Each column contains atomic values, which means that it cannot be further divided into sub-columns.

This table satisfies the 2nd Normal Form, because it is in 1NF and all non-key attributes are fully dependent on the primary key.

This table satisfies the 3rd Normal Form, because it is in 2NF and there are no non-key attributes that are dependent on another non-key attributes.

## Meals table:

Meals table satisfies the 1st Normal Form, because it has no repeating groups of data. Each column contains atomic values, which means that it cannot be further divided into sub-columns.

This table satisfies the 2nd Normal Form, because it is in 1NF and all non-key attributes are fully dependent on the primary key.

This table satisfies the 3rd Normal Form, because it is in 2NF and there are no non-key attributes that are dependent on another non-key attributes.

## Manager table:

Manager table satisfies the 1st Normal Form, because it has no repeating groups of data. Each column contains atomic values, which means that it cannot be further divided into sub-columns.

This table satisfies the 2nd Normal Form, because it is in 1NF and all non-key attributes are fully dependent on the primary key.

This table satisfies the 3rd Normal Form, because it is in 2NF and there are no non-key attributes that are dependent on another non-key attributes.

## Reports table:

Reports table satisfies the 1st Normal Form, because it has no repeating groups of data. Each column contains atomic values, which means that it cannot be further divided into sub-columns.

This table satisfies the 2nd Normal Form, because it is in 1NF and all non-key attributes are fully dependent on the primary key.

This table satisfies the 3rd Normal Form, because it is in 2NF and there are no non-key attributes that are dependent on another non-key attributes.

# **Design:**

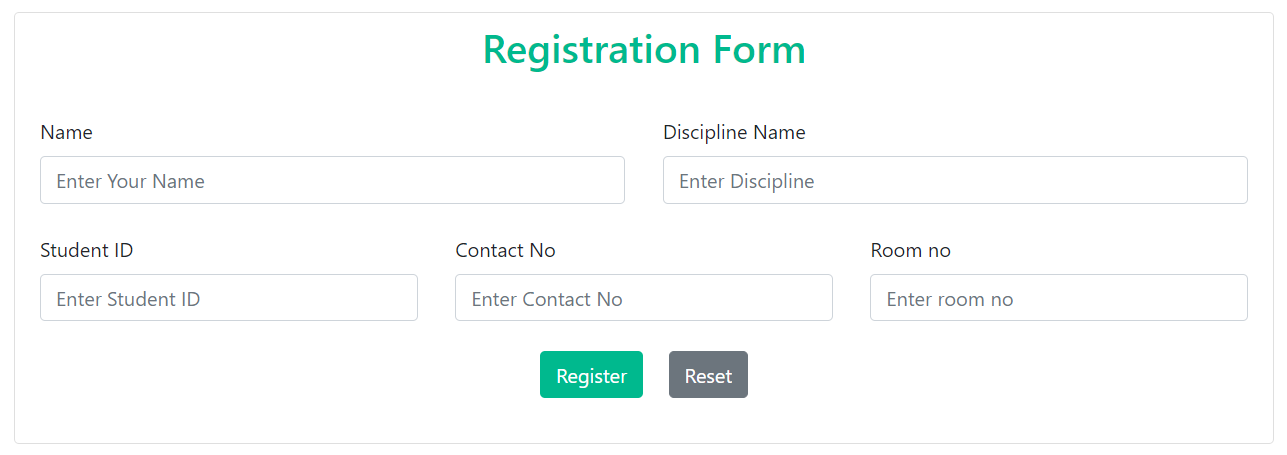
Registration Form:

Fig: Registration Form

## Payment Form:

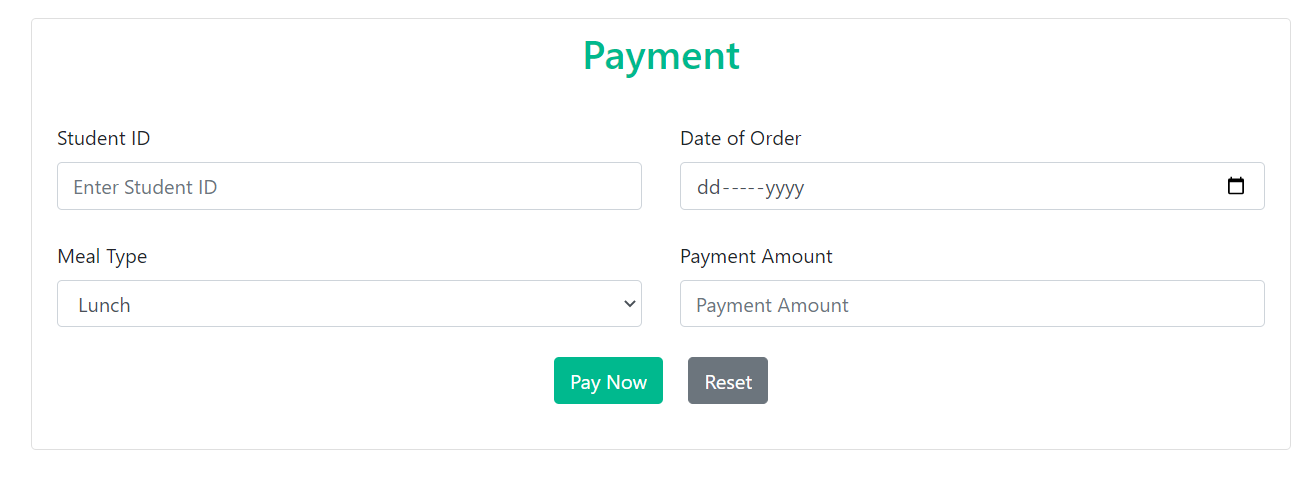


Fig: Registration Form

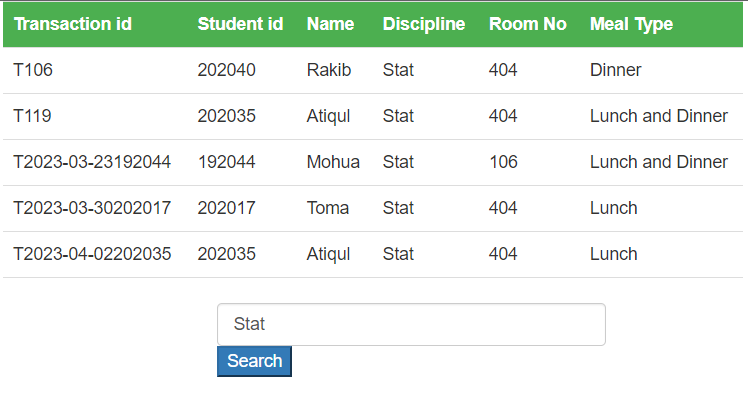
## Data:



Fig: Total Data

## Interesting Query:

* Student who take meal from Statistics Discipline



## Tools and Technologies:

To develop the whole system the tools and technologies we use are mentioned below-

Operating system : Windows 11 Home

Framework : .NET

Front end design : HTML, CSS

Back end design : PHP,

Workbench : MySQL

Connectors : jdbc, xampp