Group 12 - Database Project-Fall 2021

(Campus Eats Application)

Team Members-(Group 12)

- Jaissruti Nanthakumar - 801268098
- Janani Sri Adanur Rammohan - 801259018
- Ramanathan Sivaramakrishnan - 801243565
- Sunil Sundaram - 801221550

Campus Eats Application

Aim:

The main objective of this application is to provide schools and universities with a technology that helps them to deliver food within the campus by persons who are authorized by universities so that it may reduce the risk of infection spreading.

Proposed Solution:

We have designed a application named Campus_Eats where users can order food from restaurants that are near UNCC. We have taken 6 restaurants that are near UNCC for this project.And the delivery persons can be the students or the faculty from the university so that it will be easier for the administration to control who enters the campus.

Newly Added Tables in the Database

- ➔ MENU
- ➔ INVOICE
- ➔ PAYMENT
- ➔ ORDERED_ITEMS
- ➔ RESTAURANT_RATING
- ➔ DRIVER_RATING

Existing Table In the Table

- Delivery
- Driver
- Faculty
- Location
- Order
- Person
- Restaurant
- Staff
- Student
- Vehicle
- Staff
- Restaurant

Information On Added Tables

MENU - specifies information about item_id, restaurant_id, name and description of the food.

INVOICE - specifies the invoice_id, tip given to the driver, payment_payment_id(foreign key) for the order

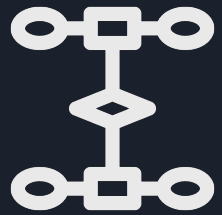
PAYMENT - specifies the payment_id, payment method, card details, order_order_id(foreign key) related to the payment.

ORDERED_ITEMS - specifies the order_id, item_id, quantity and special instructions on ordered items.

RESTAURANT_RATING - specifies information on food quality, service, atmosphere, affordability and restaurant rating

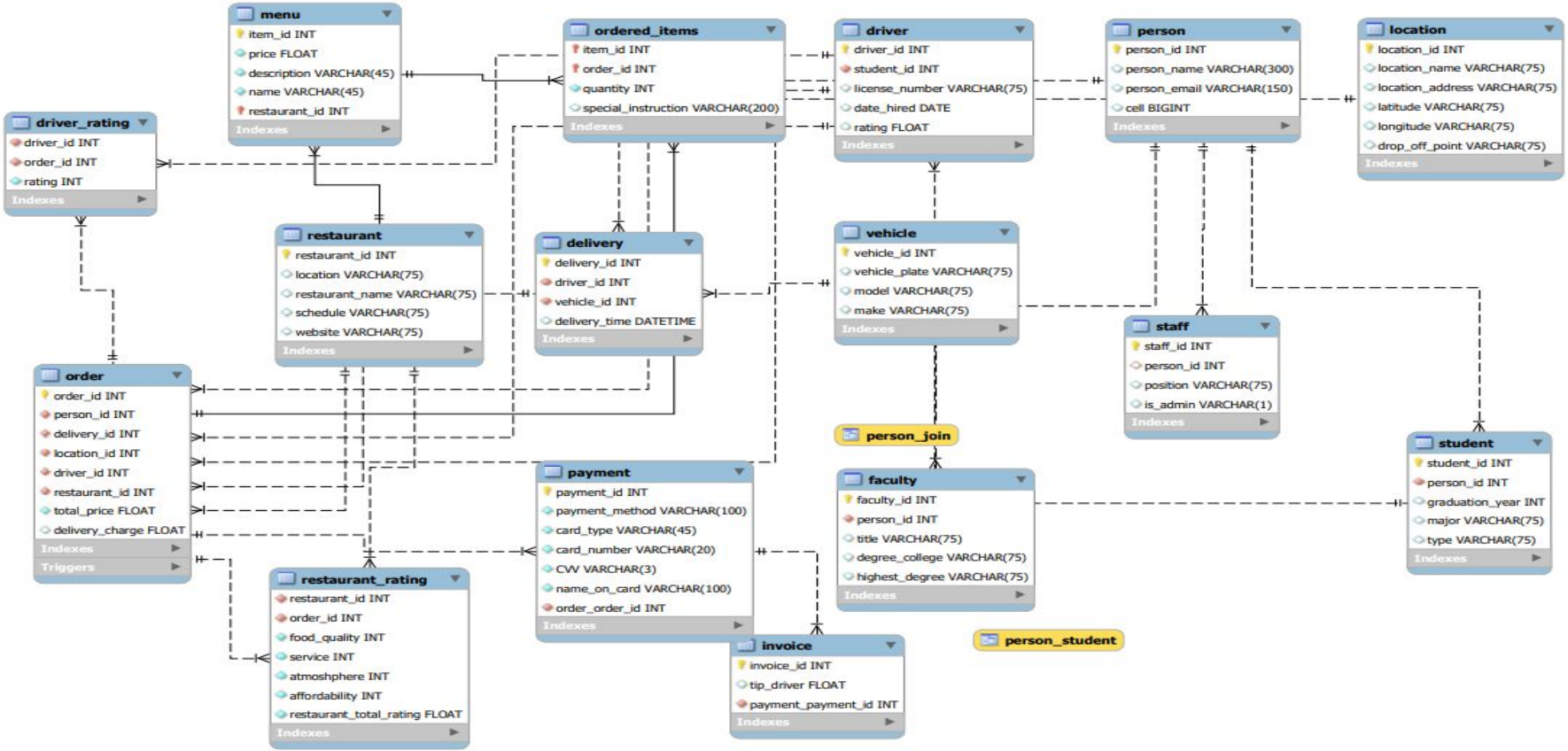
DRIVER_RATING - specifies information about driver id, order_id and rating.

Enhanced Entity Relationship Diagram(EERD)

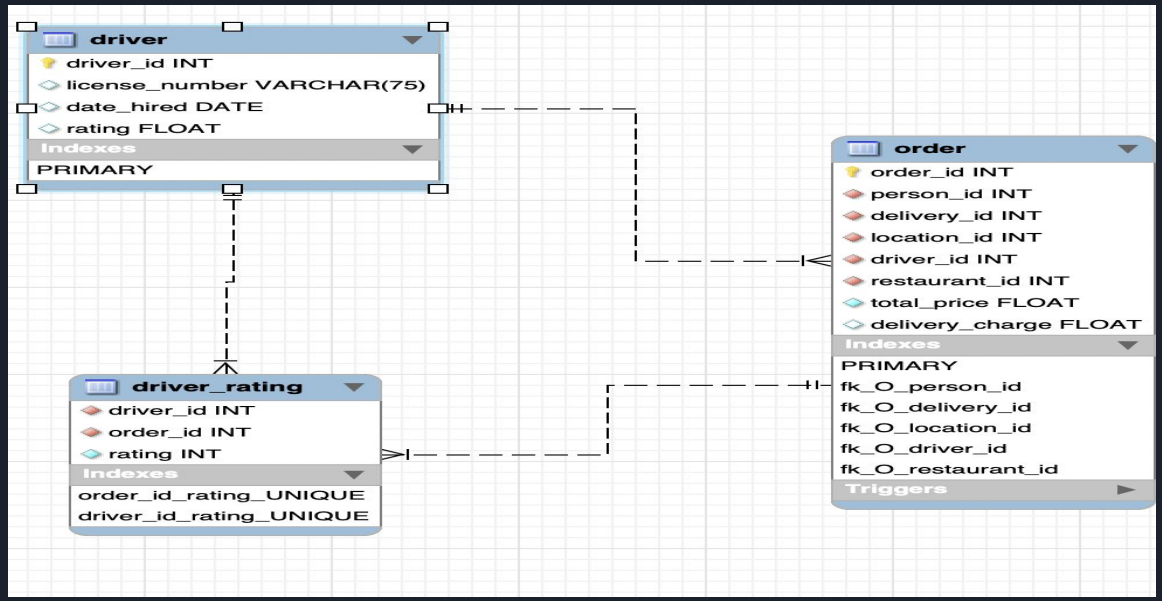


- An Entity Relationship (ER) Diagram is a type of flowchart that illustrates **how “entities” such as people, objects or concepts relate to each other within a system**.
- ER Diagrams are most often used to design or debug relational databases in the fields of software engineering, business information systems, education and research. Also known as ERDs or ER Models, they use a defined set of symbols such as rectangles, diamonds, ovals and connecting lines to depict the interconnectedness of entities, relationships and their attributes.
- They mirror grammatical structure, with entities as nouns and relationships as verbs.

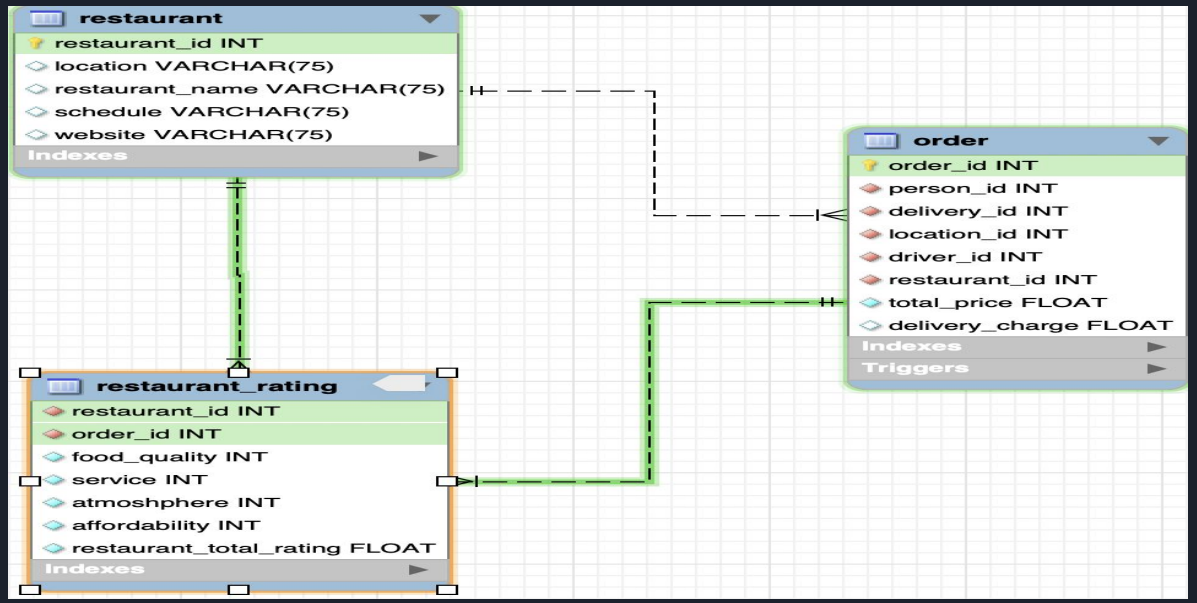
EERD Diagram for Campus_Eats



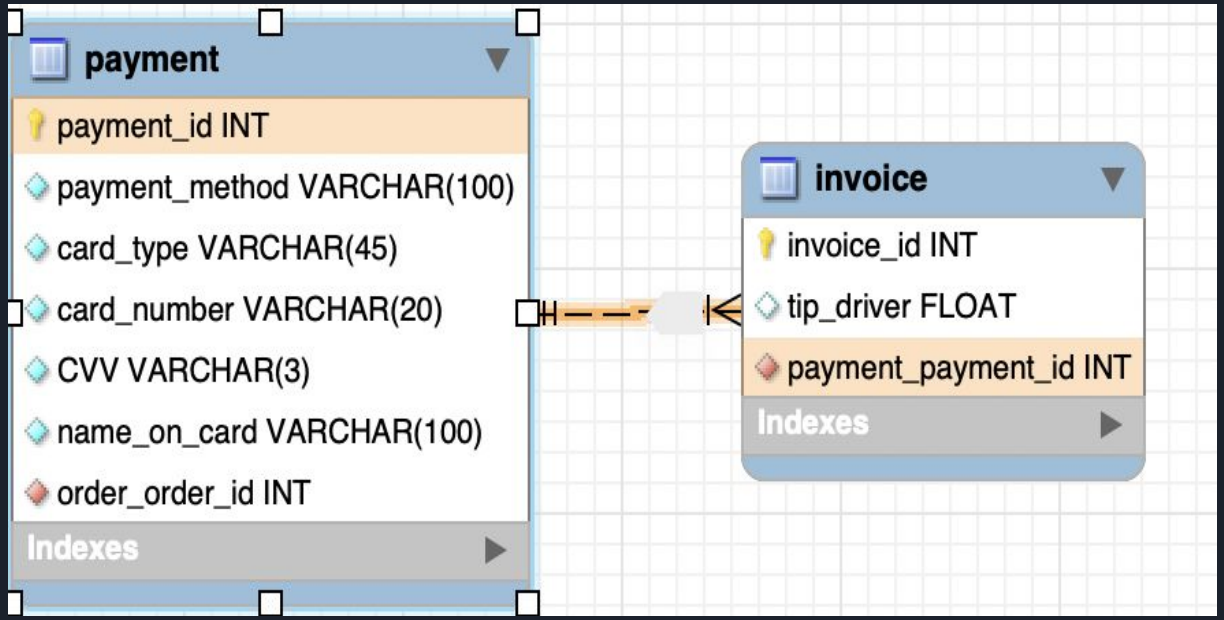
Relationship between driver_rating and other tables in database



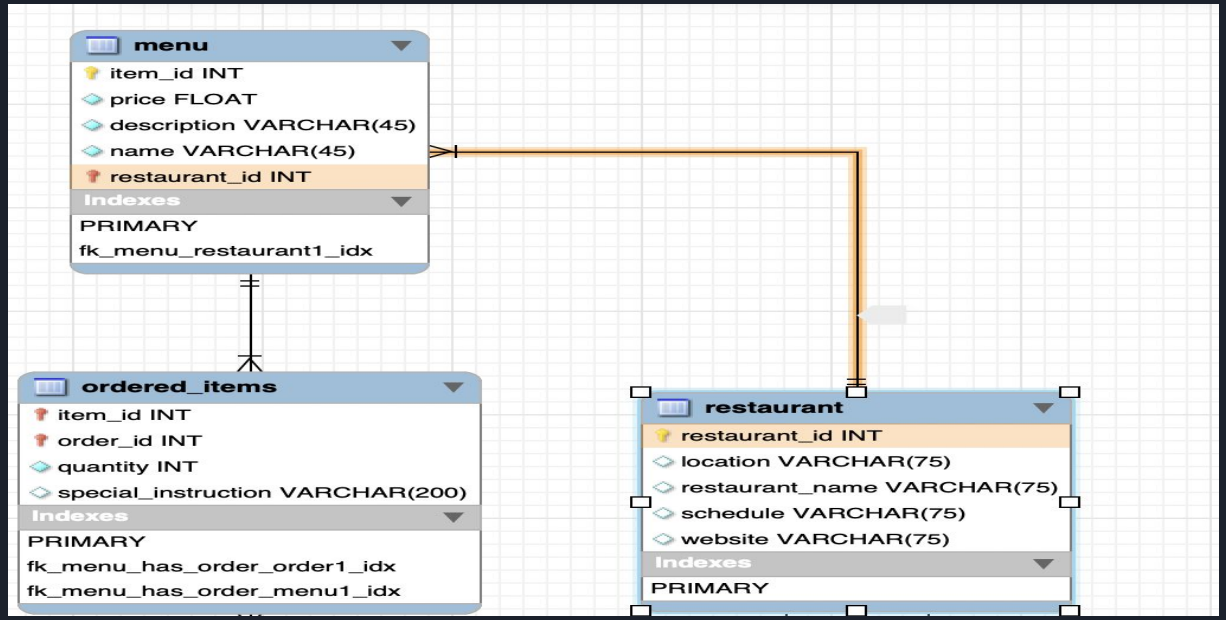
Relationship between restaurant_rating table and other tables in the database



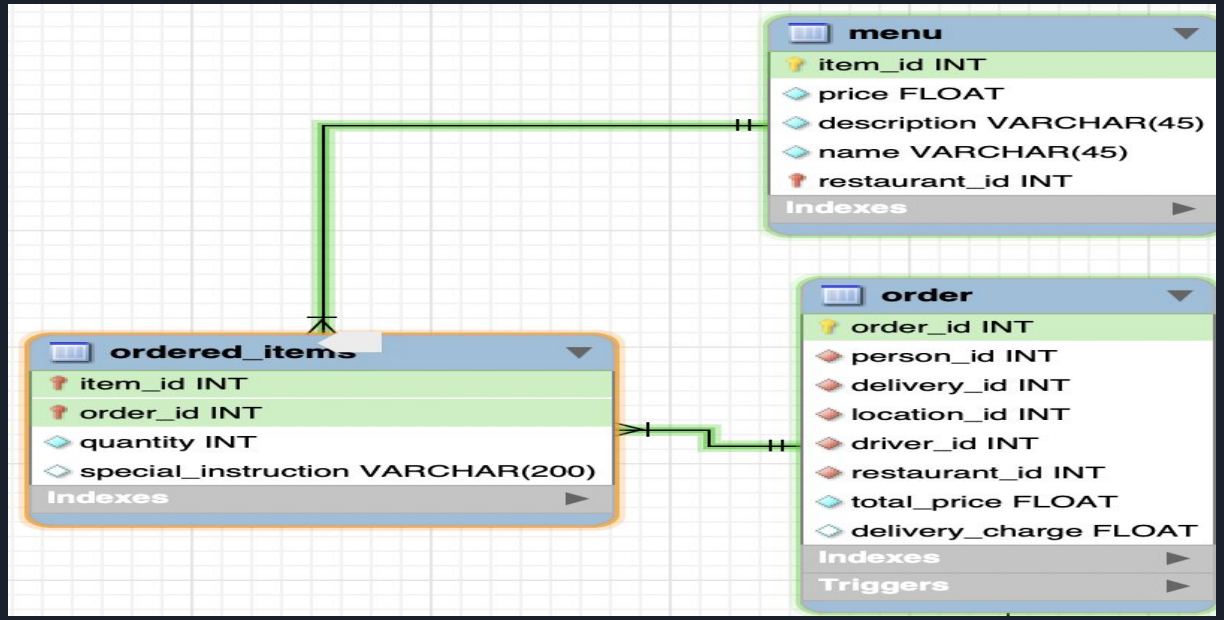
Relationship between payment table and other tables in the database



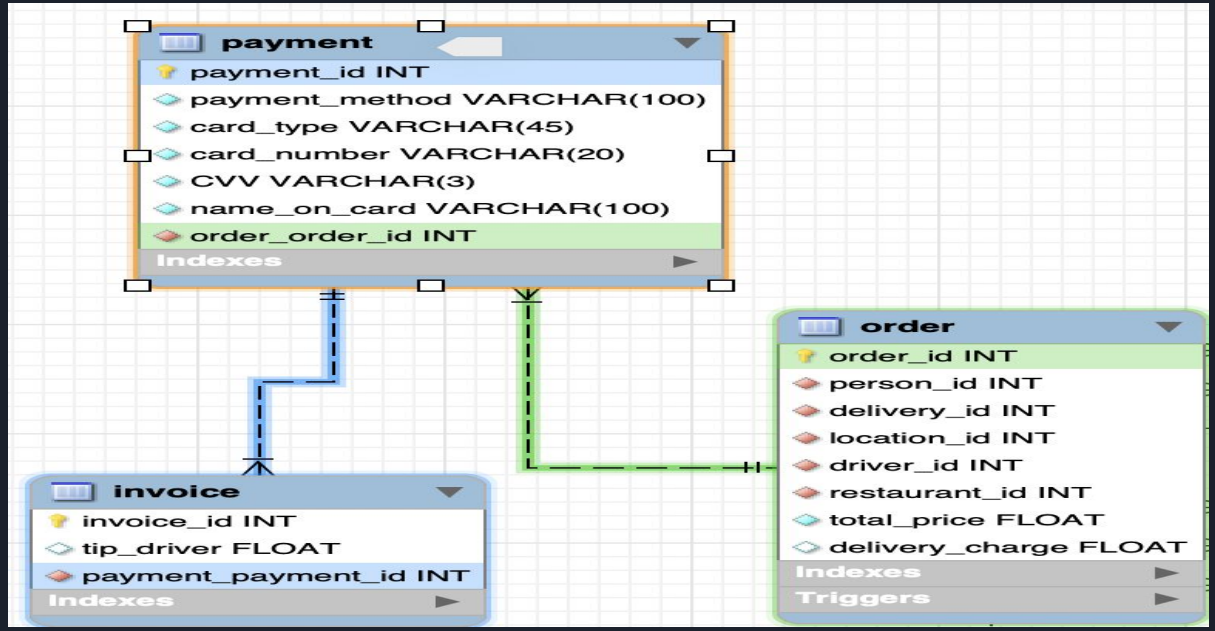
Relationship between menu table and other tables in the database



Relationship between ordered_items table and other tables in the database



Relationship between invoice table and other tables in the database



Stored Procedures

- A stored procedure is a prepared SQL code that you can save, so the code can be reused over and over again.
- So if you have an SQL query that you write over and over again, save it as a stored procedure, and then just call it to execute it.
- You can also pass parameters to a stored procedure, so that the stored procedure can act based on the parameter value(s) that is passed.
- **Syntax:**
  - Create Procedure procedure_name
  - AS Sql Statement
  - Go;
  - Exec procedure_name

Stored Procedure used in the Project

```
USE campus_eats_fall2020;

DROP PROCEDURE IF EXISTS feature_ratings;

DELIMITER //


CREATE PROCEDURE feature_ratings (IN restaurant_id INT)

BEGIN

 SELECT restaurant_id,MAX(food_quality) AS max_food_quality,MAX(service) AS
max_service,MAX(atmosphere) AS max_atmoshphere,MAX(affordability) AS max_affordability,

 MIN(food_quality) AS min_food_quality,MIN(service) AS min_service,MIN(atmosphere) AS
min_atmoshphere,MIN(affordability) AS min_affordability,

 AVG(food_quality) AS avg_food_quality,AVG(service) AS avg_service,AVG(atmosphere) AS
avg_atmoshphere,AVG(affordability) AS avg_affordability

 FROM restaurant_rating

 WHERE restaurant_id = restaurant_id;

END//

CALL feature_ratings(2);
```

Views

- In SQL, a view is a virtual table based on the result-set of an SQL statement.
- A view contains rows and columns, just like a real table. The fields in a view are fields from one or more real tables in the database.
- You can add SQL statements and functions to a view and present the data as if the data were coming from one single table.
- A view is created with the CREATE VIEW statement.
- **Syntax:**
  - CREATE VIEW view_name AS
  - SELECT column1, column2, …
  - FROM table_name
  - WHERE condition;

Views used in the Project

```
DROP VIEW IF EXISTS  Totalprice_each_customer;

CREATE VIEW Totalprice each customer AS
SELECT p.person id,ROUND(SUM(o.total price),2) AS
total price FROM campus eats fall2020.order o
INNER JOIN  campus eats_fall2020.person p    ON
o.person id=p.person id
INNER JOIN campus eats fall2020.delivery d ON
o.delivery id=d.delivery id
WHERE  DATE_FORMAT(d.delivery time, '%Y/%m/%d') BETWEEN
'1970/01/11' AND '2017/01/01' GROUP BY o.person_id ;

SELECT * FROM
campus_eats_fall2020.Totalprice_each_customer;
```

Indexes

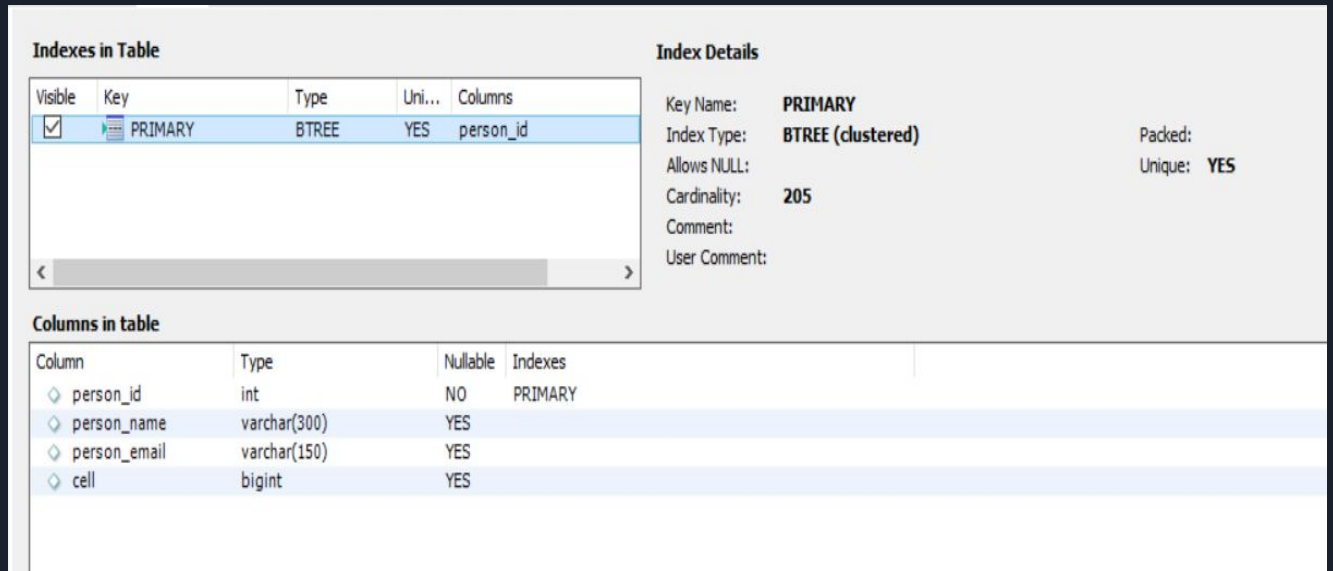
- A SQL index is used to retrieve data from a database very fast. Indexing a table or view is, without a doubt, one of the best ways to improve the performance of queries and applications.
- A SQL index is a quick lookup table for finding records users need to search frequently. An index is small, fast, and optimized for quick lookups. It is very useful for connecting the relational tables and searching large tables.
- Without indexing it will take average search time of (n+1)/2 where n is the number of rows.Using index it uses binary search which is of log2(n)-1 time.
- Syntax:
  - CREATE INDEX index_name

    ON table_name (column1, column2, ...);

Index used in the Project

Example:

- **Index used in Person table for Person_id column ,Clustered index(BTREE since we use InnoDB).**



Indexes in Table

Visible	Key	Type	Uni...	Columns
☑	PRIMARY	BTREE	YES	person_id

Index Details

Key Name: **PRIMARY**
Index Type: **BTREE (clustered)**
Allows NULL:
Cardinality: **205**
Comment:
User Comment:
Packed:
Unique: **YES**

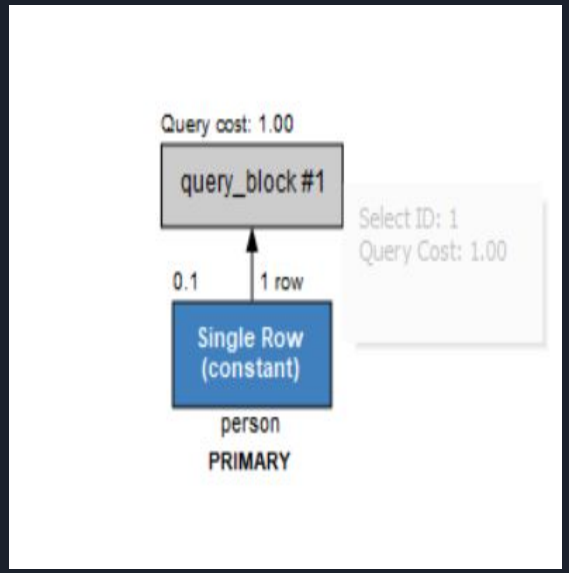
Columns in table

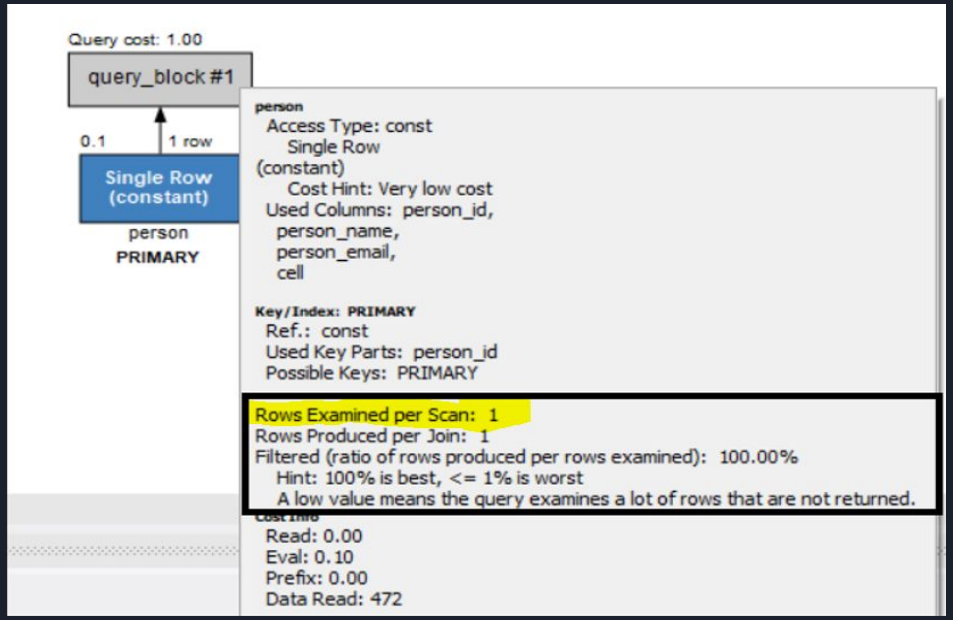
Column	Type	Nullable	Indexes
person_id	int	NO	PRIMARY
person_name	varchar(300)	YES	
person_email	varchar(150)	YES	
cell	bigint	YES	

Query performance visualization using index

- Accessing Person_id from person table clustered index
- Since it is clustered number of rows examined is 1 otherwise it will scan each and every row,and it has low cost.

```
SELECT * FROM campus_eats_fall2020.person WHERE person_id=1;
```

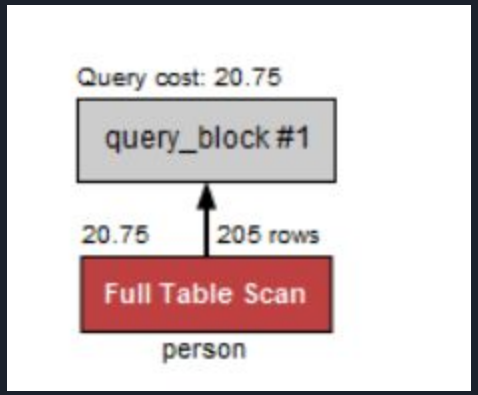


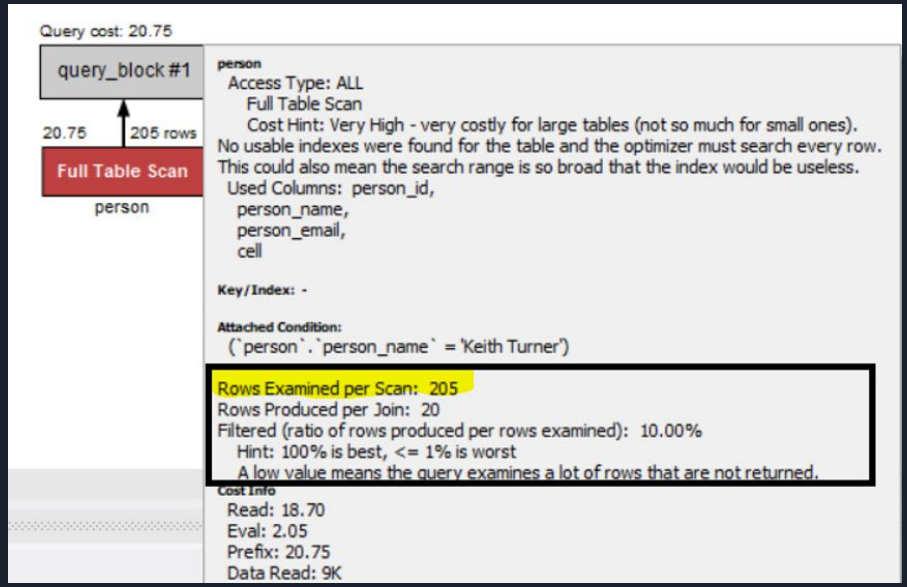


Query performance visualization without using index

- Accessing the same row using Person_name in person table which is not clustered.
- Since it is not clustered number of rows examined is total number of rows in the table,and it has very high cost.

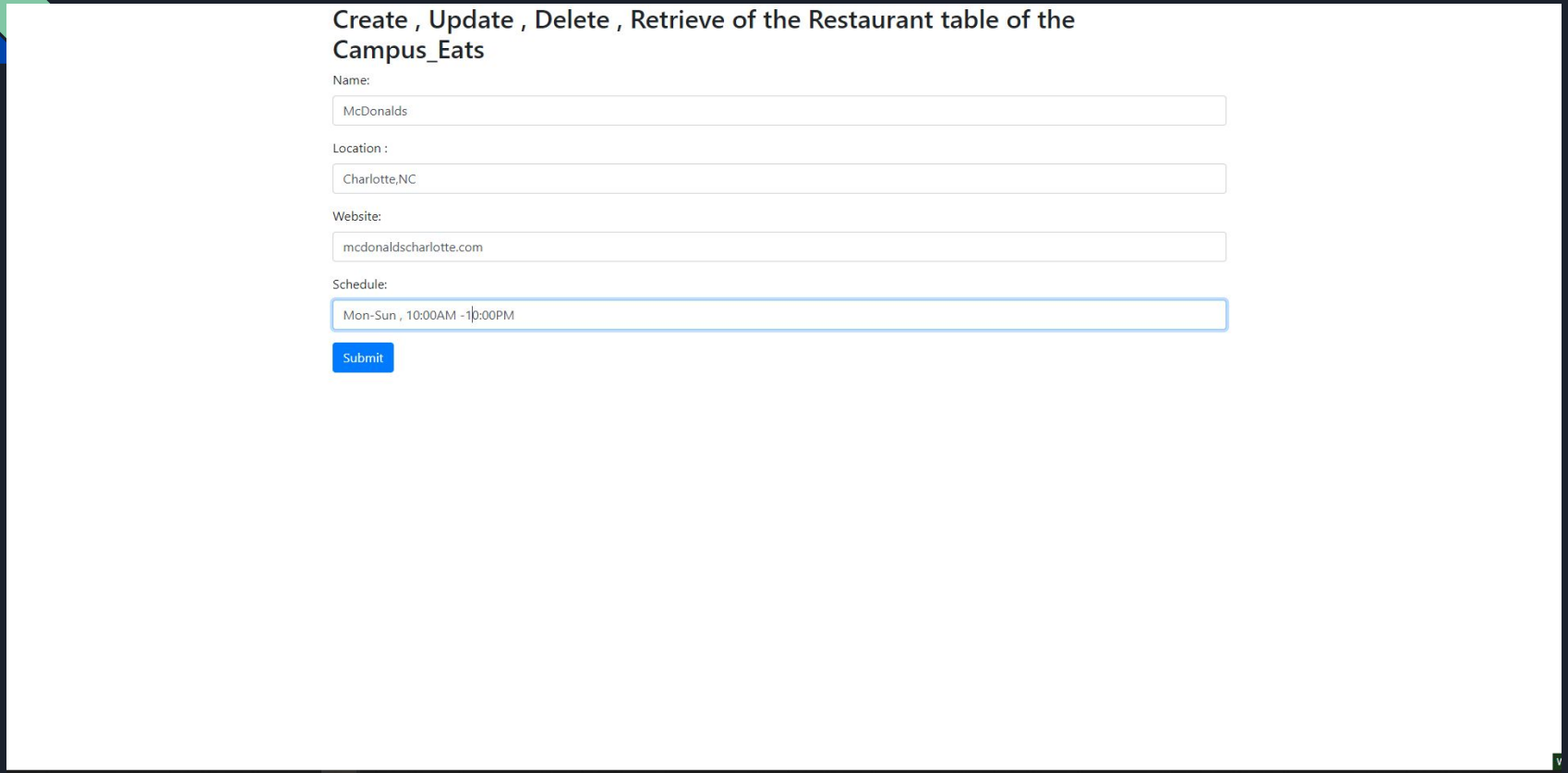
```
SELECT * FROM campus_eats_fall2020.person WHERE person_name='Keith Turner';
```



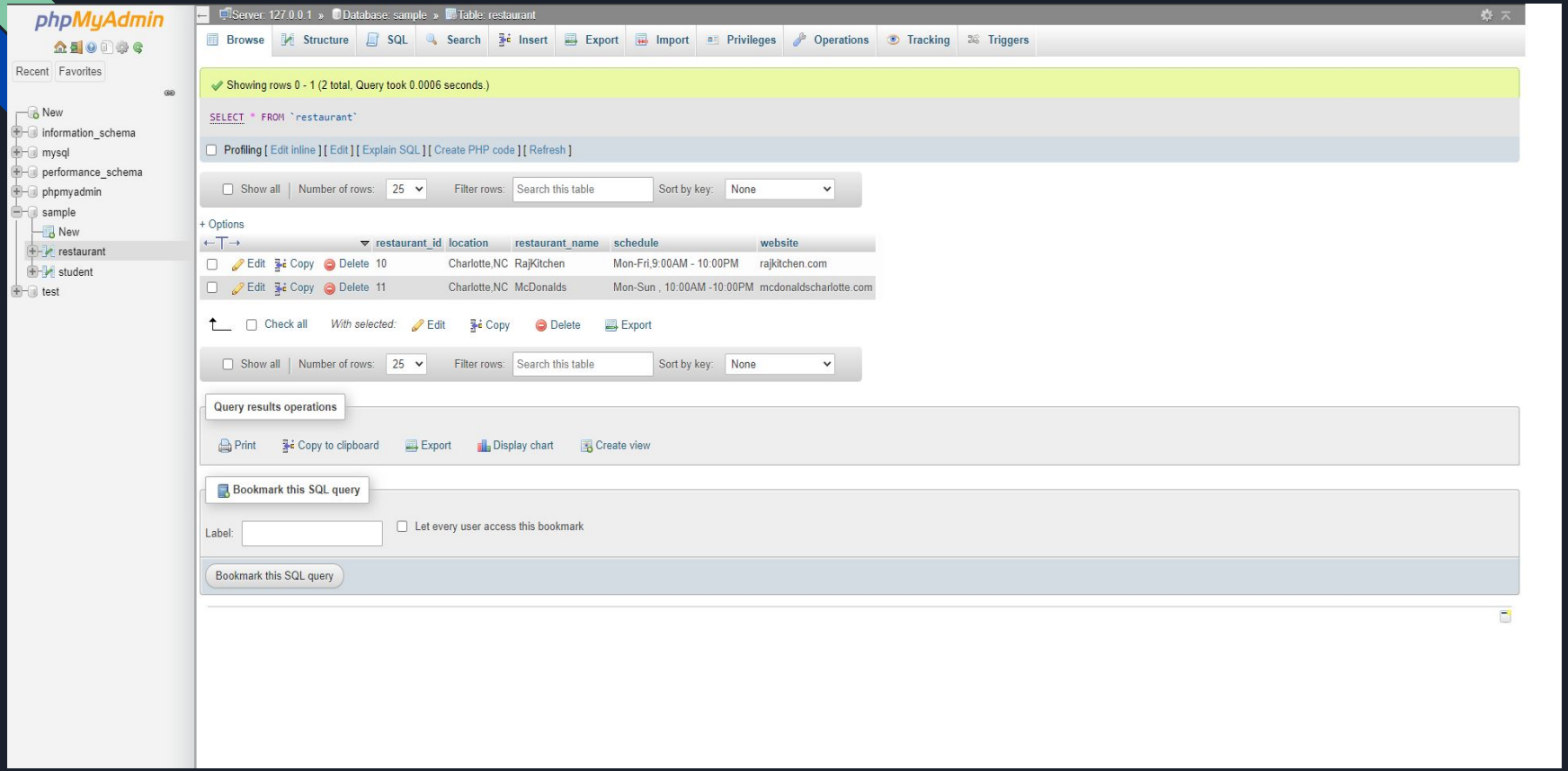


UI For Create , Update , Delete , Retrieve

Insert Page:


Create , Update , Delete , Retrieve of the Restaurant table of the Campus_Eats
Name:
McDonalds
Location :
Charlotte,NC
Website:
mcdonaldscharlotte.com
Schedule:
Mon-Sun , 10:00AM -10:00PM
Submit

Database Dashboard:



Thank You