SVNIT, SURAT

DATABASE MANAGEMENT SYSTEM

(MINI PROJECT)

SEMESTER III (2023-2024)

E-COMMERCE WEBSITE DBMS

SUBMITTED BY-

ADITYA RAI – U22CS041

PARAM PATHAK – U22CS023

SANGAM BIRLA – U22CS040

UJJAWAL TRIPATHI – U22CS037

1. SQL Scripting-

Table creation queries –

```
-- Cart Table
CREATE TABLE Cart_Ecommerce (
    Cart_id VARCHAR(7) NOT NULL,
    PRIMARY KEY (Cart_id)
);

-- Customers Table
CREATE TABLE Customers_Ecommerce (
    Customer_id VARCHAR(10) NOT NULL,
    Customer_Name VARCHAR(25) NOT NULL,
    Address VARCHAR(20) NOT NULL,
    Pincode INT(6) NOT NULL,
    Phone_number BIGINT(10) NOT NULL,
    PRIMARY KEY (Customer_id),
    Cart_id VARCHAR(7) NOT NULL,
    FOREIGN KEY (Cart_id) REFERENCES Cart_Ecommerce (Cart_id)
);

-- Seller Table
CREATE TABLE Seller_Ecommerce (
    Seller_id VARCHAR(6) NOT NULL,
    Name VARCHAR(20) NOT NULL,
    Email_id VARCHAR(50) NOT NULL,
    Phone_number BIGINT(10) NOT NULL,
    Address VARCHAR(10) NOT NULL,
    PRIMARY KEY (Seller_id),
    CONSTRAINT CHECK_Email_id CHECK (Email_id LIKE '%@')
);
```

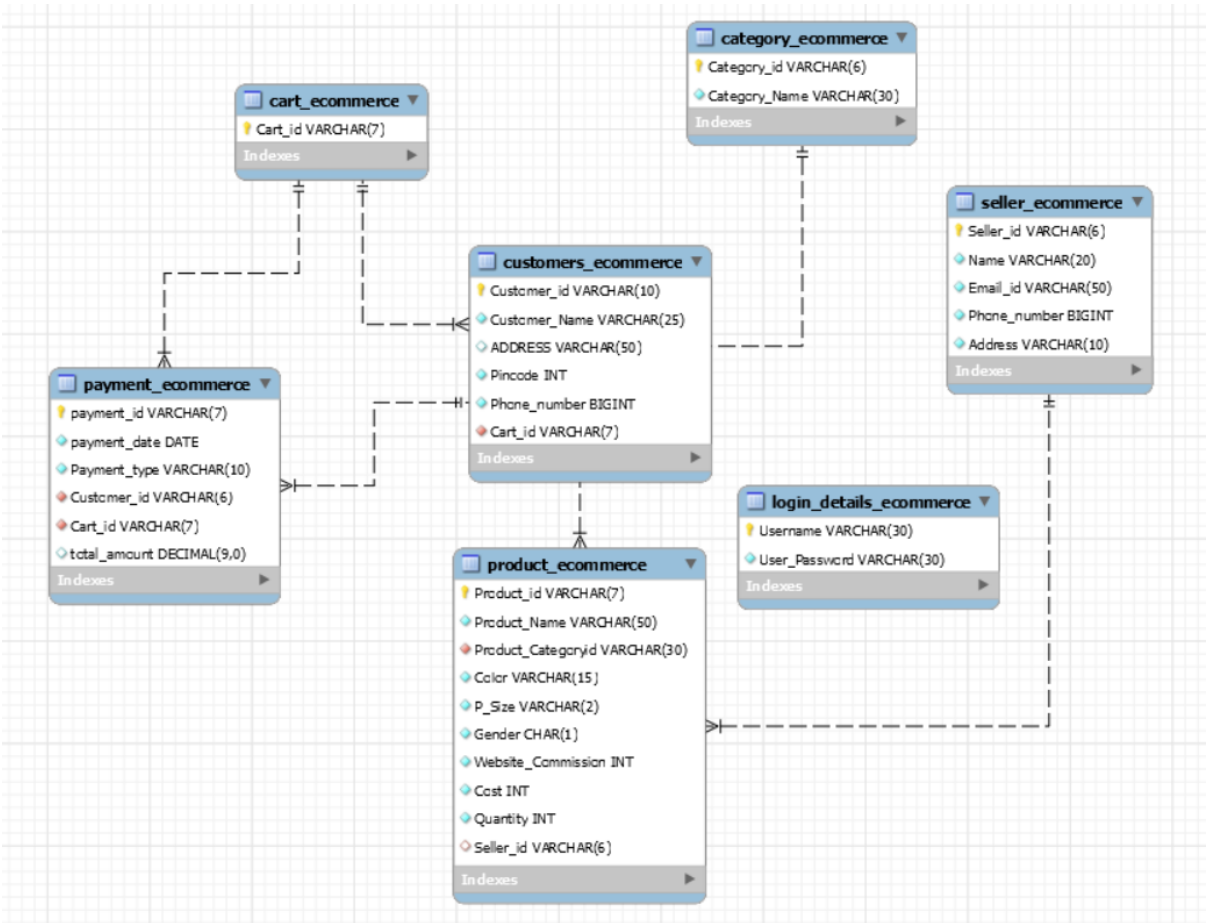
```
-- Category Table
CREATE TABLE Category_ECommerce (
    Category_id VARCHAR(6) NOT NULL,
    Category_Name VARCHAR(30) NOT NULL,
    PRIMARY KEY (Category_id)
);

-- Product Table
CREATE TABLE Product_Ecommerce (
    Product_id VARCHAR(7) NOT NULL,
    Product_Name VARCHAR(50) NOT NULL,
    Product_Categoryid VARCHAR(30) NOT NULL,
    Color VARCHAR(15) NOT NULL,
    P_Size VARCHAR(2) NOT NULL,
    Gender CHAR(1) NOT NULL,
    Website_Commission INT(2) NOT NULL,
    Cost INT(7) NOT NULL,
    Quantity INT(4) NOT NULL,
    Seller_id VARCHAR(6),
    PRIMARY KEY (Product_id),
    FOREIGN KEY (Seller_id) REFERENCES Seller_Ecommerce (Seller_id),
    FOREIGN KEY (Product_Categoryid) REFERENCES Category_ECommerce (Category_id)
);

-- Login Details Table
CREATE TABLE Login_Details_Ecommerce (
    Username VARCHAR(30) NOT NULL,
    User_Password VARCHAR(30) NOT NULL,
    PRIMARY KEY (Username),
    CONSTRAINT CHECK_Password CHECK (LENGTH(User_Password) >= 8)
);
```

```
-- Payment Table
CREATE TABLE Payment_Ecommerce (
    payment_id VARCHAR(7) NOT NULL,
    payment_date DATE NOT NULL,
    Payment_type VARCHAR(10) NOT NULL,
    Customer_id VARCHAR(6) NOT NULL,
    Cart_id VARCHAR(7) NOT NULL,
    PRIMARY KEY (payment_id),
    FOREIGN KEY (Customer_id) REFERENCES Customers_Ecommerce (Customer_id),
    FOREIGN KEY (Cart_id) REFERENCES Cart_Ecommerce (Cart_id),
    total_amount DECIMAL(9)
);
```

2. ER Diagram-



3. Normalisation-

In the process of designing the database for our e-commerce project, normalization was employed to enhance data integrity, reduce redundancy, and improve overall efficiency. The primary objectives of normalization include organizing data systematically, minimizing data duplication, and establishing logical relationships between tables.

One key normalization step involved the separation of concerns within the **Customers_Ecommerce** and **Cart_Ecommerce** tables. By creating a dedicated **Cart_Ecommerce** table with a unique identifier (**Cart_id**) as the primary key, we achieved a more streamlined and normalized structure. This not only ensures that each cart is uniquely identified but also facilitates a clear relationship between customers and their respective shopping carts.

Additionally, the **Payment_Ecommerce** table underwent normalization to enhance its coherence. The association of a payment with a specific order (identified by **Cart_id**) was introduced to strengthen the logical relationships within the database. This restructuring helps in better tracking and managing payment transactions associated with specific customer orders.

We can also Create a separate table for **Product_Category** with **Category_id** as the primary key and **Category_Name.** We can Modify **Product_Ecommerce** to reference **Product_Category** using a foreign key (**Category_id**) instead of storing the category name directly.

If a customer can have multiple addresses, we may also consider creating a separate **Addresses** table with **Address_id** as the primary key and linking it to the **Customers_Ecommerce** table.

If a cart can contain multiple products,we may also consider creating a **Cart_Items** table with **Cart_Item_id** as the primary key, linking it to the **Cart_Ecommerce** table, and referencing the **Product_Ecommerce** table.

4. Table Snaps-

```
mysql> desc cart_ecommerce;
+---------+-----------+------+-----+---------+-------+
| Field   | Type      | Null | Key | Default | Extra |
+---------+-----------+------+-----+---------+-------+
| Cart_id | varchar(7) | NO   | PRI | NULL    |       |
+---------+-----------+------+-----+---------+-------+
1 row in set (0.05 sec)

mysql> desc category_ecommerce;
+---------------+------------+------+-----+---------+-------+
| Field         | Type       | Null | Key | Default | Extra |
+---------------+------------+------+-----+---------+-------+
| Category_id   | varchar(6)  | NO   | PRI | NULL    |       |
| Category_Name | varchar(30) | NO   |     | NULL    |       |
+---------------+------------+------+-----+---------+-------+
2 rows in set (2.88 sec)

mysql> desc customers_ecommerce;
+---------------+------------+------+-----+---------+-------+
| Field         | Type       | Null | Key | Default | Extra |
+---------------+------------+------+-----+---------+-------+
| Customer_id   | varchar(10) | NO   | PRI | NULL    |       |
| Customer_Name | varchar(25) | NO   |     | NULL    |       |
| ADDRESS       | varchar(50) | YES  |     | NULL    |       |
| Pincode       | int         | NO   |     | NULL    |       |
| Phone_number  | bigint      | NO   |     | NULL    |       |
| Cart_id       | varchar(7)  | NO   | MUL | NULL    |       |
+---------------+------------+------+-----+---------+-------+
6 rows in set (0.62 sec)

mysql> desc login_details_ecommerce;
+---------------+------------+------+-----+---------+-------+
| Field         | Type       | Null | Key | Default | Extra |
+---------------+------------+------+-----+---------+-------+
| Username      | varchar(30) | NO   | PRI | NULL    |       |
| User_Password | varchar(30) | NO   |     | NULL    |       |
+---------------+------------+------+-----+---------+-------+
2 rows in set (0.03 sec)
```

```
mysql> desc seller_ecommerce;
+--------------+------------+------+-----+---------+-------+
| Field        | Type       | Null | Key | Default | Extra |
+--------------+------------+------+-----+---------+-------+
| Seller_id    | varchar(6)  | NO   | PRI | NULL    |       |
| Name         | varchar(20) | NO   |     | NULL    |       |
| Email_id     | varchar(50) | NO   |     | NULL    |       |
| Phone_number | bigint      | NO   |     | NULL    |       |
| Address      | varchar(10) | NO   |     | NULL    |       |
+--------------+------------+------+-----+---------+-------+
5 rows in set (0.02 sec)
```

```
mysql> desc payment_ecommerce;
+--------------+--------------+------+-----+---------+-------+
| Field        | Type         | Null | Key | Default | Extra |
+--------------+--------------+------+-----+---------+-------+
| payment_id   | varchar(7)   | NO   | PRI | NULL    |       |
| payment_date | date         | NO   |     | NULL    |       |
| Payment_type | varchar(10)  | NO   |     | NULL    |       |
| Customer_id  | varchar(6)   | NO   | MUL | NULL    |       |
| Cart_id      | varchar(7)   | NO   | MUL | NULL    |       |
| total_amount | decimal(9,0) | YES  |     | NULL    |       |
+--------------+--------------+------+-----+---------+-------+
6 rows in set (0.01 sec)

mysql> desc product_ecommerce;
+--------------------+-------------+------+-----+---------+-------+
| Field              | Type        | Null | Key | Default | Extra |
+--------------------+-------------+------+-----+---------+-------+
| Product_id         | varchar(7)  | NO   | PRI | NULL    |       |
| Product_Name       | varchar(50) | NO   |     | NULL    |       |
| Product_Categoryid | varchar(30) | NO   | MUL | NULL    |       |
| Color              | varchar(15) | NO   |     | NULL    |       |
| P_Size             | varchar(2)  | NO   |     | NULL    |       |
| Gender             | char(1)     | NO   |     | NULL    |       |
| Website_Commission | int         | NO   |     | NULL    |       |
| Cost               | int         | NO   |     | NULL    |       |
| Quantity           | int         | NO   |     | NULL    |       |
| Seller_id          | varchar(6)  | YES  | MUL | NULL    |       |
+--------------------+-------------+------+-----+---------+-------+
10 rows in set (0.04 sec)
```

5. Project Front-end/User-Interface-

->Demo Run-

-> Firstly, when we run the code, it asks for the username and password and the user will be able to access our database management system only if their username and password are present in the login_details_ecommerce table. After getting access, user will be presented with a menu of what they want to do.

```
Enter username: aditya__ra1
Enter password: adiLovesShopping
Welcome to our e-commerce website database management system!

1. Add a new record.
2. Search for a particular record.
3. Delete a record.
4. Display a table.
5. Update a record.
6. Exit.


Enter your choice: 4
Select a table to be displayed
1. Cart_Ecommerce
2. Customers_Ecommerce
3. Seller_Ecommerce
4. Category_ECommerce
5. Product_Ecommerce
6. Login_Details_Ecommerce
7. Payment_Ecommerce
```

```
Choose a number: 3

Seller_Ecommerce Table is empty.
1. Add a new record.
2. Search for a particular record.
3. Delete a record.
4. Display a table.
5. Update a record.
6. Exit.


Enter your choice: 1
Select a table to add a record into:
1. Cart_Ecommerce
2. Customers_Ecommerce
3. Seller_Ecommerce
4. Category_ECommerce
5. Product_Ecommerce
6. Login_Details_Ecommerce
7. Payment_Ecommerce
```

->Here, user 1st chose to display the table seller_ecommerce but it was empty. In the next few steps, user inserts data for 2 sellers into the seller_ecommerce table and displays it later.

```
Choose a number: 3
Enter seller id: SEL001
Enter seller's name: Yashwant Singh
Enter Seller email_id: seller001@gmail.com
Enter phone number: 9812713183
Enter address: 415, 2nd street, Bhatha, Surat
Data added successfully


1. Add a new record.
2. Search for a particular record.
3. Delete a record.
4. Display a table.
5. Update a record.
6. Exit.


Enter your choice: 1
Select a table to add a record into:
1. Cart_Ecommerce
2. Customers_Ecommerce
3. Seller_Ecommerce
4. Category_ECommerce
5. Product_Ecommerce
6. Login_Details_Ecommerce
7. Payment_Ecommerce
```

```
Choose a number: 3
Enter seller id: SEL002
Enter seller's name: Ishaan Khattar
Enter Seller email_id: seller002@gmail.com
Enter phone number: 9382860131
Enter address: Bholi Apartments, Park Residency, New Delhi
Data added successfully
```

```
1. Add a new record.
2. Search for a particular record.
3. Delete a record.
4. Display a table.
5. Update a record.
6. Exit.
```

->As you can see in the next snapshot, The 2 sets of data were inserted into the table seller_ecommerce and accordingly displayed.

->Next, the user decided to use the search functionality to search for the record of seller id ‘SEL002’ in the seller_ecommerce table.

```
Enter your choice: 4
Select a table to be displayed
1. Cart_Ecommerce
2. Customers_Ecommerce
3. Seller_Ecommerce
4. Category_ECommerce
5. Product_Ecommerce
6. Login_Details_Ecommerce
7. Payment_Ecommerce


Choose a number: 3

Seller_Ecommerce Table:
+-----------+---------------+--------------------+-------------+-----------------------------------------+
| SELLER_ID |      NAME     |       EMAIL_ID     | PHONE_NUMBER |                 ADDRESS                 |
+-----------+---------------+--------------------+-------------+-----------------------------------------+
|   SEL001  | Yashwant Singh | seller001@gmail.com |  9812713183  |         415,2nd street,Bhatha, Surat        |
|   SEL002  | Ishaan Khattar | seller002@gmail.com |  9382860131  | Bholi Apartments, Park Residency, New Delhi |
+-----------+---------------+--------------------+-------------+-----------------------------------------+
1. Add a new record.
2. Search for a particular record.
3. Delete a record.
4. Display a table.
5. Update a record.
6. Exit.
```

```
Enter your choice: 2
Select a table to search from:
1. Cart
2. Category
3. Customers
4. Sellers
5. Products
6. Payment
Which table do you want to search in?: 4
Enter Seller_id to be searched: SEL002

seller_ecommerce Table:
+-----------+---------------+--------------------+-------------+-----------------------------------------+
| SELLER_ID |      NAME     |       EMAIL_ID     | PHONE_NUMBER |                 ADDRESS                 |
+-----------+---------------+--------------------+-------------+-----------------------------------------+
|   SEL002  | Ishaan Khattar | seller002@gmail.com |  9382860131  | Bholi Apartments, Park Residency, New Delhi |
+-----------+---------------+--------------------+-------------+-----------------------------------------+


1. Add a new record.
2. Search for a particular record.
3. Delete a record.
4. Display a table.
5. Update a record.
6. Exit.
```

->Next, the user searches for a non-existing seller id ‘182937’ in the seller_ecommerce table and the system tells the user that such an id doesn’t exist in the table.

```
Enter your choice: 2
Select a table to search from:
1. Cart
2. Category
3. Customers
4. Sellers
5. Products
6. Payment
Which table do you want to search in?: 4
Enter Seller_id to be searched: 182937

Seller_id 182937 doesn't exist in the table seller_ecommerce.



1. Add a new record.
2. Search for a particular record.
3. Delete a record.
4. Display a table.
5. Update a record.
6. Exit.
```

->Next, the user tries out the update functionality and updates the address of seller whose seller id is ‘SEL001’ and the change is reflected in the table as well as we can see that the address was changed for that seller when the user displays seller_ecommerce table.

```
Enter your choice: 5
Select the table which you want to update:
1. Cart_Ecommerce
2. Customers_Ecommerce
3. Seller_Ecommerce
4. Category_ECommerce
5. Product_Ecommerce
6. Login_Details_Ecommerce
7. Payment_Ecommerce



Select the table which you want to update: 3
Enter seller id whose record you want to update: SEL001
Select attribute to update:
1. Seller_id
2. Name
3. Email_id
4. Phone_number
5. Address
Enter your choice: 5
Enter new Address: South Pole, Near RR Mall, Dumas Road, Surat
Seller with Seller_id SEL001 updated successfully.
1. Add a new record.
2. Search for a particular record.
3. Delete a record.
4. Display a table.
5. Update a record.
6. Exit.
```

```
Enter your choice: 4
Select a table to be displayed
1. Cart_Ecommerce
2. Customers_Ecommerce
3. Seller_Ecommerce
4. Category_ECommerce
5. Product_Ecommerce
6. Login_Details_Ecommerce
7. Payment_Ecommerce


Choose a number: 3

Seller_Ecommerce Table:
+-----------+---------------+--------------------+-------------+-----------------------------------------+
| SELLER_ID |      NAME     |       EMAIL_ID     | PHONE_NUMBER |                 ADDRESS                 |
+-----------+---------------+--------------------+-------------+-----------------------------------------+
|   SEL001  | Yashwant Singh | seller001@gmail.com |  9812713183  | South Pole, Near RR Mall, Dumas Road, Surat |
|   SEL002  | Ishaan Khattar | seller002@gmail.com |  9382860131  | Bholi Apartments, Park Residency, New Delhi |
+-----------+---------------+--------------------+-------------+-----------------------------------------+
```

->At last, the user decides to exit the program as he chooses 6 and the program ends.

```
1. Add a new record.
2. Search for a particular record.
3. Delete a record.
4. Display a table.
5. Update a record.
6. Exit.


Enter your choice: 6
```

```
Process finished with exit code 0
```