DBMS Project Deadline3

REPORT

SQL Tables and Data population with Constraints

Siddharth Gupta 2021355

Tony Thomas 2021360

In this deadline, we created tables for our database model. We defined the data type of each attribute in all the tables to make the schemas then we generated and filled in data into the tables.
The data was inserted using an online data generator.

This was done in accordance with the previously made ER model and the relational model.

The DDL integrity constraints were integrated formally into the prepared dataset while generating data.

Different types on DDL Constraints in SQL:

1. **NOT NULL**
   Whenever a table's column is declared as NOT NULL, then the value for that column cannot be empty for any row in the table.

2. **UNIQUE**
   Duplicate values are not allowed in the columns to which the UNIQUE constraint is applied. This can be applied to one or more than one column of a table

3. **PRIMARY KEY**
   PRIMARY KEY Constraint is a combination of NOT NULL and Unique constraints. The column to which we have applied the primary constraint will always contain a unique value and will not allow null values.

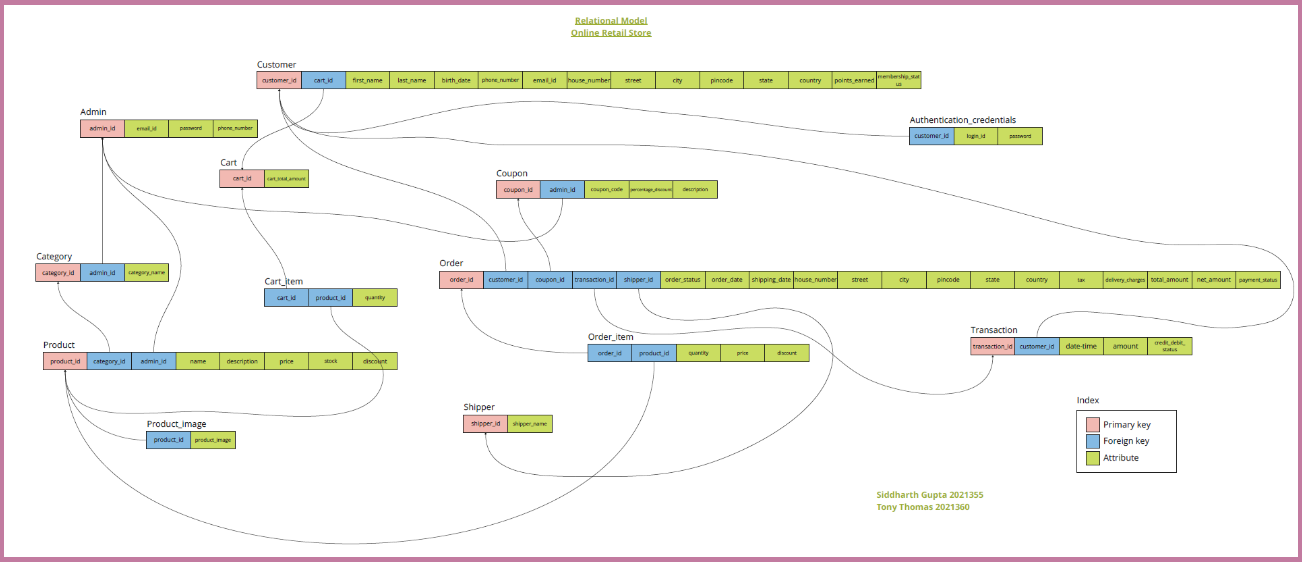
4. **FOREIGN KEY**
   A foreign key is used for referential integrity. When we have two tables, and one table takes reference from another table, i.e., the same column is present in both the tables and that column acts as a primary key in one table. That particular column will act as a foreign key in another table, so the values need to be the same for both.

5. **CHECK**
   Whenever a check constraint is applied to the table's column, the value will be checked first for certain conditions before inserting the value into that column when a user wants to insert a value. Eg- age>18 will not allow negative or less than 18 values.

6. **DEFAULT**
   Whenever a default constraint is applied to the table's column, and the user has not specified the value to be inserted in it, then the default value which was specified while applying the default constraint will be inserted.

7. **CREATE INDEX**
   CREATE INDEX constraint is used to create an index on the table. Indexes are not visible to the user, but they help the user to speed up the searching speed or retrieval of data from the database.

The following show the relational diagram that was used to make the schemas :


Relational Model
Online Retail Store
Customer
customer_id | cart_id | first_name | last_name | birth_date | phone_number | email_id | house_number | street | city | pincode | state | country | points_earned | membership_status
Admin
admin_id | email_id | password | phone_number
Authentication_credentials
customer_id | login_id | password
Cart
cart_id | cart_total_amount
Coupon
coupon_id | admin_id | coupon_code | percentage_discount | description
Category
category_id | admin_id | category_name
Cart_item
cart_id | product_id | quantity
Order
order_id | customer_id | coupon_id | transaction_id | shipper_id | order_status | order_date | shipping_date | house_number | street | city | pincode | state | country | tax | delivery_charges | total_amount | net_amount | payment_status
Product
product_id | category_id | admin_id | name | description | price | stock | discount
Order_item
order_id | product_id | quantity | price | discount
Transaction
transaction_id | customer_id | date-time | amount | credit_debit_status
Product_image
product_id | product_image
Shipper
shipper_id | shipper_name
Index
Primary key
Foreign key
Attribute
Siddharth Gupta 2021355
Tony Thomas 2021360

Some Constraints explained in our context:

- All the attributes shown in red are the primary keys (for the strong entities) and thus PRIMARY KEY constraint was applied on them, we also ensured that the data for these columns was unique for all rows.
- The attributes shown in blue are the foreign keys, so FOREIGN KEY constraint was put on them so that the data in these schemas was consistent with the corresponding data in the actual schema of which these foreign keys were the primary keys.

- Rest of the green attributes were given other constraints like:
  - For coupons, the percentage discount cannot be 0 or less, and not more than 30% as this is the maximum that can be given on the entire cart after other individual discounts on products.
  - Authentication credential values cannot be NULL.
  - The number of Customers, Carts and Authentication credentials are exactly equal as these follow a one-one and total relation.

- Similarly, the number of Orders, Transactions are exactly equal as these also follow a one-one and total relation.
- Orders and shippers follow many-one so shippers were assigned orders in random order with repetition.
- Customer ids were provided to Authentication credentials in a sequential manner with no repetition.

SYNTAX USED

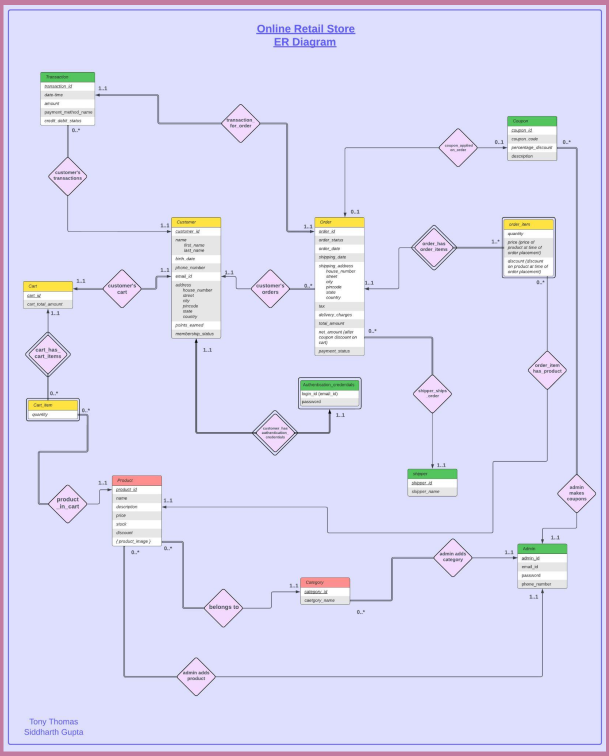
Create Tables:

```
create table Transaction (
        transaction_id varchar(8) NOT NULL,
        customer_id varchar(7),
        date-time date NOT NULL,
        amount int NOT NULL,
        credit_debit_status varchar(6) NOT NULL,
        PRIMARY KEY(transaction_id ),
        FOREIGN KEY (customer_id ) REFERENCES Customer(customer_id )
);
```

Inserting data:

```
insert into Transaction (transaction_id, customer_id, date-time, amount,
credit_debit_status) values ('TRANS001', 'CUST102', '2015-10-23 21:09:41', 7819,
'Debit');
```

Our ER Diagram showing the relations between entities, one-one, many-one, many-many and the strict cardinalities:


Online Retail Store
ER Diagram
Transaction
transaction_id
date-time
amount
payment_method_name
credit_debit_status
transaction_for_order
customer's transactions
Coupon
coupon_id
coupon_code
percentage_discount
description
coupon_applied on_order
Customer
customer_id
name
first_name
last_name
birth_date
phone_number
email_id
address
house_number
street
city
pincode
state
country
points_earned
membership_status
Order
order_id
order_status
order_date
shipping_date
shipping_address
house_number
street
city
pincode
state
country
tax
delivery_charges
total_amount
net_amount (after coupon discount on cart)
payment_status
order_has order_items
order_item
quantity
price (price of product at time of order placement)
discount (discount on product at time of order placement)
Cart
cart_id
cart_total_amount
customer's cart
customer's orders
cart_has_ cart_items
Cart_item
quantity
Authentication_credentials
login_id (email_id)
password
customer_has authentication_ credentials
shipper_ships _order
order_item has_product
shipper
shipper_id
shipper_name
admin makes coupons
product _in_cart
Product
product_id
name
description
price
stock
discount
{ product_image }
admin adds category
Admin
admin_id
email_id
password
phone_number
belongs to
Category
category_id
caetgory_name
admin adds product
1..1
0..*
0..1
1..*
Tony Thomas
Siddharth Gupta