SQL ASSIGNMENT 1

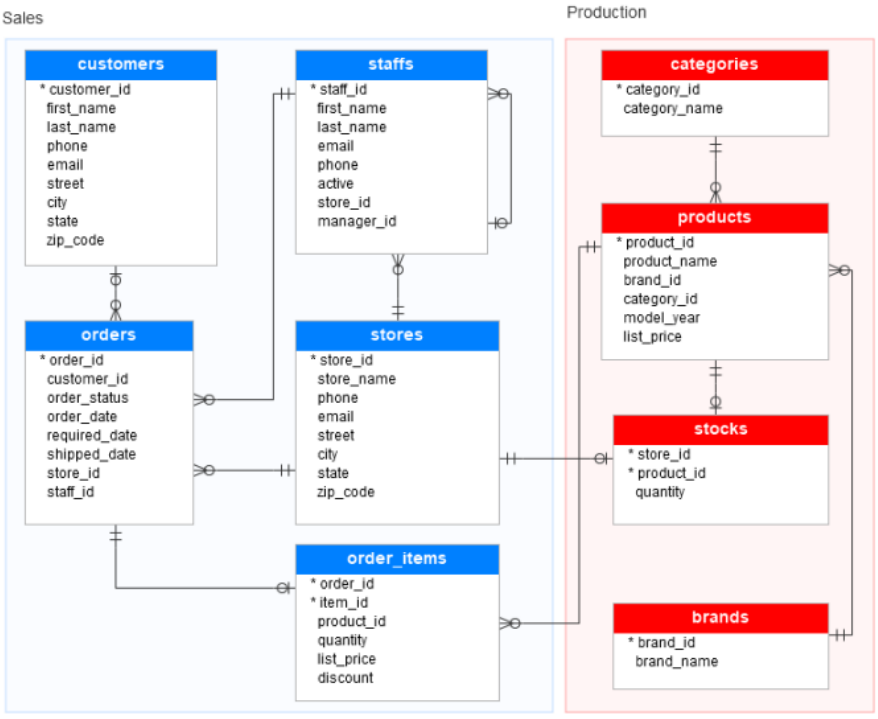
Instructions :

- ***Reference: Lectures In Snowflake & SQL folder (AWA - APP+WEBSITE)***
- ***Due Date : 01-Sept-2023 11:59 PM(Midnight)***
- ***Late submissions will not be evaluated***
- ***Its mandatory to do all questions***
- ***Use SNOWFLAKE for the task submission while for practice one can execute in MySQL Workbench too.***
- ***Proper comments should be given for the code explanation wherever required.***
- ***Proper snippets should be attached of the output(mandatory) and write the code too.***
- ***Don't do plagiarism***
- ***Kindly don't USE JOINS or WINDOWS functions in any of the problems.***
- ***Kindly upload the assignment by uploading it in the below GOOGLE DRIVE FOLDER as per the mentioned format(only pdf) as fullname_assignment_name_yyyy_mm_dd.pdf(anandjha_sql_assignment1_2023_08_29.pdf) :*** https://drive.google.com/drive/folders/1imUPPceiPICh5kjH5ZOmCUfcXAgurQqK?usp=sharing

Task Information :

Let us consider a **BikeStore** company which sells bikes across the country in multiple states along with the service centres spread across the states.

Consider the below **diagram** :



SQL ASSIGNMENT 1

As you can see from the diagram**,** the **BikeStores** sample database has two **schemas : sales and production,** and these schemas have **nine tables** (highlighted in blue and red color in their respective schemas) while ***** indicates **PRIMARY KEY** in each **table**

Database Tables :

sales.stores

The sales.stores table includes the store's information. Each store has a store name, contact information such as phone and email, and an address including street, city, state, and zip code.

sales.staffs

The sales.staffs table stores the essential information of staffs including first name, last name. It also contains the communication information such as email and phone.

A staff works at a store specified by the value in the **store_id** column. A store can have one or more staffs.

A staff reports to a store manager specified by the value in the **manager_id** column. If the value in the **manager_id** is null, then the staff is the **top manager**.

If a staff **no** longer **works** for **any stores**, the value in the active column is set to **zero**.

production.categories

The production.categories table stores the bike's categories such as children bicycles, comfort bicycles, and electric bikes.

production.brands

The production.brands table stores the brand's information of bikes, for example, Electra, Haro, and Heller.

production.products

The production.products table stores the product's information such as name, brand, category, model year, and list price.

SQL ASSIGNMENT 1

Each product belongs to a brand specified by the **brand_id** column. Hence, a brand may have zero or many products.

Each product also belongs a category specified by the **category_id** column. Also, each category may have zero or many products.

sales.customers

The sales.customers table stores customer's information including first name, last name, phone, email, street, city, state and zip code.

sales.orders

The sales.orders table stores the sales order's header information including customer, order status, order date, required date, shipped date.

It also stores the information on where the sales transaction was created (store) and who created it (staff).

Each sales order has a row in the sales_orders table. A sales order has one or many line items stored in the sales.order_items table.

sales.order_items

The sales.order_items table stores the line items of a sales order. Each line item belongs to a sales order specified by the order_id column.

A sales order line item includes product, order quantity, list price, and discount.

production.stocks

The production.stocks table stores the inventory information i.e. the quantity of a particular product in a specific store.

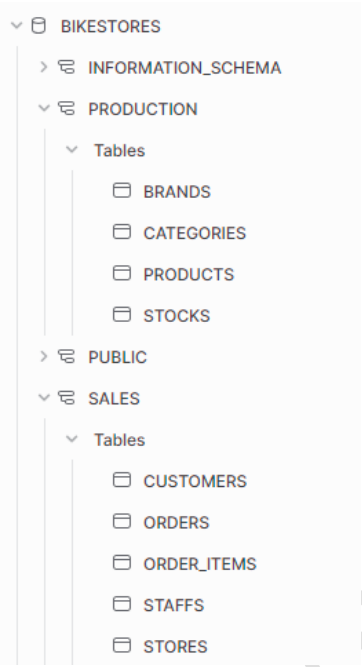
SQL ASSIGNMENT 1

Problem Statement :

1.Design the complete database + schema + tables for the diagram shown above using appropriate data type for every column along with any contraints (checks + PK) mentioned in the task description and load the below data into the requisite tables.

BikeStoreInsertData

Final Output should look like this :



2. Once the table has got created , there is a requirement of FOREIGN KEY implementation coming into picture where one needs to add(ALTER TABLE COMMAND) below foreign key on the table mentioned pointing to another table (READ ABOUT FOREIGN KEY) as :

- *sales.staffs (store_id) -> sales.stores(storeid*
- *sales.staffs (manager_id) -> sales.staffs (staff_id)*
- *production.products (category_id) -> production.categories (category_id)*
- *production.products(brand_id) -> production.brands (brand_id)*
- *sales.orders (customer_id) -> sales.customers (customer_id)*
- *sales.orders(store_id) -> sales.stores (store_id)*
- *sales.orders (staff_id) -> sales.staffs (staff_id)*
- *sales.order_items(order_id) -> sales.orders (order_id)*
- *sales.order_items (product_id) -> production.products (product_id)*
- *production.stocks (store_id) -> sales.stores (store_id)*
- *production.stocks (product_id) -> production.products (product_id)*

SQL ASSIGNMENT 1

3. Does any of the table has missing or NULL value ? If yes which are those and what are their counts ?

4.Does the datasets has any DUPLICATE(identical rows) ? If yes – can you just keep the first record and remove all rest if its possible without using any JOINS or WINDOW function

3.How many unique tables are present in each schema and under each table how many records are we having ? (Write SQL Script for the same – I don't need answer like 3/5/4 etc)

4.How many total serving customer BikeStore has ?

5.How many total orders are there ?

6. Which store has the highest number of sales ?

7. Which month the sales was highest and for which store ?

8. How many orders each customer has placed (give me top 10 customers)

9. Which are the TOP 3 selling product ?

10. Which was the first and last order placed by the customer who has placed maximum number of orders ?

11. For every customer , which is the cheapest product and the costliest product which the customer has bought.

12. Which product has orders more than 200 ?

13.Add a column TOTAL_PRICE with appropriate data type into the sales.order_items

SQL ASSIGNMENT 1

14.Calculate TOTAL_PRICE = quantity * list price and Update the value for all rows in the sales.order_items table.

14.What is the value of the TOTAL_PRICE paid for all the sales.order_items ?

**********THE END**********