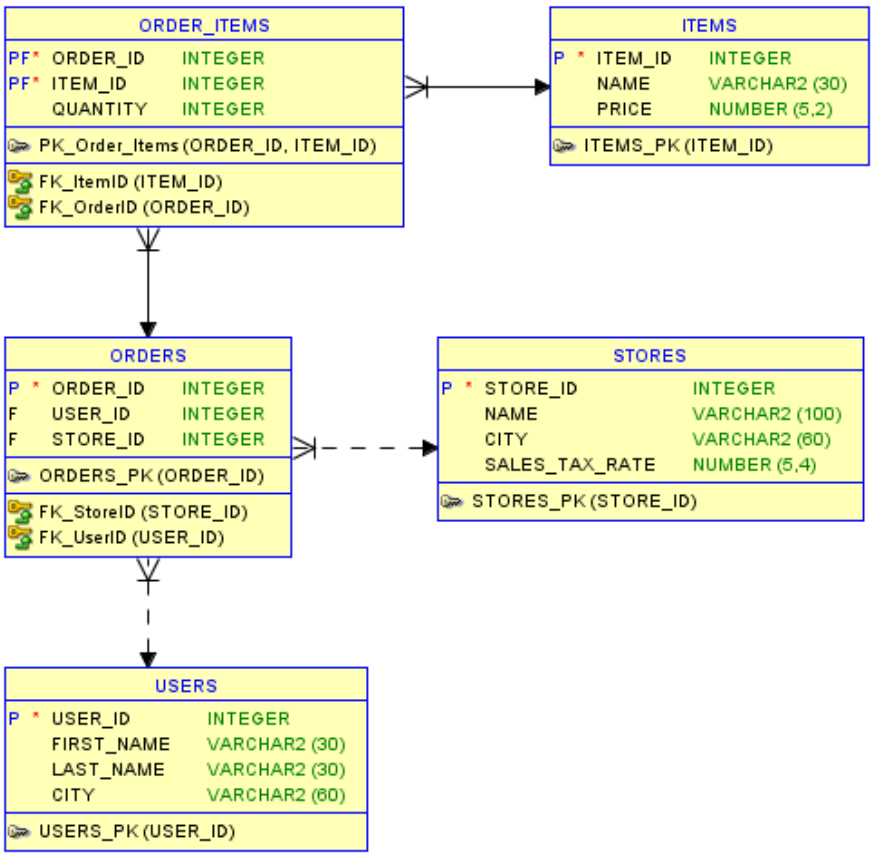

1. Create a query to return all orders made by users with the first name of “Marion”
   Expected Output:

	ORDER_ID	USER_ID	STORE_ID
1	2	3	1
2	3	3	3

2. Create a query to select all users that have not made an order
   Expected Output:

	USER_ID	FIRST_NAME	LAST_NAME	CITY
1	6	Mike	G	New York

3. Create a Query to select the names and prices of all items that have been part of 2 or more separate orders.
   Expected Output:

	NAME	PRICE
1	Hotdog	1.35
2	Fries	1
3	Cheeseburger	2.5
4	Soda	1

4. Create a query to return the Order Id, Item name, Item Price, and Quantity from orders made at stores in the city “New York”. Order by Order Id in ascending order.

Expected Output:

	ORDER_ID	NAME	PRICE	QUANTITY
1	1	Hotdog	1.35	2
2	1	Fries	1	1
3	1	Soda	1	1
4	2	Cheeseburger	2.5	1
5	2	Soda	1	1
6	3	Fries	1	8
7	5	Soda	1	1
8	7	Hotdog	1.35	2

5. Your boss would like you to create a query that calculates the total revenue generated by each item. Revenue for an item can be found as (Item Price * Total Quantity Ordered). Please return the first column as ‘ITEM_NAME’ and the second column as ‘REVENUE’.

Expected Output:

	ITEM_NAME	REVENUE
1	Hotdog	10.8
2	Fries	10
3	Cheeseburger	5
4	Soda	3

6. Create a query with the following output:
    a. Column 1 - Store Name
        i. The name of each store
    b. Column 2 - Order Quantity
        i. The number of times an order has been made in this store
    c. Column 3 - Sales Figure
        i. If the store has been involved in more than 3 orders, mark as ‘High’
        ii. If the store has been involved in less than 3 orders but more than 1 order, mark as ‘Medium’
        iii. If the store has been involved with 1 or less orders, mark as ‘Low’
    d. Should be ordered by the Order Quantity in Descending Order

Expected Output:

	NAME	ORDER_QUANTITY	SALES_FIGURE
1	Bronx Location	3	Medium
2	South Stamford Location	2	Medium
3	Harlem Location	1	Low
4	North Stamford Location	1	Low
5	Financial District	1	Low