**JEREMY PETERSON DUTOYA**

**DR. MOHAMMAD AWWAL**

**PMBA-8317-RHB APPLIED DATA MANAGEMENT FOR BUSINESS USERS**

**LAB 3 – EXECUTE BASIC SQL STATEMENTS**

*In this lab, you will be working on Marcia’s Dry Cleaning Database, using tables: CUSTOMER, INVOICE, and INVOICE\_ITEM*

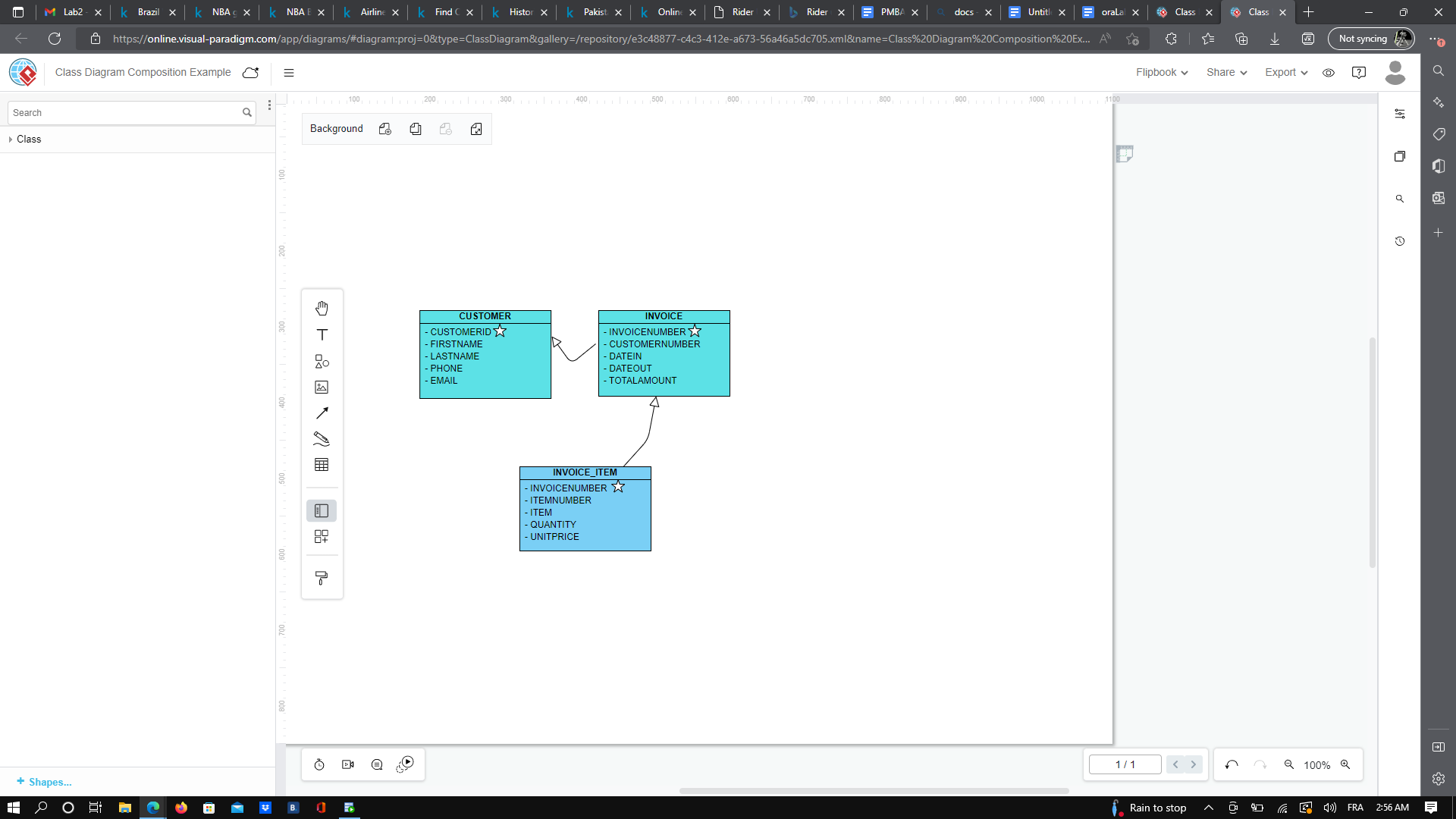
1. *Retrieve data from multiple tables using JOIN*
2. *Use aggregate functions-COUNT, AVG*
3. *USE of UNION, UNION ALL*
4. *Use of concatenation function (||)*
5. *Write SQL statements to display columns of CUSTOMER, INVOICE, INVOICE\_ITEM. Identify joining columns (Foreign key columns on child tables)*

*→* DESC CUSTOMER;

DESC INVOICE;

DESC INVOICE\_ITEM;

With this, I am able to draw the ERD and better analyze the attributes, primary keys, and Foreign keys to do my JOIN queries.



1. *Write a SQL statement to display firstname, lastname, phone from customer table for customers whose firstname starts with ‘B’*

*→* SELECT FirstName, LastName, Phone FROM CUSTOMER

WHERE FirstName LIKE 'B%';

1. *Rewrite SQL from Q2 to concatenate firstname with comma and last name (i.e. Mohammad, Awwal) and display as “Fullname”*

*→* SELECT rtrim(FirstName)|| ', ' ||rtrim(LastName) AS Fullname

FROM CUSTOMER

WHERE FirstName LIKE ‘B%’;

1. *Write a SQL statement to display lastname, firstname, lastname, phone from customer table for customers whose lastname has ‘cat’ (Hint use like ‘%cat%’ and 3rd and 4th number of the phone number is 23. (Hint. Like ‘\_23%’*

→ SELECT LastName, FirstName, Phone

FROM CUSTOMER

WHERE (Lastname LIKE '%cat%') AND (Phone LIKE '\_\_23%');

1. *Write a SQL statement to find maximum, minimum, and average of totalamount from INVOICE table (Hint. Use MAX, MIN, AVG)*

*→* SELECT MAX(TotalAmount), MIN(TotalAmount), AVG(Totalamount)

FROM INVOICE;

1. *Join two tables:*
   1. *Write SQL statement to display customer’s first name, last name, and total amount from customer and invoice tables with totalamount >10*

*→* SELECT C.FirstName, C.LastName, I.TotalAmount

FROM CUSTOMER C, INVOICE I

WHERE (C.CustomerID = I.CustomerNUMBER) AND (I.TotalAmount>10);

OR this works as well because the attribute has a different name in the tables:

→ SELECT FirstName, LastName, TotalAmount

FROM CUSTOMER, INVOICE

WHERE (CustomerID = CustomerNUMBER) AND (TotalAmount>10);

* 1. *Write a SQL statement to display SKU\_description, warehousecity, warehousestate from INVENTORY and WAREHOUSE where warehousecity is not Atlanta and Chicago*

*→* SELECT I.Sku\_Description, W.WarehouseCity, W.WarehouseState

FROM INVENTORY I, WAREHOUSE W

WHERE (I.WarehouseID=W.WarehouseID) AND (W.WarehouseCity NOT IN('Atlanta','Chicago'));

*We can’t do the following query in this case because the column WarehouseID will be ambiguously defined because it has the same name in the Inventory and Warehouse table*

*SELECT Sku\_Description,WarehouseCity, WarehouseState*

*FROM INVENTORY, WAREHOUSE*

*WHERE (WarehouseID=WarehouseID) AND (WarehouseCity NOT IN('Atlanta','Chicago'));*

* 1. *Write a SQL statement to display SKU\_description || ‘is located in ‘|| warehousecity as Itemlocation from INVENTORY and WAREHOUSE and sort by itemlocation*

*→* SELECT rtrim(Sku\_Description)||' is located in '||W.WarehouseCIty AS ItemLocation

FROM INVENTORY I, WAREHOUSE W

WHERE (I.WarehouseID=W.WarehouseID)

ORDER BY ItemLocation;

1. *Join three tables:*
   1. *Write a SQL statement to display firstname, lastname, phone from CUSTOMER table whose item = ‘Dress Shirt’. (Hint: Need to join CUSTOMER, INVOICE, and INVOICE\_ITEM tables). Sort the output by lastname*

→SELECT C.FirstName, C.LastName, C.Phone

FROM CUSTOMER C, INVOICE I, INVOICE\_ITEM IT

WHERE (C.CustomerID = I.CustomerNUMBER)

AND (I.InvoiceNUMBER=IT.InvoiceNUMBER) AND (IT.Item='Dress Shirt')

ORDER BY C.LastName;

1. *Outer joins:*
   1. *Left outer join*
      1. *Write a SQL statement to display lastname and invoicenumber of all customers who has invoice and also who do not have any invoice. (Hint: LEFT outer join on CUSTOMER table)*

*→* SELECT C.LastName, I.InvoiceNumber

FROM CUSTOMER C LEFT OUTER JOIN INVOICE I

ON (C.CustomerID = I.CustomerNUMBER);

We can also do the following query and obtain the same result:

→ SELECT C.LastName, I.InvoiceNumber

FROM CUSTOMER C LEFT JOIN INVOICE I

ON (C.CustomerID = I.CustomerNUMBER);

*We can see that every customers has at least one invoice in this database system, there is no null values in our result. I first thought i did a mistake but then I double checked analyzing the rows of the CUSTOMER and INVOICE Table and it is correct. There is 7 rows in the CUSTOMER table and there is 9 rows in the INVOICE Table. By analyzing the INVOICE Table you can see that every CustomerID appear at least one and the CustomerID #1 and #3 has 2 invoices in the database. So the result we obtained with this 2 queries is accurate.*

* 1. *Right outer join:*
     1. *Rewrite left outer join SQL with right outer join on CUSTOMER*

→ SELECT C.LastName, I.InvoiceNumber

FROM CUSTOMER C RIGHT OUTER JOIN INVOICE I

ON (C.CustomerID = I.CustomerNUMBER);

SQL HISTORY:

| SELECT MAX(TotalAmount), MIN(TotalAmount), AVG(Totalamount) FROM INVOICE; | Petersonduj | 1668327149647 | SQL | 1 | 0.009 |
| --- | --- | --- | --- | --- | --- |
| SELECT MAX(TOTALAMOUNT), MIN(TOTALAMOUNT), AVG(TOTALAMOUNT) FROM INVOICE; | Petersonduj | 1668327019661 | SQL | 1 | 0.012 |
| SELECT LastName, FirstName, Phone FROM CUSTOMER WHERE (Lastname LIKE '%cat%') AND (Phone LIKE '\_\_23%'); | Petersonduj | 1668326879361 | SQL | 1 | 0.009 |
| SELECT FirstName, LastName, Phone FROM CUSTOMER WHERE FirstName LIKE 'B%'; | Petersonduj | 1668326448029 | SQL | 1 | 0.027 |
| DESC INVOICE; DESC INVOICE\_ITEM; | Petersonduj | 1668325408125 | Script | 1 | 3.475 |
| DESC CUSTOMER; DESC INVOICE; DESC INVOICE\_ITEM; | Petersonduj | 1668324540523 | Script | 1 | 6.67 |
| DESC CUSTOMER; | Petersonduj | 1668324503889 | SQL | 2 | 2.435 |

| SELECT rtrim(FirstName)|| ', ' ||rtrim(LastName) AS Fullname FROM CUSTOMER; | Petersonduj | 1668369529269 | SQL | 1 | 0.016 |
| --- | --- | --- | --- | --- | --- |
| SELECT rtrim(FirstName)|| ', ' ||LastName AS Fullname FROM CUSTOMER; | Petersonduj | 1668369502612 | SQL | 1 | 0.014 |

| SELECT FirstName, LastName, TotalAmount FROM CUSTOMER, INVOICE WHERE (CustomerID = CustomerNUMBER) AND (TotalAmount>10); | Petersonduj | 1668327572703 | SQL | 1 | 0.062 |
| --- | --- | --- | --- | --- | --- |
| SELECT C.FirstName, C.LastName, I.TotalAmount FROM CUSTOMER C, INVOICE I WHERE (C.CustomerID = I.CustomerNUMBER) AND (I.TotalAmount>10); | Petersonduj | 1668327481138 | SQL | 1 | 0.068 |

| SELECT I.Sku\_Description, W.WarehouseCity, W.WarehouseState FROM INVENTORY I, WAREHOUSE W WHERE (I.WarehouseID=W.WarehouseID) AND (W.WarehouseCity NOT IN('Atlanta','Chicago')); | Petersonduj | 1668327864217 | SQL | 1 | 0.218 |
| --- | --- | --- | --- | --- | --- |
| desc inventory; desc warehouse; | Petersonduj | 1668327673233 | Script | 1 | 3.364 |

| SELECT rtrim(Sku\_Description)||' is located in '||W.warehouseCIty AS ItemLocation FROM INVENTORY I, WAREHOUSE W WHERE (I.WarehouseID=W.WarehouseID) ORDER BY ItemLocation; | Petersonduj | 1668328363222 | SQL | 1 | 0.065 |
| --- | --- | --- | --- | --- | --- |
| SELECT Sku\_Description ||' is located in '||W.warehouseCIty AS ItemLocation FROM INVENTORY I, WAREHOUSE W WHERE (I.WarehouseID=W.WarehouseID) ORDER BY ItemLocation; | Petersonduj | 1668328326326 | SQL | 1 | 0.066 |

| SELECT C.FirstName, C.LastName, C.Phone FROM CUSTOMER C, INVOICE I, INVOICE\_ITEM IT WHERE (C.CustomerID = I.CustomerNUMBER) AND (I.InvoiceNUMBER=IT.InvoiceNUMBER) AND (IT.Item='Dress Shirt') ORDER BY C.LastName; | Petersonduj | 1668328823269 | SQL | 1 | 0.167 |
| --- | --- | --- | --- | --- | --- |
| SELECT C.FirstName, C.LastName, C.Phone, IT.Item FROM CUSTOMER C, INVOICE I, INVOICE\_ITEM IT WHERE (C.CustomerID = I.CustomerNUMBER) AND (I.InvoiceNUMBER=IT.InvoiceNUMBER) AND (IT.Item='Dress Shirt'); | Petersonduj | 1668328696421 | SQL | 1 | 1.815 |
| SELECT C.FirstName, C.LastName, C.Phone FROM CUSTOMER C, INVOICE I, INVOICE\_ITEM IT WHERE (C.CustomerID = I.CustomerNUMBER) AND (I.InvoiceNUMBER=IT.InvoiceNUMBER) AND (IT.Item='Dress Shirt'); | Petersonduj | 1668328683590 | SQL | 1 | 0.12 |

| SELECT C.LastName, I.InvoiceNumber FROM CUSTOMER C LEFT JOIN INVOICE I ON (C.CustomerID = I.CustomerNUMBER); | Petersonduj | 1668368628719 | SQL | 1 | 0.065 |
| --- | --- | --- | --- | --- | --- |
| SELECT C.LastName, I.InvoiceNumber FROM CUSTOMER C LEFT OUTER JOIN INVOICE I ON (C.CustomerID = I.CustomerNUMBER); | Petersonduj | 1668368615940 | SQL | 1 | 0.075 |
| SELECT \* FROM CUSTOMER C LEFT OUTER JOIN INVOICE I ON (C.CustomerID = I.CustomerNUMBER); | Petersonduj | 1668368606358 | SQL | 1 | 0.077 |