LAB 5 TASKS

Submitted To:

Nazeef Ul Haq

Submitted By:

Harmain Iftikhar 2022-CS-07

1 Query

Perform all JOIN queries on any table using Northwind Schema

1.1 SQL CODE

```
Use northwind
SELECT *
FROM Shippers S1
JOIN
Shippers S2
ON S1.ShipperID = S2.ShipperID

SELECT *
FROM Shippers S1
CROSS JOIN
Shippers S2

SELECT *
FROM Orders
LEFT JOIN
[Order Details]
ON Orders.OrderID = [Order Details].OrderID

SELECT *
FROM Orders
Right JOIN
[Order Details]
ON Orders.OrderID = [Order Details].OrderID

SELECT *
FROM Orders
FULL OUTER JOIN
[Order Details]
ON Orders.OrderID = [Order Details].OrderID
```

1.2 Screenshot

2 Query

Perform self-cross join and see if there is any difference between cross join and self cross join

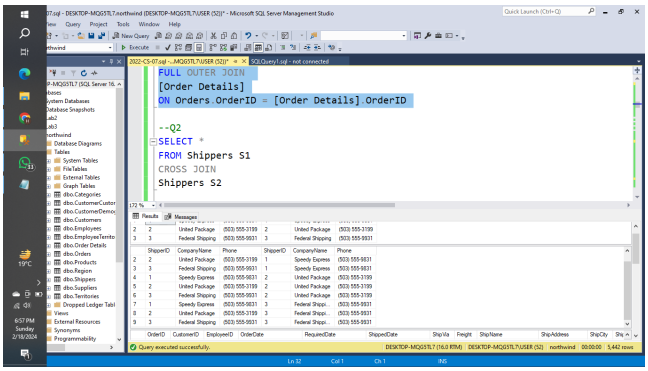

Figure 1: Screenshot of the results

2.1 SQL CODE

```
SELECT *
FROM Shippers S1
CROSS JOIN
Shippers S2

SELECT *
FROM Categories
CROSS JOIN
Shippers;
```

2.2 Screenshot

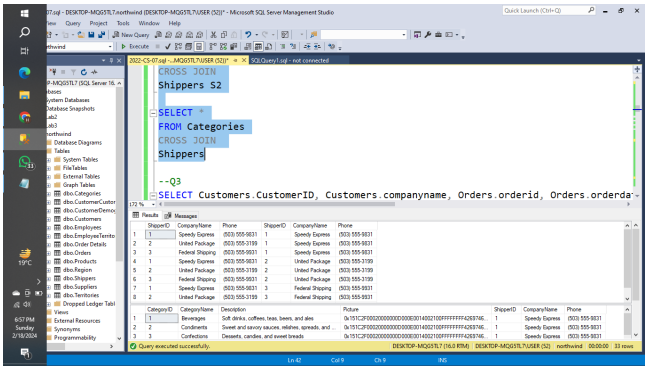

Figure 2: Screenshot of the results

3 Query

SELECT Customers.custid, Customers.companyname, Orders.orderid, Orders.orderdate FROM Sales.Customers AS C INNER JOIN Sales.Orders AS O ON Customers.custid = Orders.custid;

3.1 SQL CODE

```
SELECT Customers.CustomerID, Customers.companyname, Orders.orderid, Orders.orderdate
FROM Customers
INNER JOIN Orders
ON Customers.CustomerID = Orders.CustomerID;
```

3.2 Screenshot

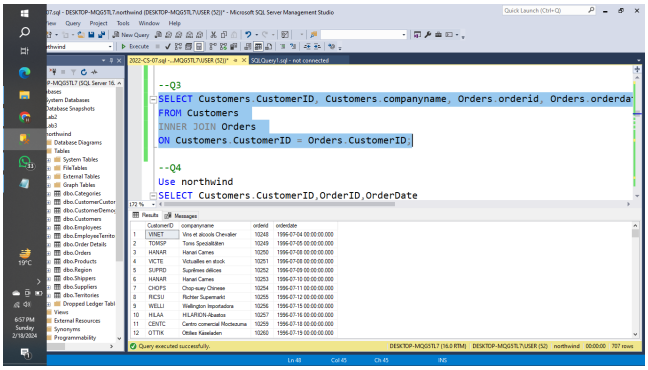

Figure 3: Screenshot of the results

4 Query

Return customers and their orders, including customers who placed no orders (CustomerID, OrderID, OrderDate)

4.1 SQL CODE

```
SELECT Customers.CustomerID,OrderID,OrderDate
FROM Customers LEFT JOIN Orders
ON Customers.CustomerID = Orders.CustomerID;
```

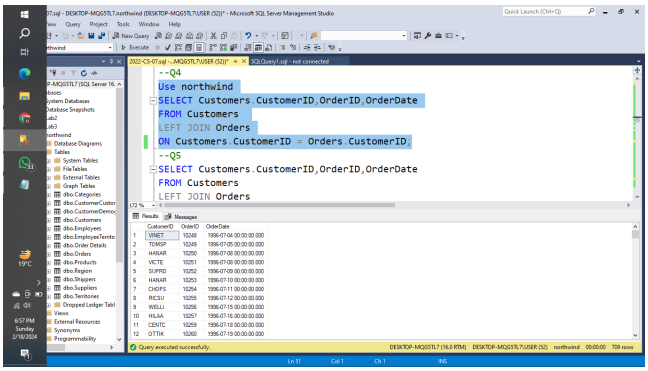

Figure 4: Screenshot of the results

4.2 Screenshot

5 Query

Report only those customer IDs who never placed any order. (CustomerID, OrderID, OrderDate)

5.1 SQL CODE

```
--Q5
SELECT Customers.CustomerID,OrderID,OrderDate
FROM Customers
LEFT JOIN Orders
ON Customers.CustomerID = Orders.CustomerID
WHERE OrderID IS NULL;
```

5.2 Screenshot

6 Query

Report those customers who placed orders on July,1997. (CustomerID, OrderID, OrderDate)

6.1 SQL CODE

```
SELECT Orders.CustomerID,OrderID,OrderDate
FROM Customers JOIN Orders
ON Customers.CustomerID = Orders.CustomerID
WHERE YEAR(OrderDate) = 1997;
```

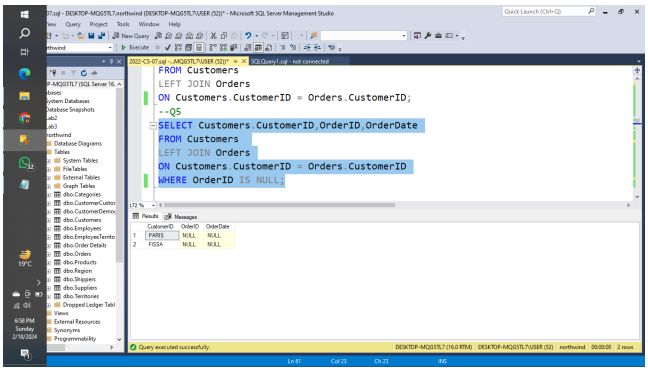

Figure 5: Screenshot of the results

6.2 Screenshot

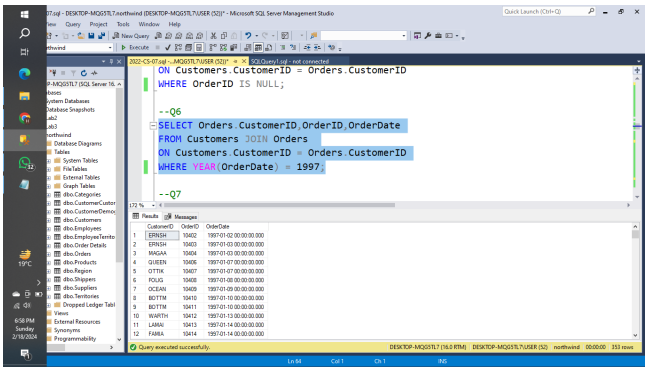

Figure 6: Screenshot of the results

7 Query

Report the total orders of each customer. (customerID, totalorders)

7.1 SQL CODE

```
SELECT Customers.CustomerID,COUNT(OrderDate) AS totalorders
FROM Customers LEFT JOIN Orders
ON Customers.CustomerID = Orders.CustomerID
GROUP BY Customers.CustomerID;
```

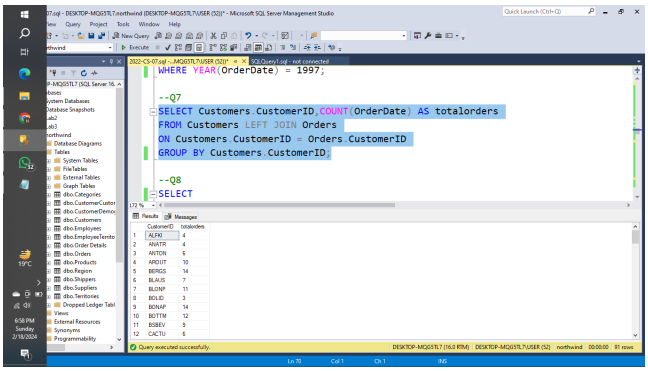

Figure 7: Screenshot of the results

7.2 Screenshot

8 Query

Write a query to generate a five copies of each employee. (EmployeeID, First-Name, LastName)

8.1 SQL CODE

```
SELECT
    E.EmployeeID,
    E.FirstName,
    E.LastName
FROM
    Employees E
CROSS JOIN
    (SELECT 1 AS n UNION ALL SELECT 2 UNION ALL SELECT 3 UNION ALL SELECT 4 UNION ALL SELECT
```

8.2 Screenshot

9 Query

Write a query that returns a row for each employee and day in the range 04-07-1996 through 04-08- 1997. (EmployeeID, Date)

9.1 SQL CODE

```
SELECT EmployeeID,OrderDate
FROM Orders
WHERE OrderDate > '04-07-1996' AND OrderDate < '04-08-1997';
```

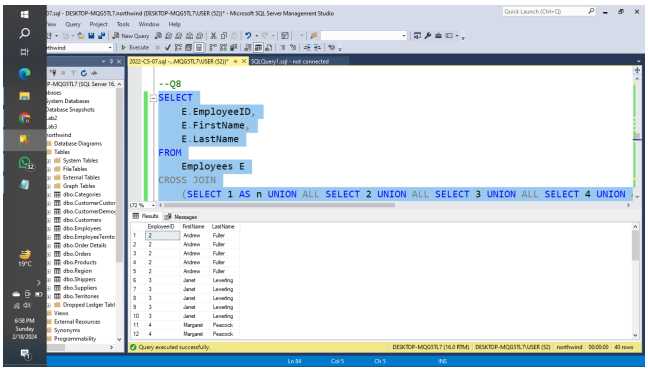

Figure 8: Screenshot of the results

9.2 Screenshot

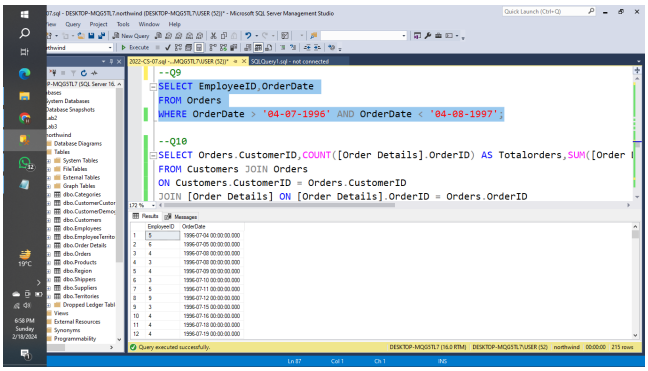

Figure 9: Screenshot of the results

10 Query

Return US customers, and for each customer return the total number of orders and total quantities. (CustomerID, Totalorders, totalquantity)

10.1 SQL CODE

```
SELECT Orders.CustomerID,COUNT([Order Details].OrderID) AS Totalorders,SUM([Order Details].O
FROM Customers JOIN Orders
ON Customers.CustomerID = Orders.CustomerID
JOIN [Order Details] ON [Order Details].OrderID = Orders.OrderID
WHERE Country = 'USA'
```

```
GROUP BY Orders.CustomerID
ORDER BY Orders.CustomerID;
```

10.2 Screenshot

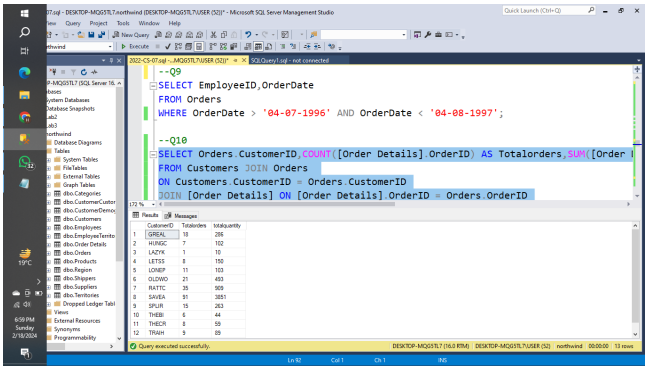

Figure 10: Screenshot of the results

11 Query

Write a query that returns all customers in the output, but matches them with their respective orders only if they were placed on July 04,1997. (CustomerID, CompanyName, OrderID, Orderdate)

11.1 SQL CODE

```
SELECT Orders.CustomerID,Customers.CompanyName,OrderID,OrderDate
FROM Customers
JOIN Orders
ON Customers.CustomerID = Orders.CustomerID
WHERE Orders.OrderDate = '1997-07-04';
```

11.2 Screenshot

12 Query

Are there any employees who are older than their managers?

12.1 SQL CODE

```
SELECT *
FROM Employees E1 JOIN Employees E2
```

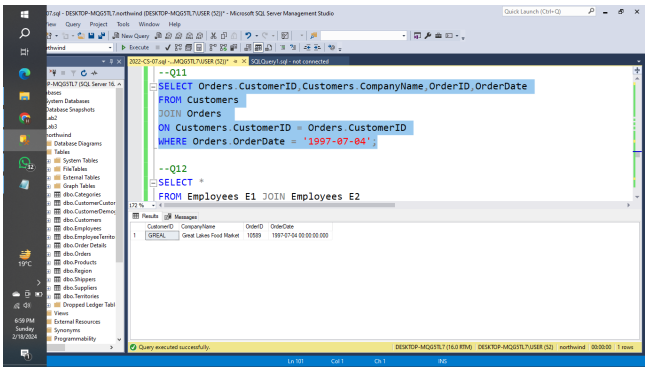

Figure 11: Screenshot of the results

```
ON E1.ReportsTo = E2.EmployeeID
WHERE E2.Title LIKE '%Manager%' AND E1.BirthDate > E2.BirthDate;
```

12.2 Screenshot

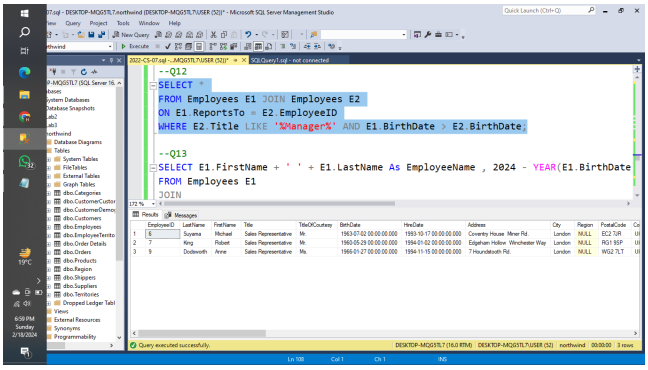

Figure 12: Screenshot of the results

13 Query

List that names of those employees and their ages. (EmployeeName, Age, Manager Age)

13.1 SQL CODE

```
SELECT E1.FirstName + ' ' + E1.LastName As EmployeeName , 2024 - YEAR(E1.BirthDate) AS Age ,
FROM Employees E1
```

```
JOIN
Employees E2 ON E1.ReportsTo = E2.EmployeeID;
```

13.2 Screenshot

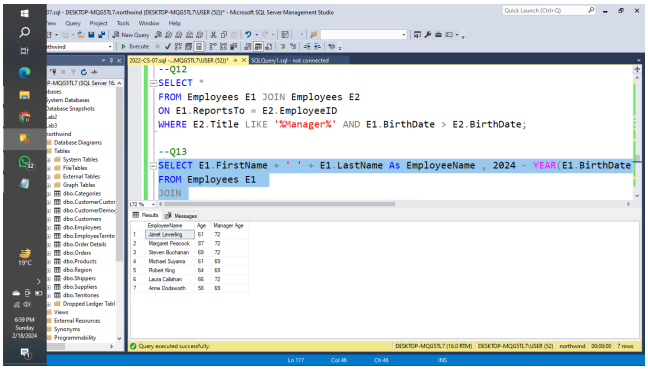

Figure 13: Screenshot of the results

14 Query

List the names of products which were ordered on 8th August 1997. (ProductName, OrderDate)

14.1 SQL CODE

```
SELECT ProductName,Orders.OrderDate
FROM [Orders]
JOIN
[Order Details] ON Orders.OrderID = [Order Details].OrderID
JOIN
Products ON [Order Details].ProductID = Products.ProductID
WHERE OrderDate = '1997-08-08';
```

14.2 Screenshot

15 Query

List the addresses, cities, countries of all orders which were serviced by Anne and were shipped late. (Address, City, Country)

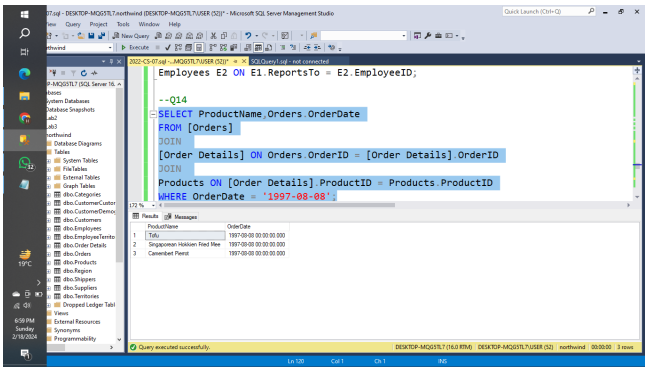

Figure 14: Screenshot of the results

15.1 SQL CODE

```
SELECT Orders.ShipAddress AS Address,Orders.ShipCity AS City,Orders.ShipCountry AS Country
FROM Orders
JOIN
[Order Details] ON Orders.OrderID = [Order Details].OrderID
JOIN
Products ON [Order Details].ProductID = Products.ProductID
JOIN
Suppliers ON Products.SupplierID = Suppliers.SupplierID
WHERE ShippedDate > RequiredDate AND Suppliers.ContactName LIKE '%Anne%'
```

15.2 Screenshot

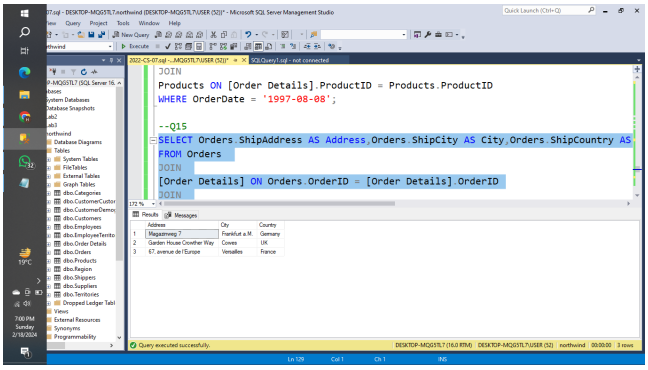

Figure 15: Screenshot of the results

16 Query

List all countries to which beverages have been shipped. (Country)

16.1 SQL CODE

```
SELECT DISTINCT(ShipCountry)
FROM Orders
JOIN
[Order Details] ON Orders.OrderID = [Order Details].OrderID
JOIN
Products ON [Order Details].ProductID = Products.ProductID
JOIN
Categories ON Products.CategoryID = Categories.CategoryID
WHERE Categories.CategoryName = 'Beverages';
```

16.2 Screenshot

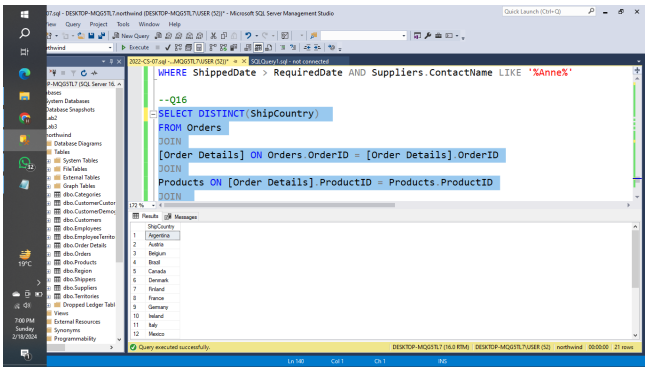

Figure 16: Screenshot of the results