Exercise 1 – Northwind Queries

1.1 Write a query that lists all Customers in either Paris or London. Include Customer ID, Company Name and all address fields.

```
SELECT c.CustomerID , c.CompanyName,
c.Address, c.City, c.Country, c.PostalCode
FROM Customers c
WHERE c.City = 'Paris' OR  c.City = 'London'
```

1.2 List all products stored in bottles.

```
SELECT *
FROM Products p
WHERE p.QuantityPerUnit LIKE  '%bottle%'
```

1.3 Repeat question above but add in the Supplier Name and Country.

```
SELECT s.SupplierID, s.CompanyName, s.Country, p.ProductName
FROM Products p
INNER JOIN Suppliers s
ON p.SupplierID = s.SupplierID
WHERE p.QuantityPerUnit LIKE  '%bottle%'
```

1.4 Write an SQL Statement that shows how many products there are in each category. Include Category Name in result set and list the highest number first.

```
SELECT c.CategoryID, c.CategoryName,
COUNT(c.CategoryID) AS "Total_Products_in_Each_Category"
FROM Products p
INNER JOIN Categories c
ON p.CategoryID = c.CategoryID
GROUP BY  c.CategoryID, c.CategoryName
ORDER BY "Total_Products_in_Each_Category" DESC
```

1.5 List all UK employees using concatenation to join their title of courtesy, first name and last name together. Also include their city of residence.

```
SELECT e.TitleOfCourtesy + ' ' + e.FirstName + ' ' + e.LastName
AS "Full_Employee_Name" , e.City
FROM Employees e
WHERE e.Country LIKE 'UK'
```

1.6 List Sales Totals for all Sales Regions (via the Territories table using 4 joins) with a Sales Total greater than 1,000,000. Use rounding or FORMAT to present the numbers.

```
SELECT  T.RegionID,
 ROUND(SUM((od.UnitPrice * od.quantity) * (1 - od.Discount)), 2)
AS "Sales_Totals"
FROM  Territories t
INNER JOIN EmployeeTerritories et
ON t.TerritoryID = et.TerritoryID
INNER JOIN Employees e
ON et.EmployeeID = e.EmployeeID
INNER JOIN Orders o
ON o.EmployeeID = e.EmployeeID
INNER JOIN [Order Details] od
ON o.OrderID = od.OrderID
GROUP BY  T.RegionID
HAVING SUM((od.UnitPrice * od.quantity) * (1 - od.Discount)) > 1000000
```

1.7 Count how many Orders have a Freight amount greater than 100.00 and either USA or UK as Ship Country.

```
SELECT o.ShipCountry,
COUNT(*) AS "Freight_Count_Over_100"
FROM Orders o
WHERE (o.Freight ) > 100.00 AND o.ShipCountry IN ('UK' , 'USA')
GROUP BY o.ShipCountry
```

1.8 Write an SQL Statement to identify the Order Number of the Order with the highest amount(value) of discount applied to that order.

```
SELECT TOP 1 od.OrderID, MAX((od.UnitPrice * od.Quantity) * od.Discount)
AS "Discounted_Value"
FROM [Order Details] od
GROUP BY od.OrderID
ORDER BY  MAX((od.UnitPrice * od.Quantity) * od.Discount) DESC
```

Exercise 2 – Create Spartans Table

2.1 Write the correct SQL statement to create the following table:

Spartans Table – include details about all the Spartans on this course. Separate Title, First Name and Last Name into separate columns, and include University attended, course taken, and mark achieved. Add any other columns you feel would be appropriate.

```
CREATE TABLE spartans_table (
 seperate_title VARCHAR(10),
 first_name VARCHAR(50),
 last_name VARCHAR(50),
 university VARCHAR (100),
 course_taken VARCHAR (100),
 mark_achieved VARCHAR (20),
 favourite_number INT
)
```

2.2 Write SQL statements to add the details of the Spartans in your course to the table you have created.

```
INSERT INTO spartans_table VALUES (
  'Mr',
  'Rashawn',
  'Henry',
  'Wolverhampton',
  'Software Engineering',
  '1st',
   13
)
INSERT INTO spartans_table VALUES (
  'Mr',
  'Alex',
  'Ng',
  'Essex',
  'Computer Science',
  '1st',
   25
)
INSERT INTO spartans_table VALUES (
  'Mr',
  'Josh',
  'Weeden',
  'Kent',
  'Computer Science',
  '2:1',
   23
)
```

```
INSERT INTO spartans_table VALUES (
  'Mr',
  'Gregory',
  'Spratt',
  'Bath',
  'Electrical Engineering',
  '2:1',
   103
)
INSERT INTO spartans_table VALUES (
  'Mr',
  'Ahmed',
  'Abdul Rahman',
  'London',
  'Economics',
  '2:2',
   19
)
INSERT INTO spartans_table VALUES (
  'Mr',
  'Andrei',
  'Pavel',
  'Romania',
  'Mechanical Engineering',
  '2:1',
   19
)
INSERT INTO spartans_table VALUES (
  'Mr',
  'Asakar',
  'Hussain',
  'Birmingham',
  'Electrical Engineering',
  '1st',
   45
)
INSERT INTO spartans_table VALUES (
  'Lord',
  'Ben',
  'Middlehurst',
  'London',
  'Electrical Engineering',
  '2:2',
   80
)
```

```sql
INSERT INTO spartans_table VALUES (
  'Mr',
  'Benjamin',
  'Balls',
  'London',
  'Chemical Engineering',
  '1st',
   2000
)
INSERT INTO spartans_table VALUES (
  'Mr',
  'Daniel',
  'Alldrit',
  'Wales',
  'Computer Science',
  '1st',
   59
)
INSERT INTO spartans_table VALUES (
  'Mr',
  'Ismail',
  'Kadir',
  'London',
  'Computer Science',
  '1st',
   56
)
INSERT INTO spartans_table VALUES (
  'Mr',
  'James',
  'Fletcher',
  'Wales',
  'Computer Science',
  '2:2',
   3
)
INSERT INTO spartans_table VALUES (
  'Mr',
  'Jammie',
  'Hammond',
  'Essex',
  'Chemical Engineering',
  '1st',
   59
)
```

```sql
INSERT INTO spartans_table VALUES (
  'Mr',
  'Nathan',
  'Johnston',
  'Leeds',
  'Mechanical Engineering',
  '2:1',
   1
)
INSERT INTO spartans_table VALUES (
  'Mr',
  'Sidhant',
  'Khosla',
  'London',
  'Economics',
  '1st',
   32
)

INSERT INTO spartans_table VALUES (
  'Mr',
  'Timin',
  'Rickaby',
  'London',
  'Computer Science',
  '2:2',
   78
)
INSERT INTO spartans_table VALUES (
  'Mr',
  'Yusuf',
  'Uddin',
  'London',
  'Business',
  '2:1',
   37
)
```

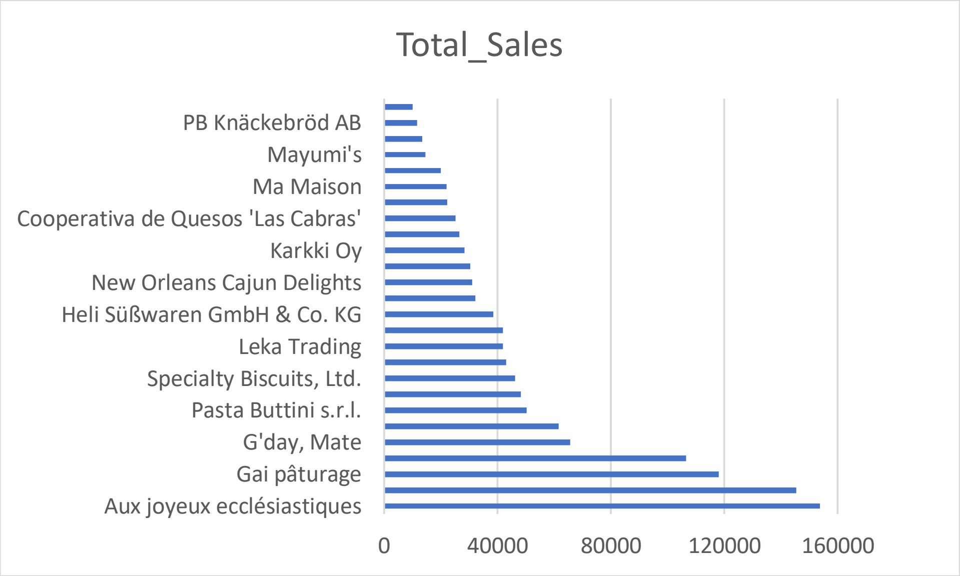
Exercise 3 – Northwind Data Analysis linked to Excel

3.1 List all Employees from the Employees table and who they report to. No Excel required.

```
SELECT e.FirstName +' '+e.LastName AS "Employee_Full_Name", e.ReportsTo, eA
.FirstName
FROM Employees e
LEFT JOIN Employees eA
ON e.ReportsTo = eA.EmployeeID
```

3.2 List all Suppliers with total sales over $10,000 in the Order Details table. Include the Company Name from the Suppliers Table and present as a bar chart.

```
SELECT  s.CompanyName,
SUM((od.UnitPrice * od.Quantity) * (1- od.Discount)) AS "Total_Sales"
FROM Suppliers s
INNER JOIN Products p
ON p.SupplierID = s.SupplierID
INNER JOIN [Order Details] od
ON od.ProductID = p.ProductID
GROUP BY s.CompanyName
HAVING SUM((od.UnitPrice * od.Quantity) * (1- od.Discount)) > 10000
ORDER BY "Total_Sales" DESC
```

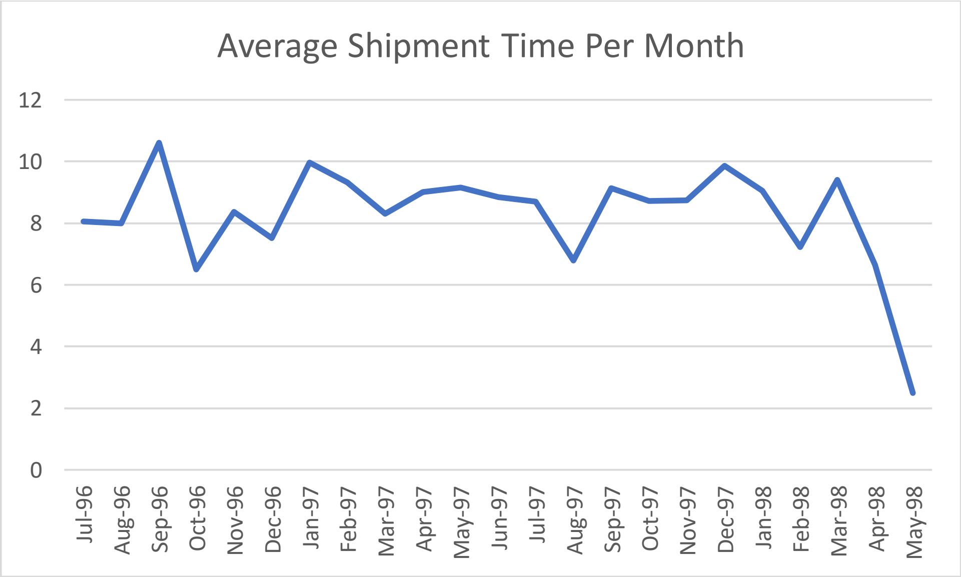


3.3 List the Top 10 Customers YTD for the latest year in the Orders file. Based on total value of orders shipped. No Excel required.

```sql
SELECT TOP 10 c.CompanyName,
SUM((od.UnitPrice * od.Quantity) * (1- od.Discount))
AS "Total_Value_Of_Shipped Orders"
FROM Orders o
INNER JOIN Customers c
ON c.CustomerID = o.CustomerID
INNER JOIN [Order Details] od
ON  od.OrderID = o.OrderID
WHERE YEAR(o.OrderDate) = '1998' AND o.ShippedDate < o.RequiredDate
GROUP BY c.CompanyName
ORDER BY "Total_Value_Of_Shipped Orders" DESC
```

3.4 Plot the Average Ship Time by month for all data in the Orders Table using a line chart.

```sql
SELECT MONTH(o.OrderDate) AS "Month",
YEAR(o.OrderDate) AS "Year" ,
FORMAT (o.OrderDate, 'MMM-yy') AS "Year-Month",
AVG(CAST(DATEDIFF(d,  o.OrderDate, o.ShippedDate) AS Decimal (4,1)))
AS "Average_Ship_Days"
FROM Orders o
GROUP BY YEAR(o.OrderDate) , MONTH(o.OrderDate) , FORMAT (o.OrderDate, 'MMM-yy')
ORDER BY YEAR(o.OrderDate), MONTH(o.OrderDate)
```


Average Shipment Time Per Month
Jul-96
Aug-96
Sep-96
Oct-96
Nov-96
Dec-96
Jan-97
Feb-97
Mar-97
Apr-97
May-97
Jun-97
Jul-97
Aug-97
Sep-97
Oct-97
Nov-97
Dec-97
Jan-98
Feb-98
Mar-98
Apr-98
May-98