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.

```
--1.1
SELECT c.CustomerID, c.CompanyName,
    --Address Fields
    CONCAT(c.Address, ', ', c.City, ', ', c.PostalCode, ', ', c.Country) AS "Company Address"
FROM Customers c
WHERE c.City In('Paris','London');
```

Result:

	CustomerID	CompanyName	Company Address
1	AROUT	Around the Horn	120 Hanover Sq., London, WA1…
2	BSBEV	B's Beverages	Fauntleroy Circus, London, E…
3	CONSH	Consolidated Holdings	Berkeley Gardens 12 Brewery…
4	EASTC	Eastern Connection	35 King George, London, WX3 …
5	NORTS	North/South	South House 300 Queensbridge…
6	PARIS	Paris spécialités	265, boulevard Charonne, Par…
7	SEVES	Seven Seas Imports	90 Wadhurst Rd., London, OX1…
8	SPECD	Spécialités du monde	25, rue Lauriston, Paris, 75…

1.2 List all products stored in bottles.

```
--1.2
SELECT p.ProductName
FROM Products p
--Any product stored in bottles
WHERE p.QuantityPerUnit LIKE '%bottle%';
```

Result:

	ProductName
1	Chang
2	Aniseed Syrup
3	Genen Shouyu
4	Sasquatch Ale
5	Steeleye Stout
6	Côte de Blaye
7	Chartreuse verte
8	Sirop d'érable
9	Louisiana Fiery Hot Pepper S…
10	Laughing Lumberjack Lager
11	Outback Lager
12	Rhönbräu Klosterbier

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

```
--1.3
SELECT p.ProductName, s.CompanyName AS "Supplier Name" , s.Country
FROM Products p
INNER JOIN Suppliers s
ON p.SupplierID = s.SupplierID
WHERE p.QuantityPerUnit LIKE '%bottle%';
```

Result:

	ProductName	Supplier Name	Country
1	Chang	Exotic Liquids	UK
2	Aniseed Syrup	Exotic Liquids	UK
3	Genen Shouyu	Mayumi's	Japan
4	Sasquatch Ale	Bigfoot Breweries	USA
5	Steeleye Stout	Bigfoot Breweries	USA
6	Côte de Blaye	Aux joyeux ecclésiastiques	France
7	Chartreuse verte	Aux joyeux ecclésiastiques	France
8	Sirop d'érable	Forêts d'érables	Canada
9	Louisiana Fiery Hot Pepper S…	New Orleans Cajun Delights	USA
10	Laughing Lumberjack Lager	Bigfoot Breweries	USA
11	Outback Lager	Pavlova, Ltd.	Australia
12	Rhönbräu Klosterbier	Plutzer Lebensmittelgroßmärk…	Germany

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.

```
--1.4
SELECT c.CategoryName,
    SUM(p.CategoryID) AS "Num Products"
FROM Categories c
LEFT JOIN Products p
ON c.CategoryID = p.CategoryID
GROUP BY c.CategoryName
ORDER BY c.CategoryName DESC;
```

Result:

	CategoryName	Num Products
1	Seafood	96
2	Produce	35
3	Meat/Poultry	36
4	Grains/Cereals	35
5	Dairy Products	40
6	Confections	39
7	Condiments	24
8	Beverages	12

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.

```
--1.5
SELECT CONCAT(e.TitleOfCourtesy, e.FirstName, ' ', e.LastName) AS "Employee Name", e.City
FROM Employees e
WHERE e.Country = 'UK'
```

Result:

	Employee Name	City
1	Mr.Steven Buchanan	London
2	Mr.Michael Suyama	London
3	Mr.Robert King	London
4	Ms.Anne Dodsworth	London

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.

```
--1.6
SELECT r.RegionDescription AS "Region",
    --Format a comma after every 3 0's for readability
    FORMAT(SUM(od.UnitPrice * od.Quantity * (1-Discount)), 'N') AS "Sales Totals by Region"
FROM Region r
INNER JOIN Territories t
ON r.RegionID = t.RegionID
INNER JOIN EmployeeTerritories et
ON t.TerritoryID = et.TerritoryID
INNER JOIN Orders o
ON o.EmployeeID = et.EmployeeID
INNER JOIN [Order Details] od
on od.OrderID = o.OrderID
--Filter for regions with sales totals only > 1,000,000
GROUP BY r.RegionDescription HAVING SUM(od.UnitPrice * od.Quantity * (1-Discount)) > 1000000;
```

Result:

	Region	Sales Totals by Region
1	Northern …	1,048,605.58
2	Eastern …	2,730,198.01
3	Western …	1,615,248.00

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

```
--1.7
SELECT COUNT(o.OrderID) AS "Orders > 100 & in UK/USA"
FROM Orders o
WHERE o.Freight > 100
AND o.ShipCountry IN('USA', 'UK');
```

Result:

	Orders > 100 & in UK/USA
1	49

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.

```
--1.8
SELECT TOP 1 od.OrderID, SUM(od.UnitPrice * od.Quantity * od.Discount) AS "Discounted Value"
FROM [Order Details] od
GROUP BY od.OrderID
ORDER BY "Discounted Value" DESC
```

Result:

	OrderID	Discounted Value
1	11030	3706.8499755859375

Exercise 2 – Create Spartans Table

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

```sql
--2.1
CREATE DATABASE connor_db;
USE connor_db;

DROP TABLE IF EXISTS Spartans
CREATE TABLE Spartans
(
    title VARCHAR(40),
    first_name VARCHAR(20),
    last_name VARCHAR(20),
    university_attended VARCHAR(30),
    course_taken VARCHAR(30),
    --Where 1.00 = %100. Allows up to 2 decimal places
    mark_achieved DECIMAL(3,2)
);
SP_HELP Spartans;
```

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

```sql
--2.2
INSERT INTO Spartans
VALUES
(
    'DummyTitle1', 'DummyFirstName1', 'DummyLastName1', 'DummyUniversity1', 'JAVA SDET', '20'
),(
    'Junior Java Dev Consultant', 'Thomas', 'Andrews', 'Birmingham University', 'Java Development', '92'
),(
    'Junior C# SDET Consultant', 'Alice', 'Bobbington', 'Surry University', 'C# SDET', '82'
)
SELECT * FROM Spartans;
```

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. Please mention the Employee Names and the ReportTo names.

```
--3.1
SELECT CONCAT(e.FirstName, ' ', e.LastName) AS "Employee Name",
    CONCAT(e2.FirstName, ' ', e2.LastName) AS "Reports To"
FROM Employees e
LEFT JOIN Employees e2
ON e.ReportsTo = e2.EmployeeID;
```

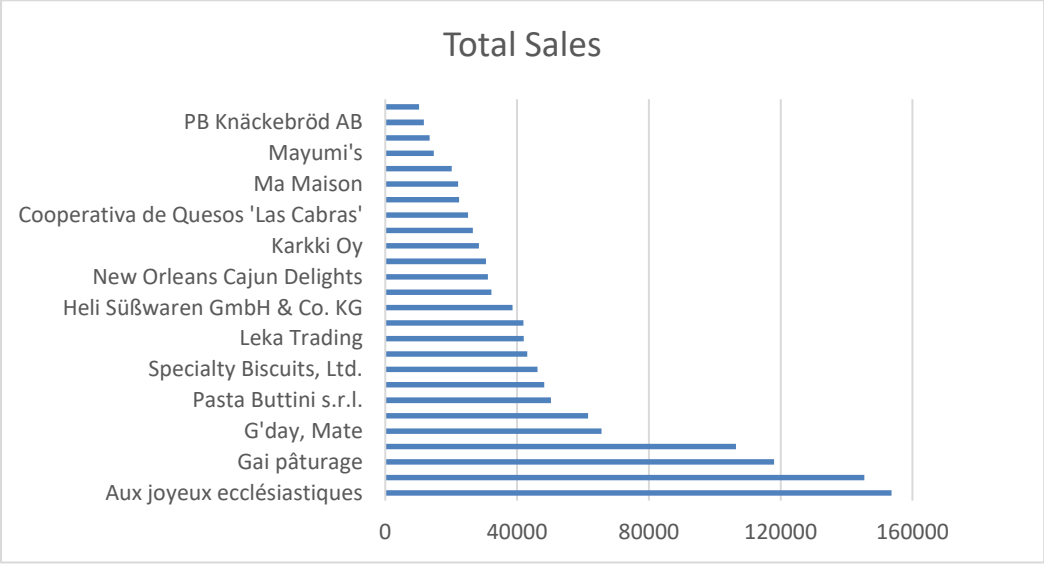
Result:

	Employee Name	Reports To
1	Nancy Davolio	Andrew Fuller
2	Andrew Fuller	
3	Janet Leverling	Andrew Fuller
4	Margaret Peacock	Andrew Fuller
5	Steven Buchanan	Andrew Fuller
6	Michael Suyama	Steven Buchanan
7	Robert King	Steven Buchanan
8	Laura Callahan	Andrew Fuller
9	Anne Dodsworth	Steven Buchanan

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 as below:

```
--3.2
SELECT s.CompanyName AS "Supplier Name",
    SUM(od.Quantity * od.UnitPrice * (1 - od.Discount)) AS "Total Sales"
FROM Suppliers s
INNER JOIN Products p
ON s.SupplierID = p.SupplierID
INNER JOIN [Order Details] od
ON p.ProductID = od.ProductID
GROUP BY s.CompanyName HAVING SUM(od.Quantity * od.UnitPrice * (1 - od.Discount)) > 10000
ORDER BY "Total Sales";
```

Graph:



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.

```
--3.3
                                        --Get exact value of each order - including applied discounts
SELECT TOP 10 c.CompanyName AS "Customers YTD",
    SUM(od.UnitPrice * od.Quantity * (1-od.Discount)) AS "Total Value of Orders Shipped"
FROM Customers c
INNER JOIN Orders o
ON c.CustomerID = o.CustomerID
INNER JOIN [Order Details] od
ON o.OrderID = od.OrderID
--Latest year is 1998, want all orders within that year
WHERE DATEDIFF(YEAR, o.OrderDate, '01/01/1998') < 1
GROUP BY c.CompanyName
--Order by most to least value to get the 10 most valuable customers
ORDER BY "Total Value of Orders Shipped" DESC
```

Result:

	Customers YTD	Total Value of Orders Shipped
1	Ernst Handel	41210.65002441406
2	QUICK-Stop	37217.315002441406
3	Save-a-lot Markets	36310.10977935791
4	Hanari Carnes	23821.199989318848
5	Rattlesnake Canyon Grocery	21238.270441055298
6	Hungry Owl All-Night Grocers	20402.11993408203
7	Königlich Essen	19582.773986816406
8	White Clover Markets	15278.89998626709
9	Folk och fä HB	13644.067497253418
10	Suprêmes délices	11644.599998474121

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

```
--3.4
SELECT CONCAT(MONTH(o.OrderDate), ', ', YEAR(o.OrderDate)) AS "Order Date",
    AVG(DATEDIFF(DAY, o.OrderDate, o.ShippedDate)) AS "Avg Ship Time"
FROM Orders o
WHERE o.ShippedDate IS NOT NULL
GROUP BY YEAR(o.OrderDate), MONTH(o.OrderDate)
ORDER BY YEAR(o.OrderDate), MONTH(o.OrderDate)
```

Graph: