SQL Project

Trainee Name: Andrew Osborne

Q1.1	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, CONCAT(c.Address,+ ',' + c.City,',' + c.Country,+ ','+ c.PostalCode) AS "Full Adress"
FROM Customers c
WHERE c.City IN ('Paris','London');
```

Q1.2	List all products stored in bottles.

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

Q1.3	Repeat question above but add in the Supplier Name and Country

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

Q1.4	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.CategoryName, COUNT(p.CategoryID) AS "Products in each category"
FROM Products p
INNER JOIN Categories c
ON c.CategoryID=p.CategoryID
GROUP BY c.CategoryName
ORDER BY 'Products in each category' DESC
```

Q1.5	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 CONCAT(e.TitleOfCourtesy,' ' ,e.FirstName,' ', e.LastName) AS "Employee Full Name", e.City
FROM Employees e
WHERE e.Country LIKE 'UK'
```

Q1.6	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, FORMAT(SUM(od.Quantity*od.UnitPrice), 'C') AS "Number of sales"
FROM Territories t
INNER JOIN EmployeeTerritories et ON t.TerritoryID=et.TerritoryID
INNER JOIN Employees e ON e.EmployeeID=et.EmployeeID
INNER JOIN Orders o ON e.EmployeeID=o.EmployeeID
INNER JOIN [Order Details]od ON O.OrderID=OD.OrderID
GROUP BY t.RegionID
HAVING SUM(od.Quantity*od.UnitPrice) > 1000000
ORDER BY t.RegionID
```

Q1.7	Count how many Orders have a Freight amount greater than 100.00 and either USA or UK as Ship Country.

```
SELECT COUNT(o.Freight) AS "Total amount of orders greater than 100"
FROM Orders o
WHERE o.ShipCountry IN ('USA','UK') AND o.Freight > 100
```

Q1.8	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, ROUND(SUM(od.UnitPrice*od.Quantity*od.Discount), 2) AS "Discount amount"
FROM [Order Details]od
GROUP BY od.OrderID
ORDER BY "Discount amount" DESC
```

Q2.1	Write the correct SQL statement to create the following table: CREATING A SPARTANS TABLE

```
CREATE TABLE [Spartans] (
    [SpartansID] INTEGER NOT NULL IDENTITY(1, 1),
    [Title] VARCHAR(255) NULL,
    [FirstName] VARCHAR(255) NULL,
    [Surname] VARCHAR(255) NULL,
    [University] VARCHAR(255) NULL,
    [Course] VARCHAR(255) NULL,
    [Mark] VARCHAR(255) NULL,
    PRIMARY KEY ([SpartansID])
);
```

Q2.2	Write SQL statements to add the details of the Spartans in your course to the table you have created.

```
INSERT INTO Spartans([Title],[FirstName],[Surname],[University],[Course],[Mark])
 VALUES('Miss.','Georgina','Bartlett','Newcastle University','Archaeology','2:1'),
('Mr.','Humza','Malak','University of Kent','Computer Science','2:2'),
('Mr.','Ibrahim','Bocus','University of Leicester','Mechanical Engineering','2:1'),
('Mr.','Bari','Allali','Lancaster University','Business Economics','2:1'),
('Mr.','Mehdi','Shamaa','University of Nottingham','Philosophy and Economics','2:2'),
('Miss.','Anais','Tang','Edinburgh University','Modern Languages','2:1'),
('Mr.','Saheed','Lamina','University of Warwick','Politics and International Studies','2:1'),
('Mr.','Man-Wai','Tse','University of Hertfordshire','Aerospace Engineering ','2:1'),
('Mr.','Sohaib','Sohail','Brunel University London','Communications and Media Studies ','2:2'),
('Miss.','Ugne','Okmanaite','Aston University','International Business & Management','2:1'),
('Mr.','Daniel','Teegan','University of Brighton','Product Design','2:2'),
('Mr.','Max','Palmer','University of Birmingham','Ancient History','2:1'),
('Mr.','Andrew','Osbourne','University of Portsmouth','Biomedical Science','2:1');
```

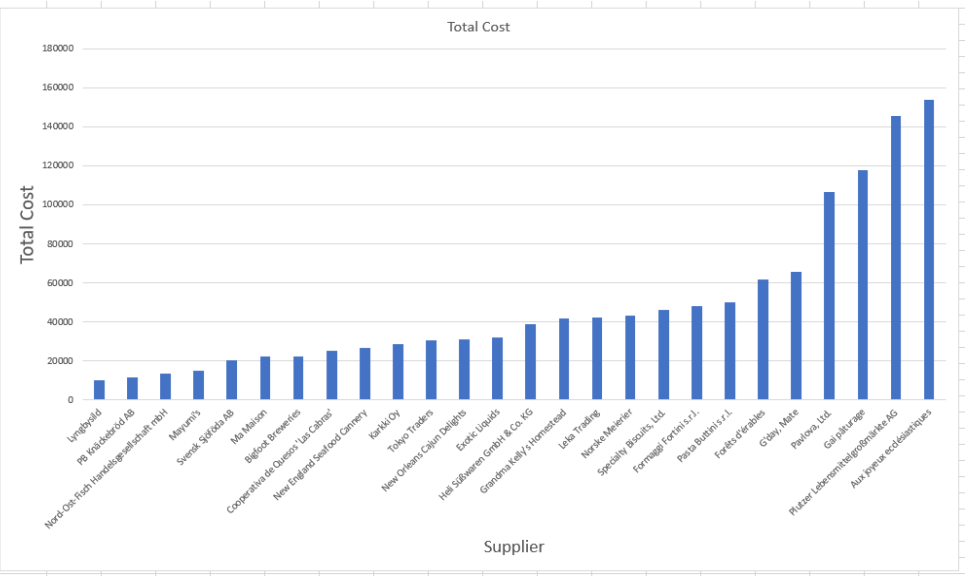
Q3.1	List all Employees from the Employees table and who they report to.

```
SELECT DISTINCT e.EmployeeID, e.ReportsTo,
CONCAT(e.FirstName, ' ', e.LastName) AS "Employee Full Name",
CONCAT(e2.FirstName,' ', e2.LastName) AS "Manager Full Name"
FROM Employees e, Employees e2
WHERE e2.EmployeeID = e.ReportsTo
```

Q3.2	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.

```
SELECT s.CompanyName AS "Supplier Company Name",
ROUND(SUM((od.UnitPrice*od.Quantity) * ( 1 -od.Discount)), 2) AS "Total Cost"
FROM [Order Details]od
INNER JOIN Products p ON p.ProductID=od.ProductID
INNER JOIN Suppliers s ON s.SupplierID=p.SupplierID
GROUP BY s.CompanyName
HAVING ROUND(SUM((od.UnitPrice*od.Quantity) * ( 1 -od.Discount)), 2) > 10000
ORDER BY 'Total Cost' DESC
```

Supplier Company Na	Total Cost
Lyngbysild	10221.17
PB Knäckebröd AB	11724.06
Nord-Ost-Fisch Hande	13424.2
Mayumi's	14736.76
Svensk Sjöföda AB	20144.06
Ma Maison	22154.64
Bigfoot Breweries	22391.2
Cooperativa de Quesc	25159.43
New England Seafood	26590.97
Karkki Oy	28442.73
Tokyo Traders	30526.34
New Orleans Cajun De	31167.99
Exotic Liquids	32188.06
Heli Süßwaren GmbH	38653.42
Grandma Kelly's Hom	41953.3
Leka Trading	42017.65
Norske Meierier	43141.51
Specialty Biscuits, Ltd	46243.98
Formaggi Fortini s.r.l.	48225.16
Pasta Buttini s.r.l.	50254.61
Forêts d'érables	61587.57
G'day, Mate	65626.77
Pavlova, Ltd.	106459.78
Gai pâturage	117981.18
Plutzer Lebensmittelgr	145372.4
Aux joyeux ecclésiast	153691.28



Q3.3	List the Top 10 Customers year to date for the latest year in the Orders file. Based on total value of orders shipped. No Excel required.

```
SELECT TOP 10 C.CompanyName, FORMAT(SUM((od.UnitPrice*od.Quantity) * ( 1 -od.Discount)), 'C') AS "Total sales"
FROM [Order Details]od
INNER JOIN Orders o ON o.OrderID=od.OrderID
INNER JOIN Customers c ON c.CustomerID=o.CustomerID
WHERE (SELECT YEAR(MAX(o.ShippedDate))FROM Orders o) = YEAR(o.ShippedDate) AND (o.ShippedDate IS NOT NULL)
GROUP BY c.CompanyName
ORDER BY "Total sales" DESC
```

Q3.4	Plot the Average Ship Time by month for all data in the Orders Table using a line chart as below.

```
SELECT FORMAT(o.ShippedDate, 'MMM-yy') AS "Shipping Month",
AVG(DATEDIFF(DAY, o.OrderDate, o.ShippedDate)) AS "Average Ship Time"
FROM Orders o
WHERE o.ShippedDate IS NOT NULL
GROUP BY
YEAR(o.ShippedDate),
MONTH(o.ShippedDate),
FORMAT(o.ShippedDate, 'MMM-yy')
ORDER BY
YEAR(o.ShippedDate),
MONTH(o.ShippedDate);
```