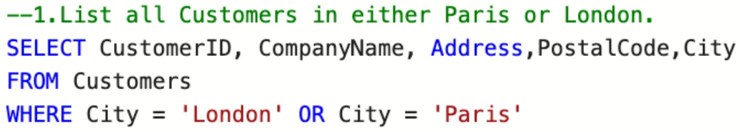
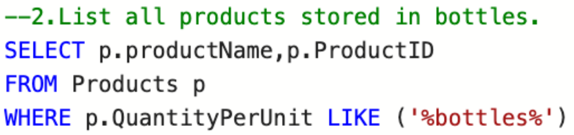
Exercise 1 – Northwind Queries

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



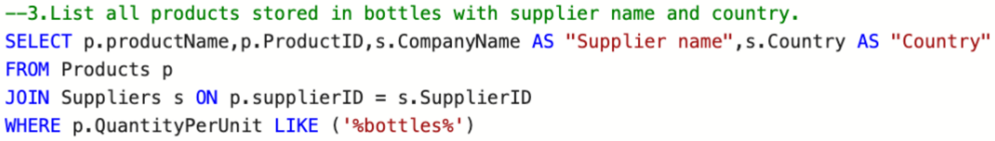
```
--1.List all Customers in either Paris or London.
SELECT CustomerID, CompanyName, Address,PostalCode,City
FROM Customers
WHERE City = 'London' OR City = 'Paris'
```

2. List all products stored in bottles:



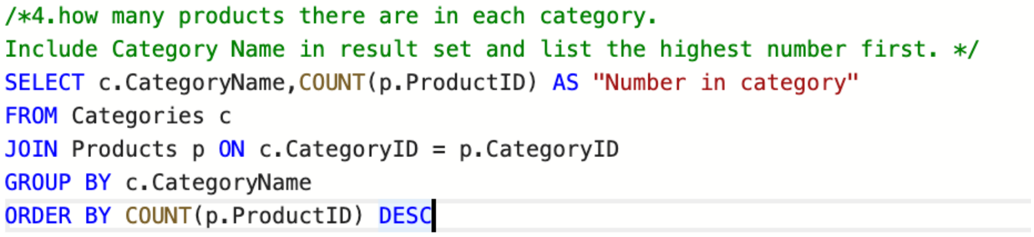
```
--2.List all products stored in bottles.
SELECT p.productName,p.ProductID
FROM Products p
WHERE p.QuantityPerUnit LIKE ('%bottles%')
```

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



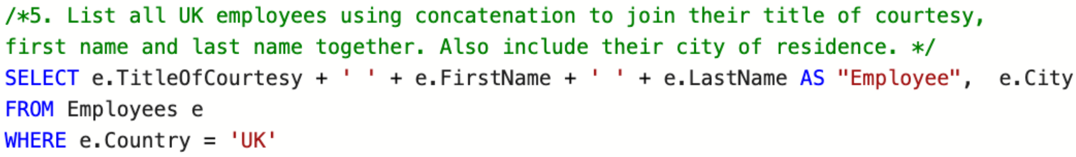
```
--3.List all products stored in bottles with supplier name and country.
SELECT p.productName,p.ProductID,s.CompanyName AS "Supplier name",s.Country AS "Country"
FROM Products p
JOIN Suppliers s ON p.supplierID = s.SupplierID
WHERE p.QuantityPerUnit LIKE ('%bottles%')
```

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:



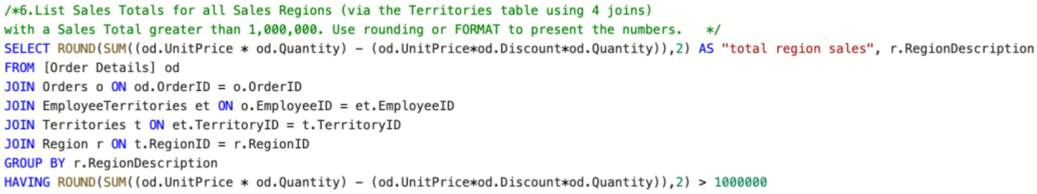
```
/*4.how many products there are in each category.
Include Category Name in result set and list the highest number first. */
SELECT c.CategoryName,COUNT(p.ProductID) AS "Number in category"
FROM Categories c
JOIN Products p ON c.CategoryID = p.CategoryID
GROUP BY c.CategoryName
ORDER BY COUNT(p.ProductID) DESC
```

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:



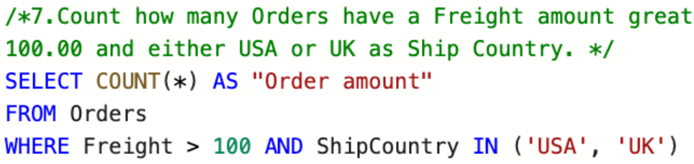
```
/*5. List all UK employees using concatenation to join their title of courtesy,
first name and last name together. Also include their city of residence. */
SELECT e.TitleOfCourtesy + ' ' + e.FirstName + ' ' + e.LastName AS "Employee",  e.City
FROM Employees e
WHERE e.Country = 'UK'
```

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:



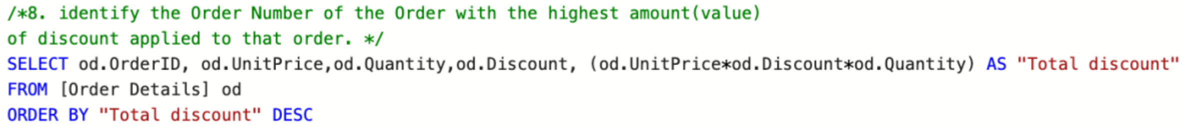
```
/*6.List Sales Totals for all Sales Regions (via the Territories table using 4 joins)
with a Sales Total greater than 1,000,000. Use rounding or FORMAT to present the numbers.   */
SELECT ROUND(SUM((od.UnitPrice * od.Quantity) - (od.UnitPrice*od.Discount*od.Quantity)),2) AS "total region sales", r.RegionDescription
FROM [Order Details] od
JOIN Orders o ON od.OrderID = o.OrderID
JOIN EmployeeTerritories et ON o.EmployeeID = et.EmployeeID
JOIN Territories t ON et.TerritoryID = t.TerritoryID
JOIN Region r ON t.RegionID = r.RegionID
GROUP BY r.RegionDescription
HAVING ROUND(SUM((od.UnitPrice * od.Quantity) - (od.UnitPrice*od.Discount*od.Quantity)),2) > 1000000
```

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



```
/*7.Count how many Orders have a Freight amount greater than
100.00 and either USA or UK as Ship Country. */
SELECT COUNT(*) AS "Order amount"
FROM Orders
WHERE Freight > 100 AND ShipCountry IN ('USA', 'UK')
```

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



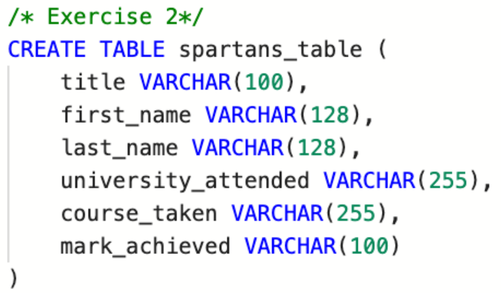
```
/*8. identify the Order Number of the Order with the highest amount(value)
of discount applied to that order. */
SELECT od.OrderID, od.UnitPrice,od.Quantity,od.Discount, (od.UnitPrice*od.Discount*od.Quantity) AS "Total discount"
FROM [Order Details] od
ORDER BY "Total discount" DESC
```

Exercise 2 – Create Spartans Table

1. Write the correct SQL statement to create the following table:

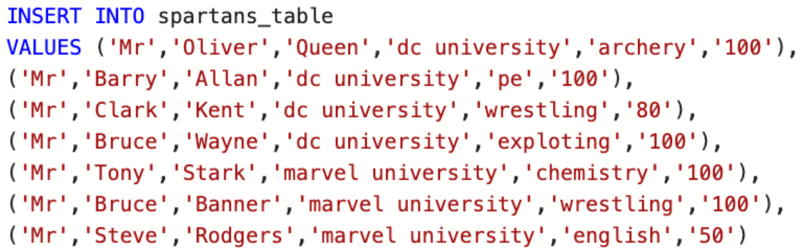
Spartans Table – include details about all the Spartans on this course. Separate Title, First Name and Last Name into separate columns, and include University attended, course taken and mark achieved. Add any other columns you feel would be appropriate.

IMPORTANT NOTE: For data protection reasons do NOT include date of birth in this exercise.



```
/* Exercise 2*/
CREATE TABLE spartans_table (
    title VARCHAR(100),
    first_name VARCHAR(128),
    last_name VARCHAR(128),
    university_attended VARCHAR(255),
    course_taken VARCHAR(255),
    mark_achieved VARCHAR(100)
)
```

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

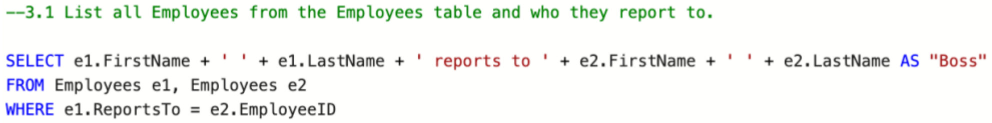


```
INSERT INTO spartans_table
VALUES ('Mr','Oliver','Queen','dc university','archery','100'),
('Mr','Barry','Allan','dc university','pe','100'),
('Mr','Clark','Kent','dc university','wrestling','80'),
('Mr','Bruce','Wayne','dc university','exploting','100'),
('Mr','Tony','Stark','marvel university','chemistry','100'),
('Mr','Bruce','Banner','marvel university','wrestling','100'),
('Mr','Steve','Rodgers','marvel university','english','50')
```

Exercise 3 – Northwind Data Analysis linked to Excel (30 marks)

Write SQL statements to extract the data required for the following charts (create these in Excel):

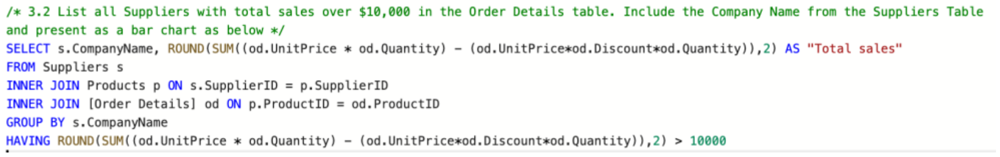
1. List all Employees from the Employees table and who they report to. No Excel required. (5 Marks)



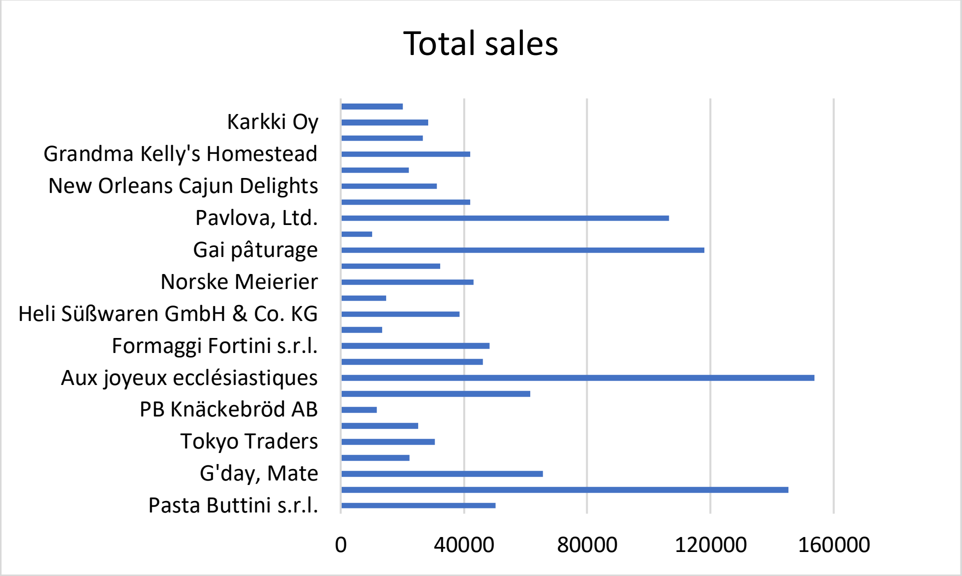
```
--3.1 List all Employees from the Employees table and who they report to.

SELECT e1.FirstName + ' ' + e1.LastName + ' reports to ' + e2.FirstName + ' ' + e2.LastName AS "Boss"
FROM Employees e1, Employees e2
WHERE e1.ReportsTo = e2.EmployeeID
```

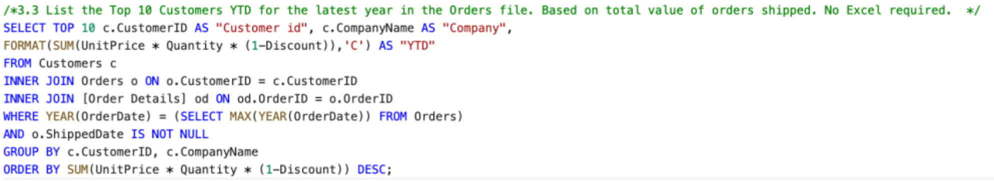
2. List all Suppliers with total sales over $10,000 in the Order Details table. Include the Company Name from the Suppliers Table and present as a bar chart as below: (5 Marks)



```
/* 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 */
SELECT s.CompanyName, ROUND(SUM((od.UnitPrice * od.Quantity) - (od.UnitPrice*od.Discount*od.Quantity)),2) 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 ROUND(SUM((od.UnitPrice * od.Quantity) - (od.UnitPrice*od.Discount*od.Quantity)),2) > 10000
```

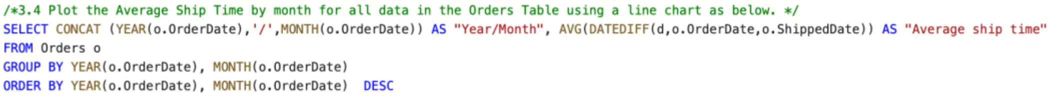


3. List the Top 10 Customers YTD for the latest year in the Orders file. Based on total value of orders shipped. No Excel required. (10 Marks)

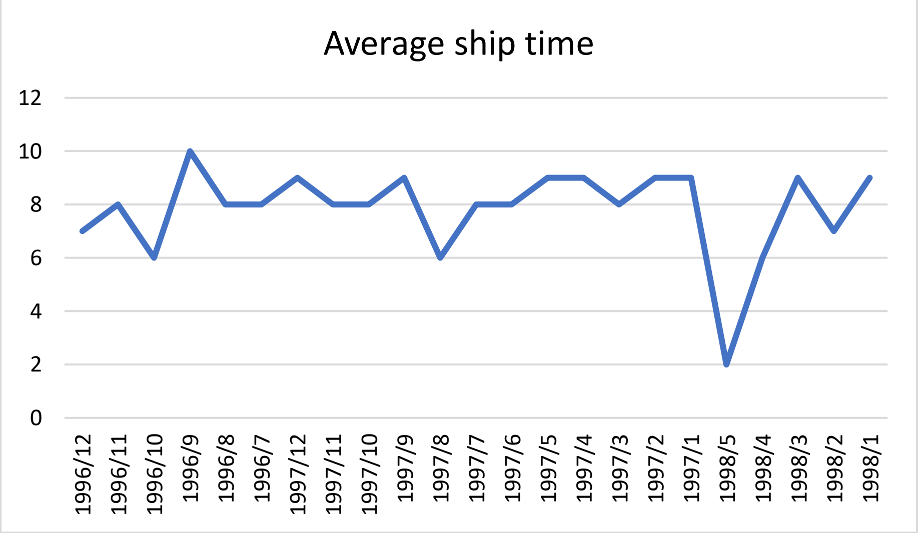


```
/*3.3 List the Top 10 Customers YTD for the latest year in the Orders file. Based on total value of orders shipped. No Excel required.  */
SELECT TOP 10 c.CustomerID AS "Customer id", c.CompanyName AS "Company",
FORMAT(SUM(UnitPrice * Quantity * (1-Discount)),'C') AS "YTD"
FROM Customers c
INNER JOIN Orders o ON o.CustomerID = c.CustomerID
INNER JOIN [Order Details] od ON od.OrderID = o.OrderID
WHERE YEAR(OrderDate) = (SELECT MAX(YEAR(OrderDate)) FROM Orders)
AND o.ShippedDate IS NOT NULL
GROUP BY c.CustomerID, c.CompanyName
ORDER BY SUM(UnitPrice * Quantity * (1-Discount)) DESC;
```

4. Plot the Average Ship Time by month for all data in the Orders Table using a line chart as below. (10 Marks)



```
/*3.4 Plot the Average Ship Time by month for all data in the Orders Table using a line chart as below. */
SELECT CONCAT (YEAR(o.OrderDate),'/',MONTH(o.OrderDate)) AS "Year/Month", AVG(DATEDIFF(d,o.OrderDate,o.ShippedDate)) AS "Average ship time"
FROM Orders o
GROUP BY YEAR(o.OrderDate), MONTH(o.OrderDate)
ORDER BY YEAR(o.OrderDate), MONTH(o.OrderDate)  DESC
```


Average ship time
1996/12
1996/11
1996/10
1996/9
1996/8
1996/7
1997/12
1997/11
1997/10
1997/9
1997/8
1997/7
1997/6
1997/5
1997/4
1997/3
1997/2
1997/1
1998/5
1998/4
1998/3
1998/2
1998/1