SQL Project

Joel Fright

Exercise 1

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,
       CONCAT(c.City, ', ',c.Country) AS "City & Country", c.PostalCode, c.Region
FROM Customers c
WHERE c.City = 'Paris' OR c.City = 'London'
```

1.2) *List all products stored in bottles.*

```
SELECT p.ProductID, p.ProductName FROM Products p
WHERE CHARINDEX('bottles',p.QuantityPerUnit) > 0
```

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

```
SELECT p.ProductID, p.ProductName , s.CompanyName AS "Supplier Name", s.Country
FROM Products p
INNER JOIN Suppliers s ON p.SupplierID = s.SupplierID
WHERE CHARINDEX('bottles',p.QuantityPerUnit) > 0
```

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.CategoryName, COUNT(*) AS "Num of Products" FROM Products p
INNER JOIN Categories c ON p.CategoryID = c.CategoryID
GROUP BY p.CategoryID, c.CategoryName ORDER BY "Num of Products"
```

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 CONCAT(e.TitleOfCourtesy , ' ' , e.FirstName , ' ' , e.LastName) AS "Name",
e.City FROM Employees e
WHERE e.Country = '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 ROUND(SUM((od.Quantity*od.UnitPrice)*(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 e.EmployeeID = o.EmployeeID
INNER JOIN [Order Details] od ON o.OrderID = od.OrderID
GROUP BY t.RegionID
HAVING ROUND(SUM((od.Quantity*od.UnitPrice) * (1-od.Discount)),2) > 1000000
```

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

```
SELECT COUNT(*) AS "Num of Orders w/ +100 Freight & from UK or USA" FROM Orders o
WHERE o.Freight > 100 AND (o.ShipCountry = 'UK' OR o.ShipCountry = 'USA')
```

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 o.OrderID,od.Discount FROM Orders o
INNER JOIN [Order Details] od ON o.OrderID = od.OrderID
ORDER BY od.Discount DESC
```

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
(
    spartan_id INT IDENTITY(1,1) PRIMARY KEY,
    first_name VARCHAR(10) NOT NULL,
    second_name VARCHAR(20) NOT NULL,
    university_attended VARCHAR(30) NOT NULL DEFAULT 'Some Uni',
    course_taken VARCHAR(30) NOT NULL DEFAULT 'Comp Sci',
    mark_achieved INT NOT NULL DEFAULT 60
)
```

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

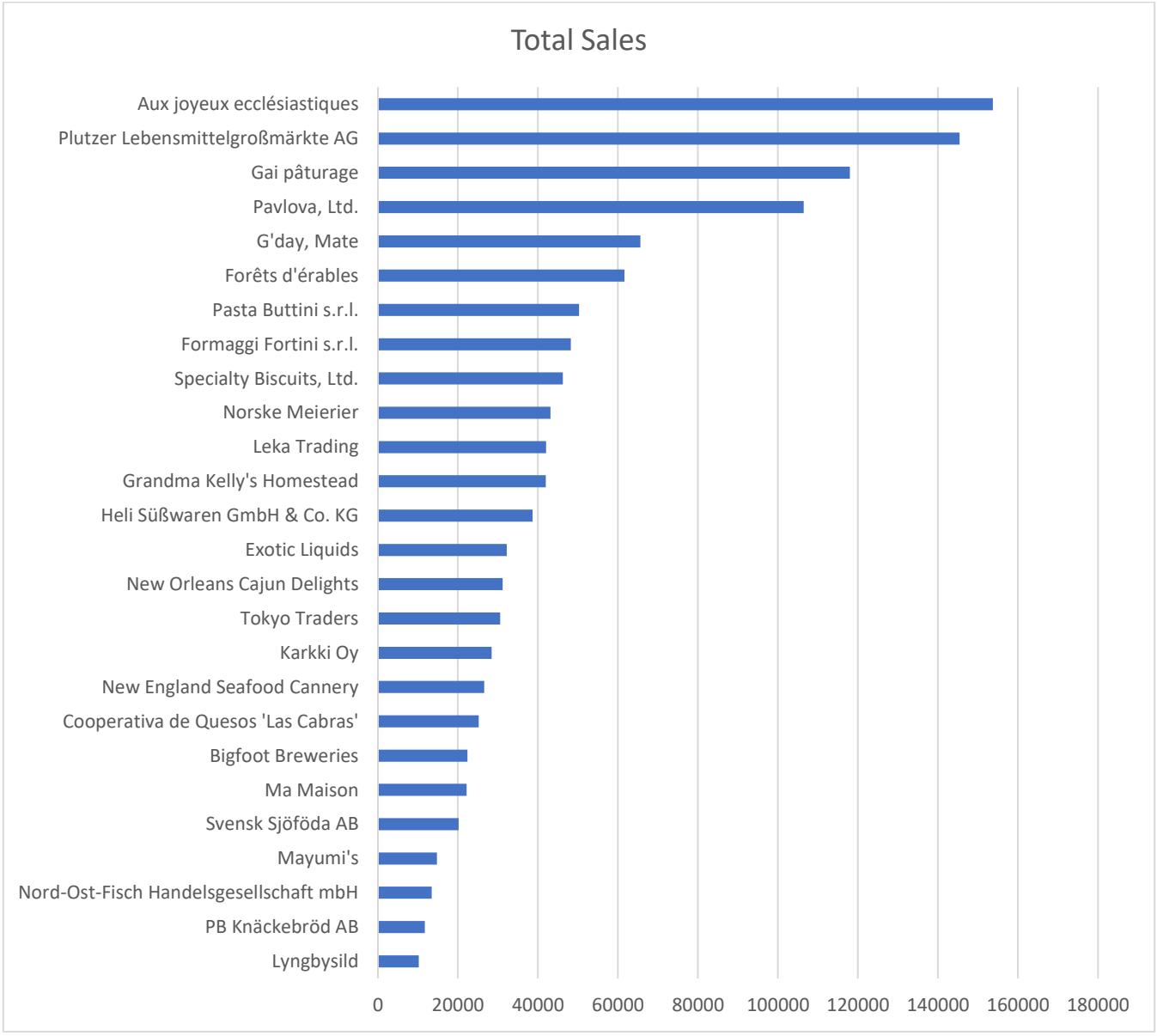
```
INSERT INTO Spartans (first_name, second_name) VALUES
('Bradley','Williams'),
('Kurtis','Hanson'),
('Aaron','Banjoko'),
('Dominic','Cogan-Tucker'),
('Wahdel','Woodhouse'),
('Malik','Shams'),
('Joel','Fright')
```

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

```
SELECT CONCAT(e.FirstName, ' ',e.LastName) AS "Name",
    CONCAT(e2.FirstName, ' ',e2.LastName) AS "Reports To"
FROM Employees e
LEFT JOIN Employees e2 ON e.ReportsTo = e2.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 AS "Suppiler Name",
    ROUND(SUM((od.Quantity*od.UnitPrice)*(1-od.Discount)),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.SupplierID, s.CompanyName
HAVING ROUND(SUM((od.Quantity*od.UnitPrice)*(1-od.Discount)),2) > 10000
ORDER BY "Total Sales"
```



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, c.CompanyName,
    ROUND(SUM((od.Quantity*od.UnitPrice)*(1-od.Discount)),2) AS "Total Spent"
FROM Customers c
INNER JOIN Orders o ON c.CustomerID = o.CustomerID
INNER JOIN [Order Details] od ON o.OrderID = od.OrderID
WHERE (YEAR(o.OrderDate)) =
(SELECT TOP 1 YEAR(o2.OrderDate) FROM Orders o2 ORDER BY YEAR(o2.OrderDate) DESC)
AND o.ShippedDate IS NOT NULL
GROUP BY c.CustomerID, c.CompanyName
ORDER BY "Total Spent" DESC
```

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

```
SELECT CONCAT(MONTH(o.OrderDate), '/' ,YEAR(o.OrderDate)) AS "Month",
    AVG(DATEDIFF(dd,o.OrderDate,o.ShippedDate)) AS "Avg Ship Time By Month"
FROM Orders o
GROUP BY MONTH(o.OrderDate), YEAR(o.OrderDate)
ORDER BY YEAR(o.OrderDate), MONTH(o.OrderDate)
```