1 SQL Practical Exercise

1 Exercise 1 – Northwind Queries (40 marks: 5 for each question)

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.Region, c.PostalCode, c.Country
FROM Customers c
WHERE c.City = 'London' OR c.City = 'Paris'
```

1.2 List all products stored in bottles.

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

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 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 "Number of Products in Category"
FROM Products p INNER JOIN Categories c ON p.CategoryID = c.CategoryID
GROUP BY c.CategoryID, c.CategoryName
ORDER BY COUNT(c.CategoryID) 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 CONCAT(e.TitleOfCourtesy, ' ', e.FirstName, ' ', e.LastName) AS "Employee Name"
FROM Employees e
WHERE 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 t.RegionID, r.RegionDescription,
FORMAT(SUM(od.Quantity*od.UnitPrice*(1-od.Discount)), 'C', 'en-gb') AS "Total Region Sales"
FROM [Order Details] od
INNER JOIN Orders o ON od.OrderID = o.OrderID
INNER JOIN Employees e ON o.EmployeeID = e.EmployeeID
INNER JOIN EmployeeTerritories et ON e.EmployeeID = et.EmployeeID
INNER JOIN Territories t ON et.TerritoryID = t.TerritoryID
INNER JOIN Region r ON t.RegionID = r.RegionID
GROUP BY t.RegionID, r.RegionDescription
HAVING SUM(od.Quantity*od.UnitPrice*(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 COUNT(*) AS "Total Number of Orders"
FROM Orders o
WHERE o.Freight > 100.00 AND o.ShipCountry IN ('UK', 'USA')
```

1 SQL Practical Exercise

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 2 *,
FORMAT(od.UnitPrice * od.Quantity, 'C', 'en-gb') AS "Total Cost Before Discount",
FORMAT(od.UnitPrice * od.Quantity * od.Discount, 'C', 'en-gb') AS "Discount to be applied",
FORMAT(od.UnitPrice * od.Quantity * (1 - od.Discount), 'C', 'en-gb') AS "Cost After Discount"
From [Order Details] od
ORDER BY (od.UnitPrice * od.Quantity * od.Discount) DESC
```

2 Exercise 2 – Create Spartans Table (20 marks – 10 each)

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

2.2 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 DATABASE ben_middlehurst

USE ben_middlehurst

CREATE TABLE spartans_table(
    firstName VARCHAR(10),
    lastName VARCHAR(30),
    universityAttended VARCHAR(30),
    courseTaken VARCHAR(30),
    markAchieved VARCHAR(10),
    favouriteColour VARCHAR(20)
)
```

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

2.4 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
('Ben', 'Middlehurst', 'University of Portsmouth', 'Mechanical Engineering', 'First', 'Green'),
('Josh', 'Weeden', 'UCL', 'Fashion','First', 'Sky Blue'),
('Ismail', 'Kadir', 'University of Life', 'Street Smarts', 'First', 'Red'),
('Ben', 'Balls', 'Cambridge', 'Bengineering', 'First', 'Deep Sky Blue'),
('Nathan', 'Johnston', 'Newcastle University', 'Hatem Ben Art', 'First', 'Black & White')
```

1 SQL Practical Exercise

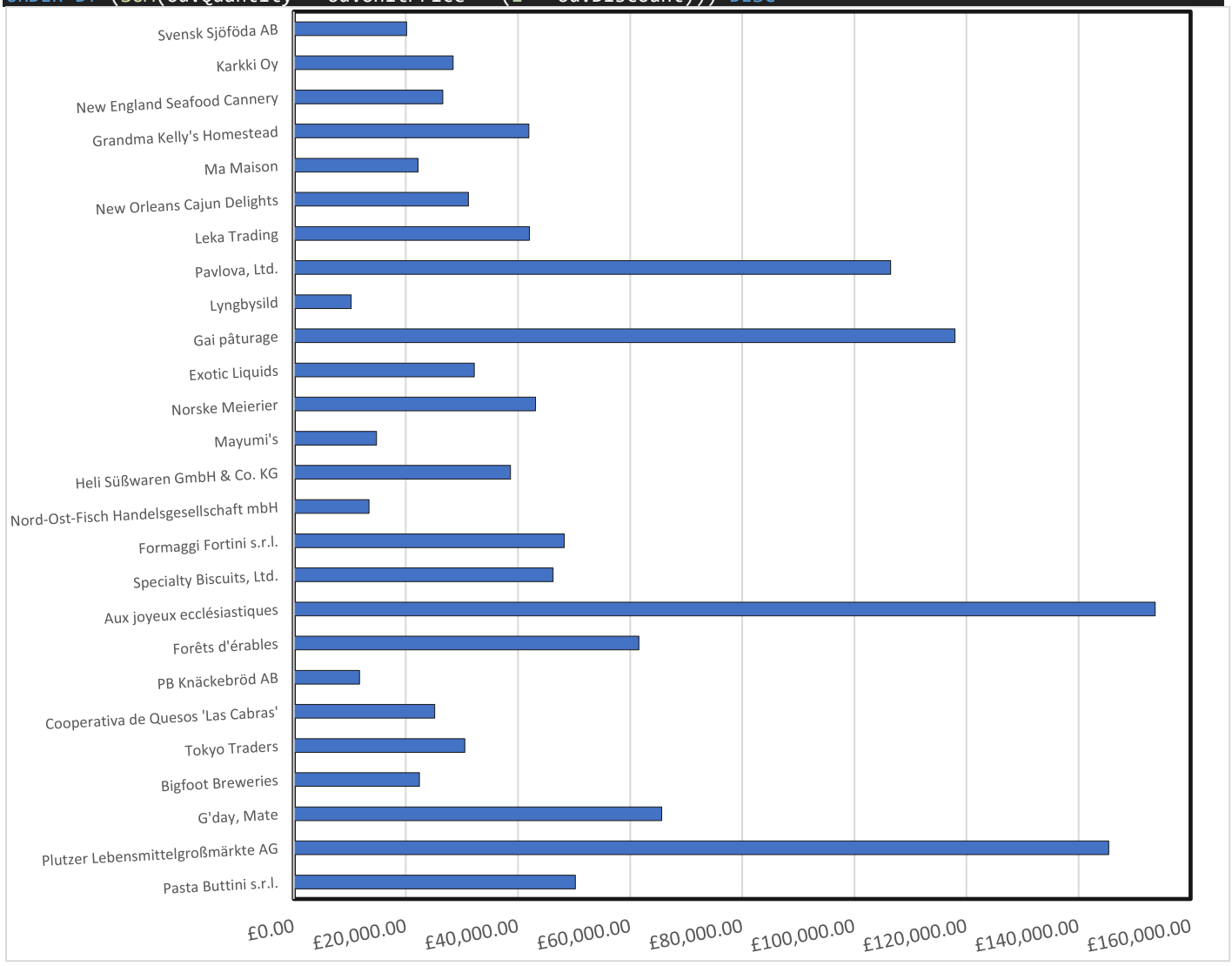
3 Exercise 3 – Northwind Data Analysis linked to Excel (30 marks)

3.1 3.1 List all Employees from the Employees table and who they report to. No Excel required. (5 Marks)

```
SELECT em.EmployeeID, CONCAT(em.FirstName,' ', em.LastName) AS "Employee Name",
    (SELECT CONCAT(e.FirstName,' ', e.LastName)
    FROM Employees e
    WHERE e.EmployeeID = em.ReportsTo) AS "Reports To"
FROM Employees em
```

3.2 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: (5 Marks)

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



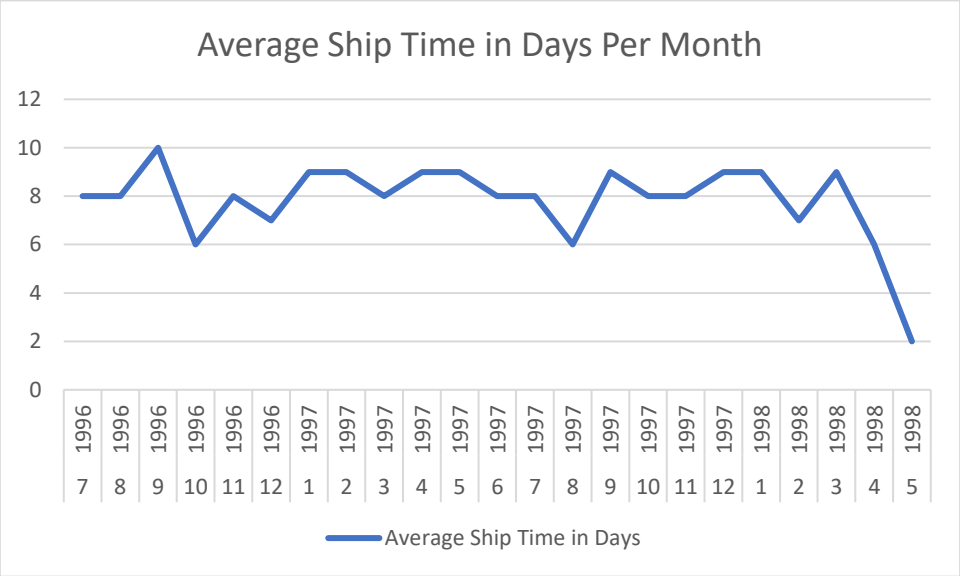
1 SQL Practical Exercise

3.3 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. (10 Marks)

```
SELECT TOP 10
c.CustomerID, c.CompanyName,
FORMAT(SUM(od.Quantity * od.UnitPrice * (1 - od.Discount)), 'C', 'en-gb') AS "Total Orders"
FROM Orders o
INNER JOIN [Order Details] od ON o.OrderID = od.OrderID
INNER JOIN Customers c ON o.CustomerID = c.CustomerID
WHERE YEAR(O.OrderDate) = '1998'
GROUP BY c.CustomerID, c.CompanyName
ORDER BY (SUM(od.Quantity * od.UnitPrice * (1 - od.Discount))) DESC
```

3.4 3.4 Plot the Average Ship Time by month for all data in the Orders Table using a line chart as below. (10 Marks)

```
SELECT MONTH(o.OrderDate) AS "MONTH", YEAR(o.orderdate) AS "YEAR",
AVG(DATEDIFF(d,o.orderdate,o.ShippedDate)) AS "Average Ship Time in Days"
FROM Orders o
GROUP BY YEAR(o.OrderDate), MONTH(o.OrderDate)
ORDER BY 2 ASC, 1 ASC
```



3.5 Standards (10 marks)

Remember to apply all the following standards:

- Use consistent capitalisation and indentation of SQL Statements
- Use concise and consistent table alias names
- Use column aliases to ensure tidy column headings (spaces and consistent capitalisation)
- Concatenate any closely related columns e.g. First Name and Last Name or Address and City etc
- Put comments throughout