



# SQL PROJECT

26-06-2020

## Abstract

A brief project to practise skills learned during the SQL week of the DevOps Stream.

Georgina Bartlett  
[GBartlett@spartaglobal.com](mailto:GBartlett@spartaglobal.com)

## Contents

Contents .....	1
Exercise 1 – General SQL data extraction .....	2
1.1 Write a query that lists all Customers in either Paris or London. Include Customer ID, Company Name and all address fields. ....	2
1.2 List all products stored in bottles. ....	2
1.3 Repeat question above but add in the Supplier Name and Country. ....	2
1.4 Write a SQL Statement that shows how many products there are in each category. Include Category Name in result set and list the highest number first. ....	3
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. ....	3
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. ....	3
1.7 Count how many Orders have a Freight amount greater than 100.00 and either USA or UK as Ship Country. ....	4
1.8 Write a SQL Statement to identify the Order Number of the Order with the highest amount(value) of discount applied to that order. ....	4
Exercise 2 – Data creation and manipulation .....	4
2.1 Write the correct SQL statement to create the following table: ....	4
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. ....	4
2.2 Write SQL statements to add the details of the Spartans in your course to the table you have created. ....	5
Exercise 3 - Extracting the data required for the following charts in Excel .....	5
3.1 List all Employees from the Employees table and who they report to. No Excel required. ....	5
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 .....	6
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. ....	7
3.4 Plot the Average Ship Time by month for all data in the Orders Table using a line chart as below. ....	8

## Exercise 1 – General SQL data extraction

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
      ,CASE
          WHEN c.Region IS NULL THEN CONCAT(c.Address, ', ', c.City, ', ', c.PostalCode, ', ', c.Country)
          ELSE CONCAT(c.Address, ', ', c.City, ', ', c.Region, ', ', c.PostalCode, ', ', c.Country)
        END AS "FullAddress"
FROM Customers c
WHERE c.City IN ('Paris', 'London');
```

1.2 List all products stored in bottles.

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

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

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

1.4 Write a 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(c.CategoryID) AS "NumProductsInCategory"
FROM Products p
INNER JOIN Categories c
      ON p.CategoryID = c.CategoryID
GROUP BY 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 "EmployeeName"
      ,e.City AS "CityOfResidence"
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 r.RegionID
      ,r.RegionDescription
      ,ROUND(SUM(od.UnitPrice*od.Quantity),2) AS "SalesTotals"
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
GROUP BY r.RegionID, r.RegionDescription
HAVING ROUND(SUM(od.UnitPrice*od.Quantity),2) >= 1000000
ORDER BY "SalesTotals" DESC;
```

1.7 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 "FreightTotal"
FROM Orders o
WHERE (o.ShipCountry = 'USA' OR o.ShipCountry = 'UK') AND o.Freight > 100;
```

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

```
SELECT od.OrderID AS "OrderNumber"
,ROUND((od.UnitPrice * od.Quantity * (od.Discount)),2) AS "HighestNetDiscount"
FROM [Order Details] od
WHERE (od.UnitPrice * od.Quantity * (od.Discount)) = (
    SELECT MAX(od2.UnitPrice * od2.Quantity * (od2.Discount)) AS "Discount"
    FROM [Order Details] od2
);
```

## Exercise 2 – Data creation and manipulation

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.

```
IF OBJECT_ID('Spartans Table', 'U') IS NOT NULL
DROP TABLE [Spartans Table];

CREATE TABLE [Spartans Table](
    StudentID INT IDENTITY(1,1)
    ,Title VARCHAR(12) NOT NULL
    ,FirstName VARCHAR(50) NOT NULL
    ,LastName VARCHAR(50) NOT NULL
    ,University VARCHAR(40) DEFAULT NULL
    ,Course VARCHAR(40) DEFAULT NULL
    ,Grade VARCHAR(4) DEFAULT NULL
    ,PRIMARY KEY (StudentID) );
```

## 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](Title, FirstName, LastName, University, Course, Grade)
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', 'Anaïs', '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', '2:1')
,('Miss', 'Ugne', 'Okmanaite', 'Aston University', 'International Business & Management ', '2:1')
,('Mr', 'John', 'Byrne', 'University of Greenwich', 'Computing with Games development', NULL)
,('Mr', 'Daniel', 'Teegan', 'University of Brighton', 'Product Design', '2:2')
,('Mr ', 'Max ', 'Palmer', 'University of Birmingham', 'Ancient History', '2:1');

SELECT *
FROM [Spartans Table]
```

## Exercise 3 - Extracting the data required for the following charts in Excel

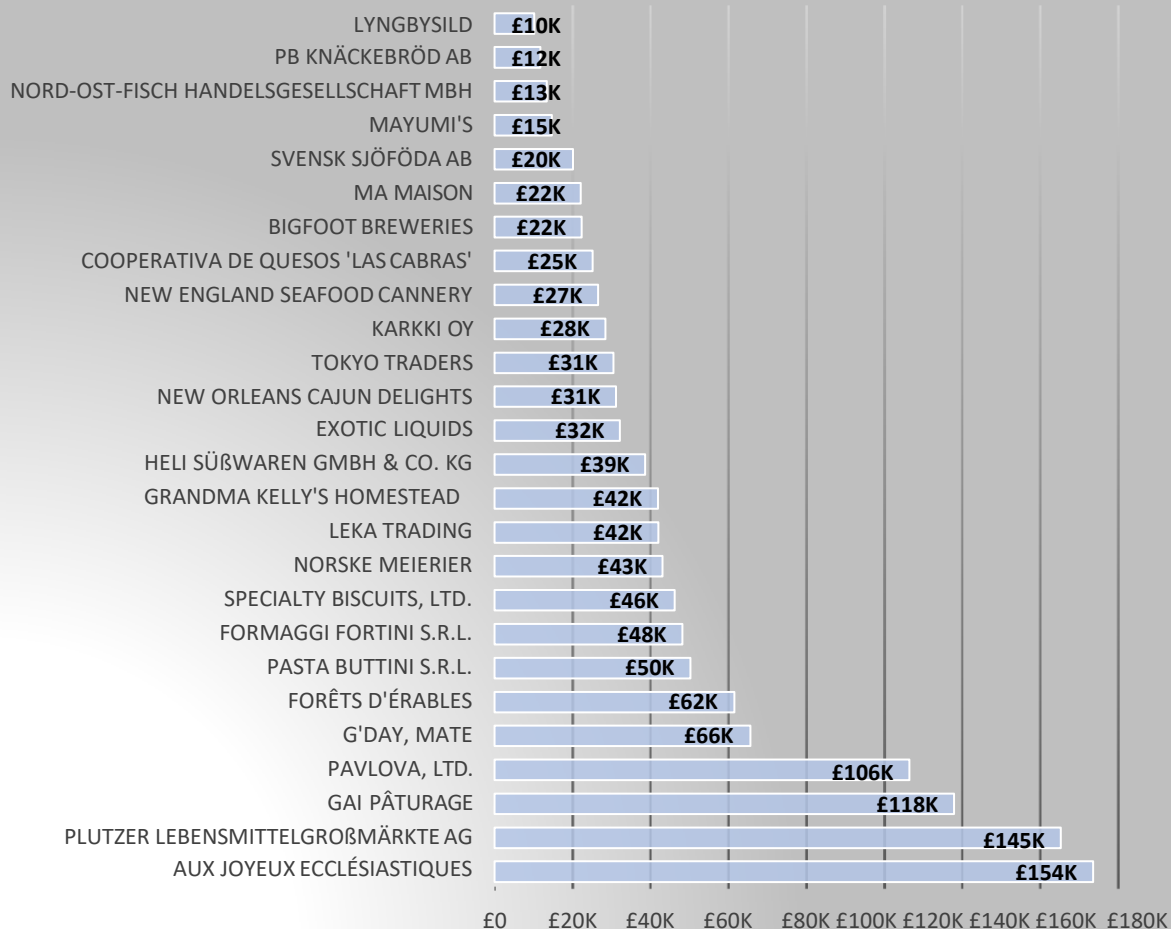
### 3.1 List all Employees from the Employees table and who they report to. No Excel required.

```
SELECT e.EmployeeID
      ,CONCAT(e.FirstName, ' ', e.LastName) AS "EmployeeName"
      ,CONCAT(e2.FirstName, ' ', e2.LastName) AS "ReportsTo"
FROM Employees e
LEFT JOIN Employees e2
ON e2.EmployeeID = e.ReportsTo;
```

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
      ,ROUND(SUM(od.UnitPrice*od.Quantity*(1-od.Discount)),2) AS "TotalSales"
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 SUM(od.UnitPrice*od.Quantity*(1-od.Discount)) > 10000
ORDER BY 2 DESC;
```

### Total Sales (over \$10k) by Company





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
    a.YearOfOrder
    ,a.CompanyName
    ,a.TotalSalePrice
FROM
    (SELECT
        YEAR(o.OrderDate) AS "YearOfOrder"
        ,c.CompanyName AS "CompanyName"
        ,ROUND(SUM(od.UnitPrice*od.Quantity*(1-od.Discount)),2) AS "TotalSalePrice"
    FROM Orders o
    INNER JOIN Customers c
    ON o.CustomerID = c.CustomerID
    INNER JOIN [Order Details] od
    ON o.OrderID = od.OrderID
    GROUP BY YEAR(o.OrderDate)
        ,c.CompanyName) a
WHERE a.YearOfOrder = (SELECT YEAR(MAX(OrderDate)) FROM Orders)
ORDER BY a.YearOfOrder
        ,a.TotalSalePrice DESC
```



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

```
SELECT MONTH(o.OrderDate) AS "OrderMonth"
      ,YEAR(o.OrderDate) AS "OrderYear"
      ,AVG(DATEDIFF(d, o.OrderDate, o.ShippedDate)*1.0) AS "ShipDays"
FROM Orders o
GROUP BY MONTH(o.OrderDate)
      ,YEAR(o.OrderDate)
ORDER BY OrderMonth, OrderYear;
```

