

## SQL practical exercise

--1.1

```
SELECT c.CustomerID, c.CompanyName, c.Address, c.City, c.PostalCode, c.Country
FROM Customers c
WHERE c.City IN('LONDON','Paris')
```

--1.2

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

--1.3

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

--1.4

```
SELECT c.CategoryName AS "Category", COUNT(*) AS "No of Products in Each Category"
FROM Products p LEFT JOIN Categories c ON p.CategoryID = c.CategoryID
GROUP BY p.CategoryID, c.CategoryName
```

--1.5

```
SELECT CONCAT(e.FirstName, ' ', e.LastName) AS "Name", e.City
FROM Employees e
WHERE e.Country = 'UK'
```

--1.6

```
SELECT r.RegionDescription, ROUND(SUM(od.UnitPrice * od.Quantity * (1-
od.Discount)),2) AS "Sales Total"
FROM Orders o
    INNER JOIN [Order Details] od ON od.OrderID = o.OrderID
    LEFT JOIN Employees e ON o.EmployeeID = e.EmployeeID
    LEFT JOIN EmployeeTerritories et ON et.EmployeeID = e.EmployeeID
    LEFT JOIN Territories t ON et.TerritoryID = t.TerritoryID
    LEFT JOIN Region r ON t.RegionID = r.RegionID
GROUP BY r.RegionDescription
HAVING SUM(od.UnitPrice*od.Quantity* (1-od.Discount)) >1000000
```

--1.7

```
SELECT COUNT(*) AS "Orders With Freight Amount Greater Than 100 and Either USA or UK Ship C
ountry"
FROM Orders o
WHERE o.Freight > 100 AND o.ShipCountry IN ('USA','UK')
```

--1.8

```
SELECT TOP 1 od.OrderID, MAX(od.Quantity*od.UnitPrice*od.Discount) AS "Highest Discount value applied"
FROM [Order Details] od
GROUP BY od.OrderID
ORDER BY MAX(od.Quantity*od.UnitPrice*od.Discount) DESC
```

--2

```
CREATE TABLE spartans
(
    person_key INT IDENTITY PRIMARY KEY,
    title VARCHAR(10),
    first_name VARCHAR(10),
    last_name VARCHAR(10),
    university VARCHAR(15),
    course VARCHAR(30),
    mark_achieved NUMERIC
)

INSERT INTO spartans(title, first_name, last_name, university, course, mark_achieved)
VALUES(
    'Mr',
    'Benjamin',
    'Balls',
    'Exeter',
    'Electronic Engineering',
    55
)
```

--3.1

```
SELECT e.FirstName + ' ' + e.LastName AS "Employee Name", er.FirstName + ' ' + er.LastName
AS "Reports To"
FROM Employees e
LEFT JOIN Employees er ON e.ReportsTo = er.EmployeeID
```

--3.2

```
SELECT s.CompanyName, SUM(od.UnitPrice*od.Quantity*(1-od.Discount)) AS "Supplier total sales"
FROM Suppliers s
JOIN Products p ON s.SupplierID = p.SupplierID
JOIN [Order Details] od ON p.ProductID = od.ProductID
JOIN Orders o ON od.OrderID = o.OrderID
GROUP BY s.CompanyName--, ((od.UnitPrice*od.Quantity)*(1-od.Discount))
HAVING SUM((od.UnitPrice*od.Quantity)*(1-od.Discount)) > 10000
ORDER BY SUM(od.UnitPrice*od.Quantity*(1-od.Discount)) DESC
```

--3.3

```
SELECT TOP 10 c.CompanyName, ROUND(SUM((1-  
od.Discount)*od.Quantity * od.UnitPrice),2) AS "sales"  
FROM [Order Details] od  
INNER JOIN Orders o ON o.OrderID = od.OrderID  
INNER JOIN Customers c ON o.CustomerID = c.CustomerID  
GROUP BY c.CompanyName, o.ShippedDate  
HAVING o.ShippedDate > '1997-12-31'  
ORDER BY sales DESC
```

--3.4

```
SELECT YEAR(o.OrderDate) AS "Year",  
       MONTH(o.OrderDate) AS "Month",  
       FORMAT(o.OrderDate, 'MMM-yy') AS "Year-Month",  
       AVG(CAST(DATEDIFF(d, o.OrderDate, o.ShippedDate) AS Decimal(4,2))) AS "Average  
Number of Ship Days" -  
- Might need to format here as not sure if getting the correct answer with rounding  
FROM Orders o  
GROUP BY YEAR(o.OrderDate), MONTH(o.OrderDate), FORMAT(o.OrderDate, 'MMM-yy')  
ORDER BY 1, 2
```