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

SELECT CustomerID, CompanyName, Address

FROM Customers

WHERE City='Paris' OR City='London'

1.2 List all products stored in bottles.

SELECT \*

FROM Products p

WHERE p.QuantityPerUnit LIKE '%bottle%'

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

SELECT p.ProductID AS "Product ID",

p.ProductName AS "Product Name",

s.CompanyName AS "Company Name",

s.Country AS "Country"

FROM Products p

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.CategoryName AS "Category",

COUNT(\*) AS "Number of products in Category"

FROM Products p

JOIN Categories c ON p.CategoryID = c.CategoryID

GROUP BY c.CategoryName

ORDER BY COUNT(\*) 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",

e.City AS "City of Residence"

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.RegionDescription,

ROUND(SUM(od.Quantity \* od.UnitPrice \* (1 - od.Discount)),2) total\_sales

FROM [Order Details] od

INNER JOIN Orders o ON o.OrderID = od.OrderID

INNER JOIN EmployeeTerritories et ON et.EmployeeID = o.EmployeeID

INNER JOIN Territories t ON t.TerritoryID = et.TerritoryID

INNER JOIN Region r ON r.RegionID = t.RegionID

GROUP BY r.RegionDescription

HAVING ROUND(SUM(od.Quantity \* od.UnitPrice),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(OrderID) AS "Orders", o.ShipCountry AS "Ship Country"

FROM Orders o

WHERE ShipCountry IN ('UK', 'USA')

AND o.Freight > 100

GROUP BY o.ShipCountry

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 (od.UnitPrice\*od.Quantity\*od.Discount) AS "Biggest\_amount\_of\_discount", od.OrderID

FROM [Order Details] od

WHERE od.UnitPrice\*od.Discount\*od.Quantity = (SELECT MAX(orr.UnitPrice\*orr.Discount\*orr.Quantity) FROM [Order Details] orr)

ORDER BY Biggest\_amount\_of\_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 DATABASE sidhant\_db

USE sidhant\_db

CREATE TABLE Spartans\_table

(

Title VARCHAR(20),

firstName VARCHAR(20),

lastName VARCHAR(20),

universityAttended VARCHAR(20),

courseTaken VARCHAR(20),

markAchieved INT,

tieGame VARCHAR(20),

)

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

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

SELECT \* FROM Spartans\_table --SELECT everything from a table

INSERT INTO

Spartans\_table(Title, firstName, lastName, universityAttended, courseTaken, markAchieved, tieGame)

VALUES

('Mr.', 'Josh', 'Weeden', 'Oxford', 'All of them', '80', 'Blue-STRONG'),

('Mr.', 'Nathan', 'Johnston', 'Susex', 'IDK', '100', 'STRONG'),

('Mr.', 'Asakar', 'Hussain', 'Middlesex', 'CS','100', 'BOW-TIE'),

('Mr.' , 'Sidhant', 'Khosla', 'Brunel', 'Business','100' , 'FANCY')

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

SELECT CONCAT(e.TitleOfCourtesy, e.FirstName, ' ' ,e.LastName) AS "Employee Name",

CONCAT(id.TitleOfCourtesy, id.FirstName, ' ' ,id.LastName) AS 'Report To'

FROM Employees e

LEFT JOIN Employees id

ON id.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

SELECT s.CompanyName,

ROUND(SUM(od.Quantity\*(od.UnitPrice-od.UnitPrice\*od.Discount)),2) AS "Total Sales"

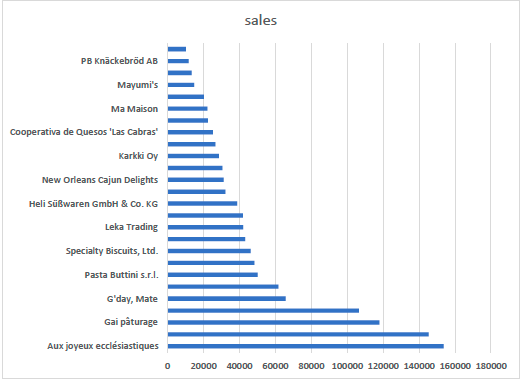
FROM Suppliers s

JOIN Products p ON s.SupplierID = p.SupplierID

JOIN [Order Details] od ON p.ProductID = od.ProductID

GROUP BY s.CompanyName

HAVING SUM(od.Quantity\*(od.UnitPrice-od.UnitPrice\*od.Discount)) > 10000



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.CompanyName AS "Company Name",

ROUND(SUM(od.Quantity\*(od.UnitPrice-od.UnitPrice\*od.Discount)),2) AS "Total Value Of Orders Shipped"

FROM Customers c

JOIN Orders o ON c.CustomerID = o.CustomerID

JOIN [Order Details] od ON o.OrderID = od.OrderID

WHERE YEAR(o.OrderDate)=(SELECT MAX(YEAR(oo.OrderDate)) FROM Orders oo)

GROUP BY c.CompanyName

ORDER BY "Total value of orders shipped" DESC

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

SELECT AVG(DATEDIFF(DAY,o.OrderDate,o.ShippedDate)) AS "Average Ship Time",

FORMAT(o.OrderDate,'MMM-yyyy') AS "Month"

FROM Orders o

GROUP BY FORMAT(o.OrderDate,'MMM-yyyy'),DATEPART(YEAR, o.OrderDate), DATEPART(MONTH, o.OrderDate)

ORDER BY DATEPART(YEAR, o.OrderDate),DATEPART(MONTH, o.OrderDate)

