**DATABASE COMPULSORY TASK**

Make sure you’ve met **“Useful preconditions”** provided thereto.

Please use **“deliveries” database** to perform the following task.

Graphical user interface, text, application, chat or text message

Description automatically generated

**Database diagram:**

*Graphical user interface, application, table, Excel

Description automatically generated*

*The results of your work should be SQL-queries inserted in this document.*

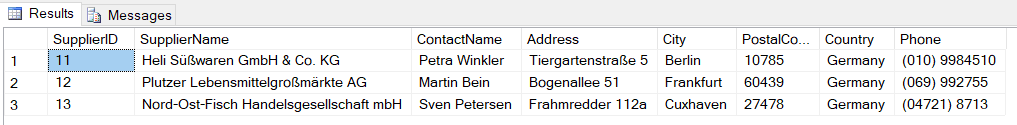
Use the Database diagram and screenshots of the expected results to each SQL-query to adjust them. **Ensure that Columns in your resulting table correspond to the expected results provided.**

**Tasks and expected results:**

**1.** Select all the information about Suppliers from Germany

**SQL Query:**

SELECT \* FROM Suppliers WHERE Country LIKE '%Germany';

****

**2.** Select all the information about Products with CategoriesID “1” and “6” and sort the results by Price

**SQL Query:**

SELECT \* FROM products

WHERE CategoryID IN (1, 6)

ORDER BY Price;

Table

Description automatically generated

**3.** Select all Seafood products with the Price up to 20 inclusive

*Use ‘Product’ table, but do not hesitate to take a look into other tables for any additional info.*

**SQL Query:**

SELECT ProductName, CategoryID, Price FROM products WHERE CategoryID = 8 AND Price <= 20;

Graphical user interface, application

Description automatically generated

**4.** Select all the Employees who has the First Name with the 3rd  letter “N” and whose ID number is between 2 and 10

**SQL Query:**

SELECT EmployeeID, FirstName, LastName, BirthDate FROM Employees

WHERE FirstName LIKE '\_\_n%'

AND EmployeeID BETWEEN 2 AND 10;

Graphical user interface, text, application

Description automatically generated

**5.** Select the product with the maximum price and name the resulting column “MAXPrice”

**SQL Query:**

SELECT ProductName, Price AS MAXPrice

FROM Products

WHERE PRICE = (SELECT MAX(Price) FROM Products);

Graphical user interface, application, table

Description automatically generated

**\*6.** Show the average price of the products delivered in jars, round to 2 decimal places and name the resulting column “AVGPrice”

**SQL Query:**

SELECT ROUND(AVG(Price), 2) AS AVGPrice

FROM Products

WHERE Unit LIKE '%jar%';

Graphical user interface, application

Description automatically generated

**7.** Show the number of customers who do not live in the USA and Spain. Name the resulting column “FinalResult”

**SQL Query:**

SELECT COUNT(CustomerName) AS FinalResult

FROM Customers

WHERE Country NOT IN ('USA', 'Spain');

Graphical user interface, application

Description automatically generated with medium confidence

**8.** Show a list of the customers whose name ends with “s”, from the country which name does not contain “U” letter and starts with “B”. Sort the list by the customers’ name from Z to A

**SQL Query:**

SELECT CustomerName, Address, Country

FROM Customers

WHERE CustomerName LIKE '%s' AND Country LIKE 'B%' AND Country NOT LIKE '%u%'

ORDER BY CustomerName DESC;

Graphical user interface, text, application, table

Description automatically generated

**9.** Count the number of customers in each country and name the resulting column “ClientNumber”

**SQL Query:**

SELECT Country, COUNT(CustomerName) AS ClientNumber

FROM Customers

GROUP BY Country;

Table

Description automatically generated with low confidence

**10.** Show the list of employees who was born in winter till 1963. Sort the list by EmployeeID in descending order.

**SQL Query:**

SELECT FirstName, LastName, BirthDate FROM Employees

WHERE BirthDate < '1963-01-01' AND MONTH(BirthDate) IN (01,02, 12);

ORDER BY EmployeeID DESC;

Table

Description automatically generated

**11.** Count all the customers per city and show the list of cities with the number of customers is more than 1. Name the resulting column “CustomerNumber”

**SQL Query:**

SELECT Country, City, COUNT(CustomerName) AS CustomerNumber

FROM Customers

GROUP BY COUNTRY, City

HAVING COUNT(CustomerName)>1;

Table

Description automatically generated

**12.** Show the list of products supplied by “Pavlova, Ltd.” with the price higher than 30.

**SQL Query:**

SELECT SupplierName, ProductName, Price

FROM Products

INNER JOIN Suppliers

ON Products.SupplierID=Suppliers.SupplierID

WHERE SupplierName = 'Pavlova, Ltd.' AND price>30;

Table

Description automatically generated

**13.** Show the list of the shippers who sent orders to the Customer with ID # 46, but have the Customer Name displayed instead of its ID number

**SQL Query:**

SELECT ShipperName, OrderDate, CustomerName

FROM Shippers

INNER JOIN Orders

ON Shippers.ShipperID=Orders.ShipperID

INNER JOIN Customers

ON Orders.CustomerID=Customers.CustomerID

WHERE Customers.CustomerID = 46;

Table

Description automatically generated

**14.** Count the total price of all the products by Categories

**SQL Query:**

SELECT CategoryName, SUM(Price) AS TotalPrice

FROM Categories

INNER JOIN Products

ON Categories.CategoryID=Products.CategoryID

GROUP BY CategoryName;

Graphical user interface, application, table

Description automatically generated

**15.** Select the shipper who delivers the most expensive product

**SQL Query:**

SELECT SupplierName, ProductName, Unit, Price

FROM Suppliers

INNER JOIN Products

ON Suppliers.SupplierID=Products.SupplierID

WHERE Price = (SELECT MAX(Products.Price) FROM Products);

Table

Description automatically generated