**DATABASE ADDITIONAL TASK**

Make sure you’ve followed **“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 products with the price higher than 50 and *not from* categories with ID # 2, 3, 5 and 7

**SQL Query:**

SELECT ProductName, CategoryID, Price

FROM Products

WHERE CategoryID NOT IN(2,3,5,7) AND Price>50

ORDER BY CategoryID;

Table

Description automatically generated

**2.** Select orders with quantity of products 80 and more and show only 7 first results in descending order.

**SQL Query:**

SELECT TOP (7) OrderID, ProductName, Quantity

FROM Orderdetails

INNER JOIN Products

ON Products.ProductID=Orderdetails.ProductID

WHERE Quantity>=80

ORDER BY Quantity DESC;

Graphical user interface, application, table, Excel

Description automatically generated

**3.\*** Select customers and change the results in the Summary column “Language” depending on the City condition: “English” for London, “French” for Paris and “I don’t know” for Graz and Sevilla cities.

**SQL Query:**

SELECT CustomerName, Address, City,

IIF(City= 'London', 'English', IIF(City='Paris', 'French', IIF(City IN('Graz', 'Sevilla'), 'I don’t know', NULL))) AS Language

FROM Customers

WHERE City IN('London', 'Paris', 'Graz', 'Sevilla')

ORDER BY Language;

Graphical user interface, application, table, Excel

Description automatically generated

**4.** Select the first 15 customers and show their full address in 1 column.

**SQL Query:**

SELECT TOP (15) CustomerName, CONCAT(Country, ',' , city, ',', address) AS 'Full Address'

FROM Customers;

Text

Description automatically generated

**5.** Show all the cities and their countries where customers and suppliers from.

**Note!** *Total number of cities in the resulting table is* ***94*** *and they are not repeated.*

**SQL Query:**

SELECT City, Country FROM Customers UNION SELECT City, Country FROM Suppliers;

Table

Description automatically generated

**6.** Select information about orders that were delivered to the following address: “Austria, Graz, Kirchgasse 6” in 1996.

**SQL Query:**

SELECT OrderID, OrderDate, CustomerName, CONCAT (Country, ', ' , city, ', ', address) AS 'Full Address'

FROM Orders

INNER JOIN Customers

ON Orders.CustomerID=Customers.CustomerID

WHERE CONCAT(Country, ', ' , city, ', ', address) = 'Austria, Graz, Kirchgasse 6' AND YEAR(OrderDate) ='1996';

Table

Description automatically generated

**7.** Display the information for the order submitted on the 17th of January 1997 with the following details:

- customer name

- the name of the employee who processed the order

- the name of the shipper involved in this transaction

**SQL Query:**

SELECT OrderDate, CustomerName, CONCAT(Employees.FirstName,' ', Employees.LastName) AS EmployeeName, ShipperName

FROM Orders

INNER JOIN Customers

ON Orders.CustomerID= Customers.CustomerID

INNER JOIN Employees

ON Orders.EmployeeID=Employees.EmployeeID

INNER JOIN Shippers

ON Orders.ShipperID=Shippers.ShipperID

WHERE OrderDate = '1997-01-17';

Graphical user interface, application, table

Description automatically generated

**8.** Show all the products that consist of 5 words from the orders submitted in July 1997.

**SQL Query:**

SELECT Orders.OrderID AS OrderID, OrderDate, ProductName

FROM Orders

INNER JOIN Orderdetails

ON Orders.OrderID=Orderdetails.OrderID

INNER JOIN Products

ON Orderdetails.ProductID= Products.ProductID

WHERE ProductName LIKE '% % % % %' AND OrderDate BETWEEN '1996-06-30' AND '1996-08-01';

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

Description automatically generated

**9.** Show product names that was ordered by the customer “Hanari Carnes”. Categories of these products should consist of 7 symbols.

**SQL Query:**

SELECT CustomerName AS CUSTOMER, ProductName

FROM Customers

INNER JOIN Orders

ON Customers.CustomerID=Orders.CustomerID

INNER JOIN Orderdetails

ON Orders.OrderID=Orderdetails.OrderID

INNER JOIN Products

ON Orderdetails.ProductID=Products.ProductID

INNER JOIN Categories

ON Products.CategoryID=Categories.CategoryID

WHERE Categories.CategoryName LIKE '\_\_\_\_\_\_\_' AND CustomerName LIKE 'Hanari Carnes';

Table

Description automatically generated

**10.** Select the shipper who performed the greatest number of shippings. Show the number of performed shippings as well.

**SQL Query:**

SELECT TOP (1) COUNT(OrderID) AS TotalNumber, ShipperName

FROM Orders

INNER JOIN Shippers

ON Orders.ShipperID=Shippers.ShipperID

GROUP BY ShipperName

ORDER BY TotalNumber DESC;

Table

Description automatically generated

**11.** Insert 2 new employees (“Employees” table) with the following info:

1) Stepan Green, 09.07.1958, his description: “*Stepan comes from Belarusian. He graduated from BSU and spent 8 years in China. He is fluent in Chineeese and Inglish.*”

2) Arthur King, 17.02.1969, his description: “*Arthur has expeirience in maintaining project documentation and risk management.*”

**Note!** *Do not correct any given information about new employees. That’s critically important for the next task.*

**SQL Query:**

INSERT INTO Employees

VALUES (11, 'Green','Stepan', '09.07.1958', 'Stepan comes from Belarusian. He graduated from BSU and spent 8 years in China. He is fluent in Chineeese and Inglish.'),

(12, 'King', 'Arthur', '02.17.1969', 'Arthur has expeirience in maintaining project documentation and risk management.');

SELECT \* FROM Employees;

A picture containing text

Description automatically generated

**12.** Correct the mistakes in “Employee” table made when adding new Employees.

**SQL Query:**

UPDATE Employees

SET Notes = 'Stepan comes from Belarus. He graduated from BSU and spent 8 years in China. He is fluent in Chineese and English.'

WHERE EmployeeID = 11;

UPDATE Employees

SET Notes = 'Arthur has experience in maintaining project documentation and risk management.'

WHERE EmployeeID = 12;

SELECT \* FROM Employees;

A picture containing text

Description automatically generated

**13.** Delete *Adam West* and *Nancy Davolio* from the list of employees.

**SQL Query:**

DELETE FROM Employees

WHERE LastName IN('West', 'Davolio') AND FirstName IN('Adam', 'Nancy');

SELECT \* FROM Employees;

Table

Description automatically generated