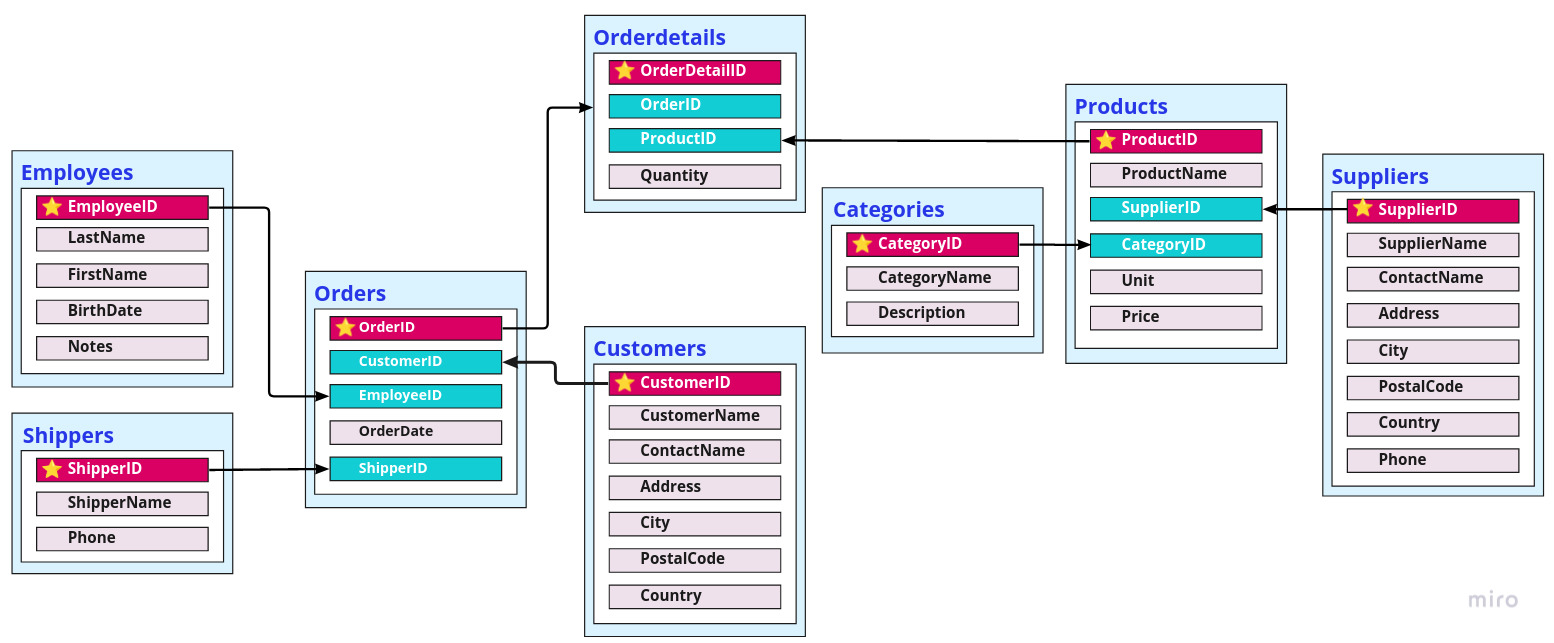
**Database diagram:**

**

**1.** Select the 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 Price > 50 AND CategoryID NOT IN (2,3,5,7)**

**ORDER BY CategoryID;**

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

**SQL Query:**

**SELECT TOP(7) Orders.OrderID, ProductName, Quantity**

**FROM Orders**

**JOIN OrderDetails**

**ON Orders.OrderID = OrderDetails.OrderID**

**JOIN Products**

**ON Orderdetails.ProductID = Products.ProductID**

**WHERE Quantity >= 80**

**ORDER BY Quantity DESC;**

**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,**

**CASE**

**WHEN City = 'London' THEN 'English'**

**WHEN City = 'Paris' THEN 'French'**

**WHEN City IN ('Graz', 'Sevilla') THEN 'I don’t know'**

**END AS 'Language'**

**FROM Customers**

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

**ORDER BY Language;**

**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;**

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

**SQL Query:**

**SELECT City, Country FROM Customers**

**UNION**

**SELECT City, Country FROM Suppliers;**

**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**

**JOIN Customers**

**ON Customers.CustomerID = Orders.CustomerID**

**WHERE CONCAT(Country,', ', City, ', ', Address) = 'Austria, Graz, Kirchgasse 6' AND YEAR(OrderDate) = 1996**

**ORDER BY OrderID;**

**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(FirstName, ' ', LastName) AS 'EmployeeName', ShipperName**

**FROM Orders**

**JOIN Customers**

**ON Customers.CustomerID = Orders.CustomerID**

**JOIN Employees**

**ON Orders.EmployeeID = Employees.EmployeeID**

**JOIN Shippers**

**ON Shippers.ShipperID = Orders.ShipperID**

**WHERE OrderDate = '1997-01-17';**

**8.** Show all the products that consist of 5 words from the orders submitted in July 1996.

**SQL Query:**

**SELECT Orderdetails.OrderID, OrderDate, ProductName**

**FROM Orders**

**JOIN Orderdetails**

**ON Orders.OrderID = Orderdetails.OrderID**

**JOIN Products**

**ON Orderdetails.ProductID = Products.ProductID**

**WHERE ProductName LIKE '% % % % %' AND YEAR(OrderDate) = 1996 AND MONTH(OrderDate) = 7;**

**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 Products**

**JOIN Orderdetails**

**ON Products.ProductID = Orderdetails.ProductID**

**JOIN Orders**

**ON Orders.OrderID = Orderdetails.OrderID**

**JOIN Customers**

**ON Orders.CustomerID = Customers.CustomerID**

**JOIN Categories**

**ON Products.CategoryID = Categories.CategoryID**

**WHERE CustomerName = 'Hanari Carnes' AND CategoryName LIKE '\_\_\_\_\_\_\_';**

**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**

**JOIN Shippers**

**ON Orders.ShipperID = Shippers.ShipperID**

**GROUP BY Shippers.ShipperID, ShipperName**

**ORDER BY TotalNumber DESC;**

**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.*”

**SQL Query:**

**INSERT INTO Employees**

**VALUES**

**('11', 'Green', 'Stepan', '1958-07-09', 'Stepan comes from Belarusian. He graduated from BSU and spent 8 years in China. He is fluent in Chineeese and Inglish.'),**

**('12', 'King', 'Arthur', '1969-02-17', 'Arthur has expeirience in maintaining project documentation and risk management.');**

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

**SQL Query:**

**UPDATE Employees**

**SET Notes = CASE EmployeeID**

**WHEN 11 THEN 'Stepan comes from Belarus. He graduated from BSU and spent 8 years in China. He is fluent in Chinese and English.'**

**WHEN 12 THEN 'Arthur has experience in maintaining project documentation and risk management.'**

**END**

**WHERE EmployeeID IN (11, 12);**

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

**SQL Query:**

**DELETE FROM Employees**

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

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

**SQL Query:**

**SELECT \* FROM Suppliers**

**WHERE Country = 'Germany';**

**15.** 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;**

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

**SQL Query:**

**SELECT ProductName, CategoryID, Price**

**FROM Products**

**WHERE CategoryID = (SELECT CategoryID FROM Categories WHERE CategoryName = 'Seafood')**

**AND Price <=20;**

**17.** 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;**

**18.** 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);**

**19.** 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 '%jars%';**

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

**SQL Query:**

**SELECT COUNT(CustomerID) AS 'FinalResult'**

**FROM Customers**

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

**21.** 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;**

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

**SQL Query:**

**SELECT Country, COUNT(CustomerID) AS 'ClientNumber'**

**FROM Customers**

**GROUP BY Country;**

**23.** 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 Month(BirthDate) IN (1, 2, 12) AND YEAR(BirthDate) < 1963**

**ORDER BY EmployeeID DESC;**

**24.** 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(CustomerID) AS 'CustomerNumber'**

**FROM Customers**

**GROUP BY City, Country**

**HAVING COUNT(CustomerID) > 1**

**ORDER BY City;**

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

**SQL Query:**

**SELECT SupplierName, ProductName, Price**

**FROM Suppliers**

**JOIN Products**

**ON Suppliers.SupplierID = Products.SupplierID**

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

**26.** 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**

**JOIN Orders**

**ON Orders.ShipperID = Shippers.ShipperID**

**JOIN Customers**

**ON Customers.CustomerID = Orders.CustomerID**

**WHERE Customers.CustomerID = 46;**

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

**SQL Query:**

**SELECT CategoryName, SUM(Price) AS 'TOTAL PRICE'**

**FROM Products**

**JOIN Categories**

**ON Products.CategoryID = Categories.CategoryID**

**GROUP BY Categories.CategoryName;**

**28.** Select the supplier who delivers the most expensive product in the bottles.

**SQL Query:**

**SELECT SupplierName, ProductName, Unit, Price**

**FROM Suppliers**

**JOIN Products**

**ON Suppliers.SupplierID = Products.SupplierID**

**WHERE Price = (SELECT MAX(Price) FROM Products WHERE Unit LIKE '%bottles%');**