

Напишите SQL SELECT запросы, выполняющие условия заданий, описанных ниже. Каждое задание должно сопровождаться SELECT-запросом

Database: https://www.w3schools.com/sql/trysql.asp?filename=trysql_select_columns

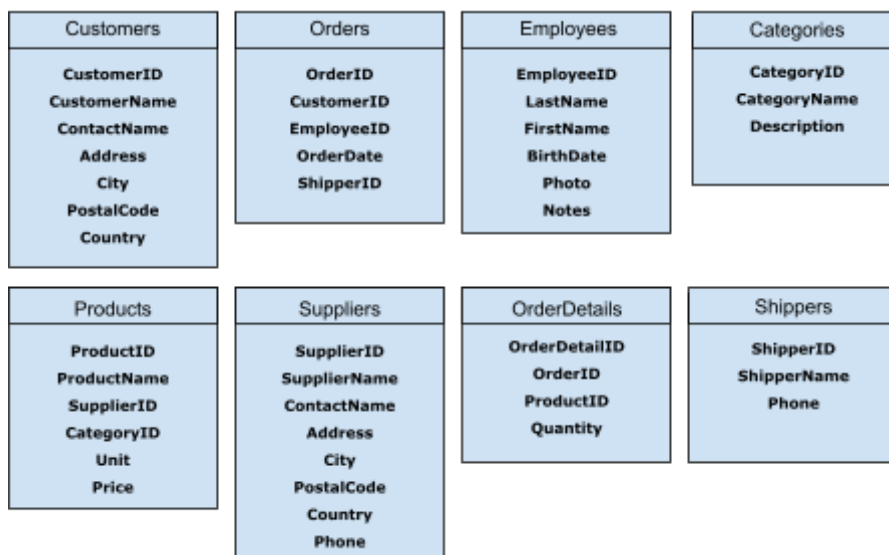


Таблица Customers:

- Вывести все столбцы из таблицы Customers (команда SELECT со списком конкретных полей и без указания списка (символ “*”). **2 sql-команды (через * и через перечисление столбцов)**
 - SELECT * FROM Customers**
 - SELECT CustomerID, CustomerName, ContactName, Address, City, PostalCode, Country FROM [Customers]**
- Выбрать **ContactName, City** из таблицы Customers
 - SELECT ContactName, City FROM Customers
- Выбрать всех покупателей (вывести все столбцы) из таблицы Customers живущих в Испании(Spain) (команда SELECT с условием). 1 sql-команда.
 - SELECT * FROM Customers
Where Country = 'Spain'
- Выбрать **CustomerName, Address and Customer_ID** из таблицы Customers, живущих в Берлине (Berlin), Лондоне (London) и Мадриде (Madrid). Выполнить 2 способами (через OR и IN операторы)
 - SELECT CustomerName, Address, CustomerID FROM Customers
Where City = 'Berlin' or city = 'London' or City = 'Madrid'
 - SELECT CustomerName, Address, CustomerID FROM Customers
Where City In ('Berlin', 'London', 'Madrid')

5. Показать уникальные города (**City**) и **PostalCode** из таблицы Customers по стране США (USA) (команда SELECT с условием). 1 sql-команда.

```
SELECT Distinct City, PostalCode FROM Customers  
Where Country = 'USA'
```

Таблица Employees:

6. Выбрать всех покупателей (**FirstName, LastName, BirthDate**) из таблицы Employees имеющих заметки (Notes) и родившихся после первого июля 1963 года (команда SELECT с условием анализа на NULL-значение). 1 sql-команда.

```
SELECT FirstName, LastName, BirthDate FROM Employees  
Where BirthDate > '1963-07-01' and Notes is not null
```

7. Показать сотрудников (**FirstName, LastName, BirthDate**) из таблицы Employees, которые родились до 30.05.1960. Отсортировать результаты по имени (команда SELECT с условиями, сортировки) . 1 sql-команда.

```
SELECT FirstName, LastName, BirthDate FROM Employees  
Where BirthDate < '1960-05-30'  
Order By FirstName
```

8. Выбрать самого молодого сотрудника(FirstName) из таблицы Employees и его дату рождения(BirthDate). 1 sql-команда.

```
SELECT FirstName, MAX(BirthDate) FROM Employees
```

Таблица Products:

9. Выбрать **ProductID, ProductName, Price, SupplierID** из таблицы Products, где идентификатор поставщика(SupplierID) равен 1, 3 и 5 и стоимость больше 18 (команда SELECT с несколькими условиями). 1 sql-команда.

```
SELECT ProductID, ProductName, Price, SupplierID FROM Products  
Where SupplierID in (1,3,5) And price > 18
```

10. Выбрать все продукты (**ProductID, ProductName**) название которых(ProductName) начинается на С, из таблицы Products, цена которых от 22 до 263.5 (команда SELECT с несколькими условиями). 1 sql-команда.

```
SELECT ProductID, ProductName, FROM Products  
Where ProductName Like "C%" and Price between 22 and 263.5
```

11. Показать продукты (**ProductName**) из таблицы Products, у которых в названии присутствует буква ö (поиск по шаблону с использованием LIKE) . 1 sql-команда.

```
SELECT ProductName FROM Products
Where ProductName Like '%ö%'
```

JOINS

12. Показать имена сотрудников(**FirstName, LastName**) из таблицы Employees, и соответствующий им номер заказа (**Order ID**) из таблицы Orders (простой JOIN). 1 sql-команда.

```
SELECT Employees.FirstName, Employees.LastName, Orders.OrderId
FROM Employees
JOIN Orders ON Orders.EmployeeId = Employees.EmployeeId
```

13. Показать названия стран (**Country**) и городов (**City**) из таблицы Customers, и названия продуктов (**ProductName**) из таблицы Products, и название категории(**CategoryName**) из таблицы Categories (Несколько простых JOINS). 1 sql-команда.

```
SELECT Customers.City, Customers.Country, Products.ProductName,
Categories.CategoryName
FROM Customers
JOIN Orders ON Customers.CustomerID = Orders.CustomerID
JOIN OrderDetails ON Orders.OrderID = OrderDetails.OrderID
JOIN Products ON OrderDetails.ProductID = Products.ProductID
JOIN Categories ON Products.CategoryID = Categories.CategoryID
```

14. Выбрать покупателей(**CustomerName**) и их адрес(**Address**) из таблицы Customers, имя(**FirstName**) и фамилию(**LastName**) сотрудника из таблицы Employees и имя поставщика(**SupplierName**) из таблицы Suppliers, отфильтровать по продуктам из заказа, цена которых не превышает 25 (Команда SELECT с условием, Несколько JOINS). 1 sql-команда.

```
SELECT Customers.CustomerName, Customers.Address, Employees.FirstName,
Employees.LastName, Suppliers.SupplierName
FROM Customers
JOIN Orders ON Customers.CustomerID = Orders.CustomerID
JOIN Employees ON Orders.EmployeeID = Employees.EmployeeID
JOIN OrderDetails ON Orders.OrderID = OrderDetails.OrderID
JOIN Products ON OrderDetails.ProductID = Products.ProductID
JOIN Suppliers ON Products.SupplierID = Suppliers.SupplierID
Where Products.Price <=25
```

15. Выбрать имена контактов(**ContactName**) из таблицы Customers, их заказы(**OrderID**) из таблицы Orders, продукты и цену из таблицы Products, отфильтровать только те имена контактов, в чьем заказе количество продуктов(**Quantity**) не больше 30 и **Unit** содержит «pkg» (Команда SELECT с условиями, Несколько JOINS) . 1 sql-команда

```
SELECT Customers.ContactName, Orders.OrderID, Products.Price,  
Products.ProductName FROM Customers  
JOIN Orders ON Customers.CustomerID = Orders.CustomerID  
JOIN OrderDetails ON Orders.OrderID = OrderDetails.OrderID  
JOIN Products ON OrderDetails.ProductID = Products.ProductID  
WHERE OrderDetails.Quantity <= 30 AND Products.Unit LIKE '%pkg%'
```

16. Выбрать всю информацию о продуктах из таблицы Products, их категорию (**CategoryName**) и описание категории (**Description**) из таблицы Categories и имя поставщика (**SupplierName**), который их поставляет, из таблицы Suppliers, отфильтровать только те продукты, поставщики которых находятся в Германии (Germany) или США(USA). (символ “*” для указанной таблицы, команда SELECT с условием, Несколько простых JOINS) . 1 sql-команда

```
SELECT Products.*, Categories.CategoryName, Categories.Description,  
Suppliers.SupplierName  
FROM Products  
JOIN Suppliers ON Products.SupplierID = Suppliers.SupplierID  
JOIN Categories ON Products.CategoryID = Categories.CategoryID  
WHERE Suppliers.Country IN ('USA', 'Germany')
```

17. Показать всех покупателей(**CustomerName**), их заказы(**OrderID**), которые они совершили, и названия продуктов(**ProductName**), которые входят в заказ. Отсортировать по названию продуктов (Несколько простых JOINS) 1 sql-команда.

```
SELECT Customers.CustomerName, Orders.OrderID, Products.ProductName  
FROM Customers  
JOIN Orders ON Customers.CustomerID = Orders.CustomerID  
JOIN OrderDetails ON Orders.OrderID = OrderDetails.OrderID  
JOIN Products ON OrderDetails.ProductID = Products.ProductID  
ORDER BY Products.ProductName
```

18. Показать все продукты(**ProductName**) и поставщиков(**SupplierName**), которые их поставляют, и среднюю стоимость каждого продукта (LEFT JOINS, групповая функция AVG). 1 sql-команда.

```
SELECT Products.ProductName, Suppliers.SupplierName, AVG(Products.Price)  
FROM Suppliers  
Left JOIN Products ON Suppliers.SupplierID = Products.SupplierID  
GROUP BY ProductName, SupplierName
```

19. Подсчитать суммарную цену каждого заказа (AS 'Total Price') и номер заказа(OrderID). 1 sql-команда.

```
SELECT Orders.OrderID, SUM(Products.Price*OrderDetails.Quantity) AS Total_Price
FROM Orders
JOIN OrderDetails ON Orders.OrderID = OrderDetails.OrderID
JOIN Products ON OrderDetails.ProductID = Products.ProductID
GROUP BY Orders.OrderID
```

20. Показать заказы(**OrderID**), в которые входят самые дорогостоящие продукты (исходя из их цены(**Price**)) ИЛИ которые были оформлены очень давно (исходя из их даты оформления (**OrderDate**)) (Join, MAX) . 1 sql-команда.

```
SELECT Orders.OrderID, Products.Price, Orders.OrderDate
FROM Orders
JOIN OrderDetails ON Orders.OrderID = OrderDetails.OrderID
JOIN Products ON OrderDetails.ProductID = Products.ProductID
WHERE Products.Price = (SELECT MAX(Price) FROM Products) OR
Orders.OrderDate = (SELECT MIN(OrderDate) FROM Orders)
GROUP BY Orders.OrderID
```