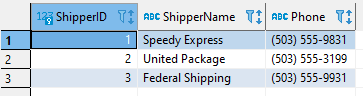
**SQL запросы и скриншоты:**

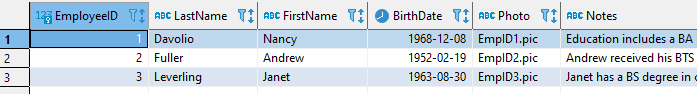
1. Выбрать все строки из таблицы c перевозчиками.

**SELECT** \* **FROM** Shippers;



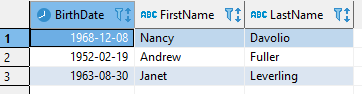
2. Выбрать первые 3 строки из таблицы c сотрудниками.

**SELECT** \* **FROM** Employees **LIMIT** 3;



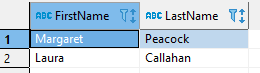
3. Из таблицы сотрудников выбрать все имена, фамилии, дни рождения в следующем порядке: BirthDate, FirstName, LastName, количество строк в выборке ограничить 3-мя.

**SELECT** BirthDate, FirstName, LastName **FROM** Employees **LIMIT** 3;



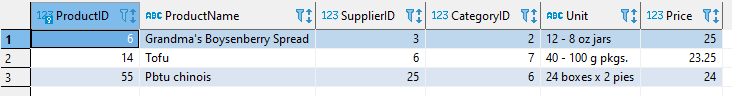
4. Выбрать имена и фамилии сотрудников, родившихся в 1958 году.

**SELECT** FirstName, LastName **FROM** Employees **WHERE** **YEAR**(BirthDate) = 1958;



5. Выбрать все товары с ценой от 23 до 25.

**SELECT** \***FROM** Products **WHERE** Price **BETWEEN** 23 **AND** 25;



6. Найти товары с минимальной ценой.

**SELECT** \* **FROM** Products

**WHERE** Price = (**SELECT** **MIN**(Price) **FROM** Products);



7. Найти товары с максимальной ценой.

**SELECT** \* **FROM** Products

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



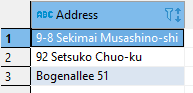
8. Выбрать все товары, у которых Unit '10 pkgs.'.

**SELECT** \* **FROM** Products **WHERE** Unit = '10 pkgs.';



9. Выбрать адреса поставщиков, которые проживают в одном из городов: Tokyo, Frankfurt, Osaka.

**SELECT** Address **FROM** Suppliers **WHERE** City **IN**('Tokyo', 'Frankfurt', 'Osaka');



10. Выбрать название товаров начинающихся с буквы “G”, у которых цена больше 37.

**SELECT** ProductName **FROM** Products **WHERE** ProductName **LIKE** 'G%' **AND** Price > 37;



11. Вывести список стран начинающихся на S и состоящих из 5 букв, из которых есть поставщики.

**SELECT** Country **FROM** Suppliers **WHERE** Country **LIKE** 'S\_\_\_\_';



12. Вывести сумму всех товаров, в названии которых содержится ”od”, столбец назвать Summ.

**SELECT** **SUM**(Price) **AS** Summ **FROM** Products **WHERE** ProductName **LIKE** '%od%';



13. Вывести среднюю сумму товаров, поставляемых в бутылках, округлив до 2-х знаков после запятой, столбец назвать Summ.

**SELECT** **ROUND**(**AVG**(Price), 2) **AS** Summ **FROM** Products **WHERE** Unit **LIKE** '%bottle%';



14. Найти количество клиентов, которые НЕ проживают в Франции и Германии, столбец назвать Countt.

**SELECT** **COUNT**(CustomerID) **AS** Countt **FROM** Customers **WHERE** Country **NOT** **IN** ('France', 'Germany');



15. Вывести имена сотрудников, родившихся после 01.01.1968 года. Отсортировать результат по имени.

**SELECT** FirstName **FROM** Employees **WHERE** BirthDate > '1968-01-01'

**ORDER** **BY** FirstName;



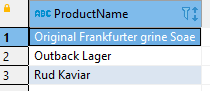
16. Выбрать названия товаров, у которых Price = 13 или 15 и отсортировать по возрастанию, использовать Select команды с объединением результатов через UNION.

**SELECT** ProductName **FROM** Products **WHERE** Price = 13

**UNION**

**SELECT** ProductName **FROM** Products **WHERE** Price = 15

**ORDER** **BY** ProductName;



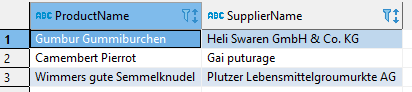
17. Показать имена товаров, в названии которых третья буква m и названия их поставщиков.

**SELECT** Products.ProductName, Suppliers.SupplierName **FROM** Products

**JOIN** Suppliers

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

**WHERE** ProductName **LIKE** '\_\_m%';



18. Показать имена и фамилии сотрудника, который оформил заказ 1996-11-27 (написать запрос двумя способами: через INNER Join, и используя подзапрос).

Вариант 1.

**SELECT** FirstName, LastName **FROM** Employees

**INNER** **JOIN** Orders

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

**WHERE** OrderDate = '1996-11-27';

Вариант 2.

**SELECT** FirstName, LastName **FROM** Employees

**WHERE** EmployeeID = (**SELECT** EmployeeID **FROM** Orders **WHERE** OrderDate = '1996-11-27');



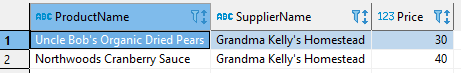
19. Выбрать все товары, у которых поставщик «Grandma Kelly's Homestead» и цена > 27. В результате вывести 3 колонки: Product, Supplier, Price.

**SELECT** Products.ProductName, Suppliers.SupplierName, Products.Price **FROM** Products

**INNER** **JOIN** Suppliers

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

**WHERE** Suppliers.SupplierName = "Grandma Kelly's Homestead" **AND** Price > 27;



20. Вывести суммарное количество продукта 'Queso Cabrales' (столбец обозвать Summ), отправленного всем покупателям (написать запрос двумя способами: через INNER Join, и используя подзапрос).

Вариант 1.

**SELECT** **SUM**(Quantity) **AS** Summ **FROM** OrderDetails

**INNER** **JOIN** Products

**ON** OrderDetails.ProductID = Products.ProductID

**WHERE** ProductName = 'Queso Cabrales';

Вариант 2.

**SELECT** **SUM**(Quantity) **AS** Summ **FROM** OrderDetails

**WHERE** ProductID = (**SELECT** ProductID **FROM** Products **WHERE** ProductName = 'Queso Cabrales');



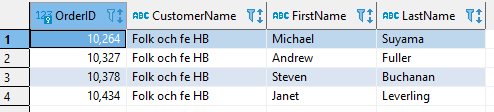
21. Показать все заказы, которые были отправлены по адресу «Ekergatan 24» с их заказчиками и сотрудниками. В результате вывести 3 колонки – ID заказа, имя заказчика, имя сотрудника, фамилия сотрудника.

**SELECT** Orders.OrderID, Customers.CustomerName, Employees.FirstName, Employees.LastName **FROM** Customers

**INNER** **JOIN** Orders **ON** Customers.CustomerID = Orders.CustomerID

**INNER** **JOIN** Employees **ON** Orders.EmployeeID = Employees.EmployeeID

**WHERE** Customers.Address = 'Ekergatan 24';



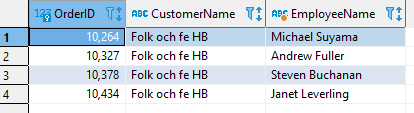
22. Преобразовать предыдущий запрос таким образом, чтобы те же данные выводились в 3-х колонках – объединить LastName и FirstName из Employees в одну колонку через пробел и назвать ее EmployeeName (2 LEFT JOINS).

**SELECT** Orders.OrderID, Customers.CustomerName, **CONCAT**(FirstName,' ', LastName) **AS** EmployeeName **FROM** Customers

**LEFT** **JOIN** Orders **ON** Customers.CustomerID = Orders.CustomerID

**LEFT** **JOIN** Employees **ON** Orders.EmployeeID = Employees.EmployeeID

**WHERE** Customers.Address = 'Ekergatan 24';



23. Показать все продукты, содержащиеся в заказах 1997-го года и в названии которых менее 5 букв. В результате вывести OrderID, OrderDate, ProductName (написать запрос двумя способами: через INNER JOINS, и используя подзапросы).

Вариант 1.

**SELECT** OrderDetails.OrderID, Orders.OrderDate, Products.ProductName **FROM** Products

**INNER** **JOIN** OrderDetails **ON** Products.ProductID = OrderDetails. ProductID

**INNER** **JOIN** Orders **ON** OrderDetails.OrderID = Orders.OrderID

**WHERE** **YEAR** (OrderDate) = 1997 **AND** **CHAR\_LENGTH**(ProductName) < 5;

Вариант 2.

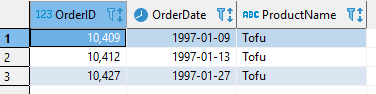
**SELECT** Orders.OrderID, Orders.OrderDate, Products.ProductName **FROM** Orders, Products, OrderDetails

**WHERE** Products.ProductID = OrderDetails.ProductID

**AND** OrderDetails.OrderID = Orders.OrderID

**AND** Orders.OrderID **IN** (**SELECT** OrderID **FROM** Orders **WHERE** **YEAR** (Orders.OrderDate) = 1997

**AND** ProductName = (**SELECT** ProductName **FROM** Products **WHERE** **CHAR\_LENGTH**(ProductName) < 5));



24. Показать названия продуктов и их категорий, которые используются в заказах от заказчика по имени Blondel père et fils и категории которых состоят как минимум из 2-х слов.

**SELECT** CategoryName, ProductName **FROM** Categories

**INNER** **JOIN** Products **ON** Categories.CategoryID = Products.CategoryID

**INNER** **JOIN** OrderDetails **ON** Products.ProductID = OrderDetails.ProductID

**INNER** **JOIN** Orders **ON** OrderDetails.OrderID = Orders.OrderID

**INNER** **JOIN** Customers **ON** Orders.CustomerID = Customers.CustomerID

**WHERE** CustomerName = 'Blondel pere et fils'

**AND** CategoryName **LIKE** '% %';



25. Вывести количество заказчиков (колонку назвать Buyers), которые сделали заказали один из продуктов: «Queso Cabrales», «Gustaf's Knäckebröd», «Louisiana Fiery Hot Pepper Sauce», «Schoggi Schokolade», «Gnocchi di nonna Alice».

**SELECT** **COUNT**(**DISTINCT** CustomerName) **As** Buyers **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

**WHERE** ProductName **IN**("Queso Cabrales", "Gustaf's Knäckebröd", "Louisiana Fiery Hot Pepper Sauce", "Schoggi Schokolade", "Gnocchi di nonna Alice");



26. \*Найти города в которые было отправлено больше всего заказов, вывести название города и количество заказов (колонку назвать Amount).

**SELECT** Customers.City, **COUNT**(CustomerName) **AS** Amount

**FROM** Customers

**INNER** **JOIN** Orders **ON** Customers.CustomerID = Orders.CustomerID

**GROUP** **BY** Customers.City

**HAVING** **COUNT**(Amount) = (**SELECT** **MAX**(amount) **AS** max\_amount

**FROM**

(**SELECT** Customers.City, **COUNT**(CustomerName) **AS** Amount

**FROM** Customers

**INNER** **JOIN** Orders **ON** Customers.CustomerID = Orders.CustomerID

**GROUP** **BY** Customers.City) query);



27. \*Найти из какого города было поставлено наибольшее количество единиц товаров в Лондон, вывести название города и количество единиц (колонку назвать Amount).

**SELECT** City, **SUM**(Quantity) **AS** Amount **FROM**

(**SELECT** Suppliers.City, Quantity **FROM** Customers

**INNER** **JOIN** Orders **USING** (CustomerID)

**INNER** **JOIN** OrderDetails **USING** (OrderID)

**INNER** **JOIN** Products **USING** (ProductID)

**INNER** **JOIN** Suppliers **USING** (SupplierID)

**WHERE** Customers.City = 'London'

**GROUP** **BY** Suppliers.City, Customers.CustomerID, OrderDetails.OrderID, ProductID, Quantity)query\_in\_1

**GROUP** **BY** City

**HAVING** Amount = (**SELECT** **MAX**(Amount) **FROM**

(**SELECT** City, **sum**(Quantity) **AS** Amount **FROM**

(**SELECT** Suppliers.City, Quantity **FROM** Customers

**INNER** **JOIN** Orders **USING** (CustomerID)

**INNER** **JOIN** OrderDetails **USING** (OrderID)

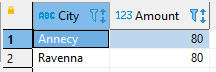
**INNER** **JOIN** Products **USING** (ProductID)

**INNER** **JOIN** Suppliers **USING** (SupplierID)

**WHERE** Customers.City = 'London'

**GROUP** **BY** Suppliers.City, Customers.CustomerID, OrderDetails.OrderID, ProductID, Quantity)query\_in\_2

**GROUP** **BY** City)query\_in\_3);



28. \*Найти перевозчиков, которые перевезли более 30 разнообразных напитков (Beverages), вывести имена перевозчиков, категорию товара и количество перевезенных видов товара (колонку назвать Amount).

**SELECT** ShipperName, CategoryName, **COUNT**(CategoryName) **AS** Amount **FROM** Shippers

**INNER** **JOIN** Orders **ON** Shippers.ShipperID = Orders.ShipperID

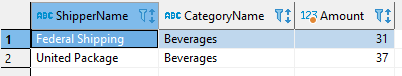
**INNER** **JOIN** OrderDetails **ON** Orders.OrderID = OrderDetails.OrderID

**INNER** **JOIN** Products **ON** OrderDetails.ProductID = Products.ProductID

**INNER** **JOIN** Categories **ON** Products.CategoryID = Categories.CategoryID

**GROUP** **BY** ShipperName, CategoryName

**HAVING** CategoryName = 'Beverages' **AND** Amount > 30;



29. \*Найти среднюю стоимость приправ (Condiments) отправленных в штаты, заказы на которые оформлены Margaret Peacock, вывести стоимость округленную до 2-х знаков после запятой (колонку назвать Average)

**SELECT** **ROUND**(**AVG**(Price), 2) **AS** Average **FROM** Products

**INNER** **JOIN** OrderDetails **USING** (ProductID)

**INNER** **JOIN** Orders **USING** (OrderID)

**WHERE** Orders.CustomerID **IN** (**SELECT** CustomerID **FROM** Customers

**WHERE** Country = 'USA')

**AND** EmployeeID = (**SELECT** EmployeeID **FROM** Employees

**WHERE** LastName = 'Peacock' **AND** FirstName = 'Margaret')

**AND** OrderDetails.ProductID **IN** (**SELECT** ProductID **FROM** Products

**WHERE** CategoryID = 2);



30. \*\* Найти сотрудников, которые оформили заказов на такой процент от общей стоимости всех оформленных заказов, который больше, чем процент общей стоимости заказов оформленных сотрудником, о котром в базе содержится самое длинное примечание (Notes), к общей стоимости всех заказов, который были перевезены перевозчиками, у которых номер телефона совпадает с номером телефона одного из поставщиков. Вывести полные имена сотрудников (в одной ячейке через пробел, назвав колонку EmployeeName) и процент от общей стоимости оформленных ими заказов к общей стоимости всех заказов, округленный до 2-х знаков после запятой, со значком процента через пробел после самой величины (назвав колонку Ratio).

**SELECT** EmployeeName, Ratio **FROM**

(**SELECT** EmployeeID, EmployeeName, **ROUND**((**SUM**(Cost)\*100/(**SELECT** **SUM**((Price\*Quantity)) **AS** Cost **FROM** OrderDetails

**INNER** **JOIN** Products **USING** (ProductID))), 2) **AS** Ratio **FROM**

((**SELECT** EmployeeID, **CONCAT**(FirstName, ' ', LastName) **AS** EmployeeName, (Price\*Quantity) **AS** Cost **FROM** OrderDetails

**INNER** **JOIN** Products **USING** (ProductID)

**INNER** **JOIN** Orders **USING** (OrderID)

**INNER** **JOIN** Employees **USING** (EmployeeID))query\_1)

**GROUP** **BY** EmployeeID)query\_2

**WHERE** Ratio > (**SELECT** **ROUND**((**SUM**(Cost)\*100/(**SELECT** **SUM**((Price\*Quantity)) **AS** Cost **FROM** OrderDetails

**INNER** **JOIN** Products **USING** (ProductID)

**INNER** **JOIN** Orders **USING** (OrderID)

**INNER** **JOIN** Employees **USING** (EmployeeID)

**WHERE** ShipperID = (**SELECT** ShipperID **FROM** Shippers

**WHERE** Phone = (**SELECT** Phone **FROM** Suppliers

**WHERE** Phone **LIKE** '(503) 555-9831' **OR** Phone **LIKE** '(503) 555-3199' **OR** Phone **LIKE** '(503) 555-9931')))), 2) **AS** Percent\_2 **FROM**

((**SELECT** EmployeeID, **CONCAT**(FirstName, ' ', LastName) **AS** EmployeeName, (Price\*Quantity) **AS** Cost **FROM** OrderDetails

**INNER** **JOIN** Products **USING** (ProductID)

**INNER** **JOIN** Orders **USING** (OrderID)

**INNER** **JOIN** Employees **USING** (EmployeeID))query\_3)

**GROUP** **BY** EmployeeID

**HAVING** EmployeeID = (**SELECT** EmployeeID **FROM** Employees

**WHERE** **CHAR\_LENGTH**(Notes) = (**SELECT** **MAX**(**CHAR\_LENGTH**(Notes)) **FROM** Employees)));

