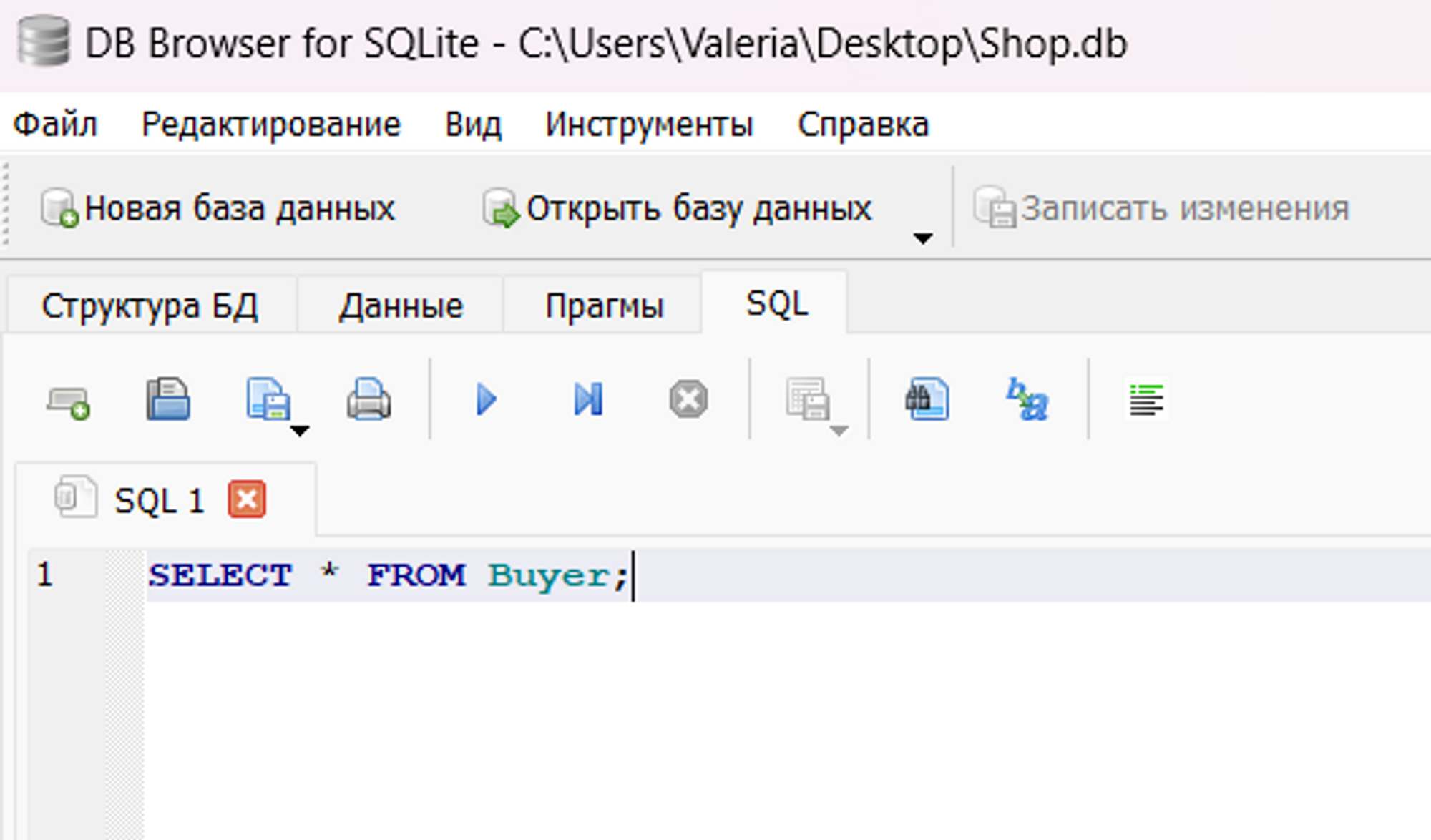
**Реализация запросов SQL.**

Все запросы выполнены через СУБД SQLite. При использовании PostgreSQL используется аналогичный синтаксис, практически не имеющий значительных отличий.

**Начало работы с запросами.**

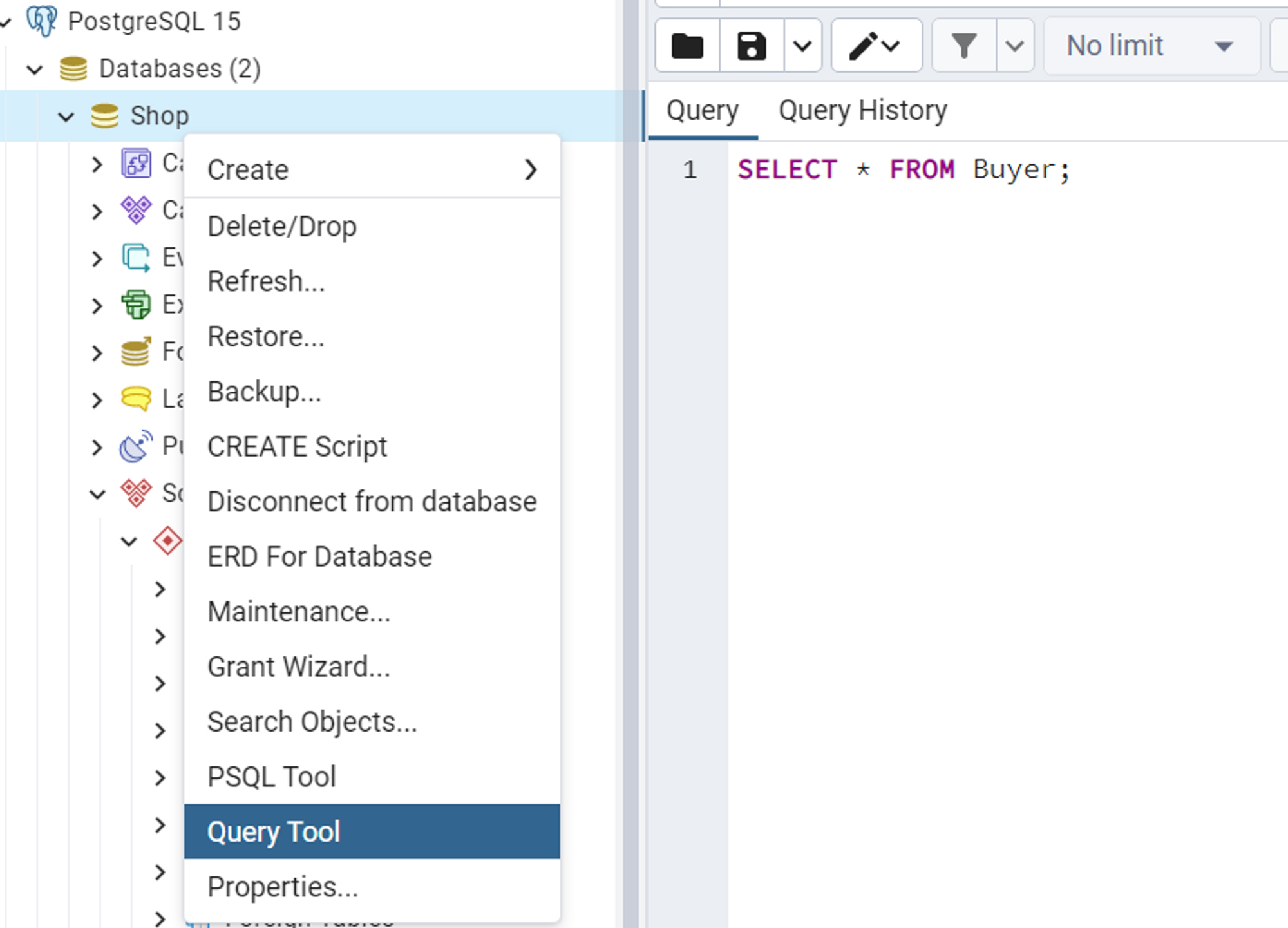
Для создания запроса следует перейти в SQLite и выбрать окно «SQL», как указано на Рисунке 1.

Рисунок 1. Запросы через интерфейс DB Browser for SQLite



Для написания запросов через PostgreSQL следует запустить ранее установленный pgAdmin, нажать правой кнопкой мыши на базу данных и выбрать «Query Tool» (Рисунок 2).

Рисунок 2. Запросы через интерфейс pgAdmin



**Создание запросов и реализация.**

Ниже указаны задания по извлечению информации из базы данных, а также соответствующий такому запросу код и результат выполнения. В каждом запросе используется только одна таблица. Для каждого оператора указано пять запросов, но будут проиллюстрированы лишь два запроса и их результат. Оставшиеся три предлагается реализовать самостоятельно.

**Агрегирующие функции.**

**COUNT.**

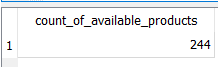
1. Вывести количество уникальных наименований товаров:

|  |  |
| --- | --- |
| 1  2 | **SELECT** **COUNT**(**DISTINCT** name)  **FROM** product; |



1. Вывести количество товаров, имеющихся в наличии:

|  |  |
| --- | --- |
| 1  2  3 | **SELECT** **COUNT**(\*) **AS** count\_of\_available\_products  **FROM** product  **WHERE** availability = **true**; |



1. Вывести количество отмененных заказов;
2. Вывести количество складов, находящихся в г. Казань;
3. Вывести количество товаров, которые находятся в категории «Продукты питания» и дороже 100.

**MAX.**

1. Вывести адрес склада, в котором находится самый дорогой товар:

|  |  |
| --- | --- |
| 1  2  3  4  5  6  7  8 | **SELECT** ('г. ' || city || ', ' || address) **AS** **location**  **FROM** store  **WHERE** (**SELECT** price  **FROM** product  **WHERE** price = (  **SELECT** **MAX**(price)  **FROM** product)  **AND** product.store\_id = store.id); |



1. Вывести товар, которого больше всего на складе:

|  |  |
| --- | --- |
| 1  2  3  4  5 | **SELECT** name  **FROM** product  **WHERE** quantity = (  **SELECT** **MAX**(quantity)  **FROM** product); |

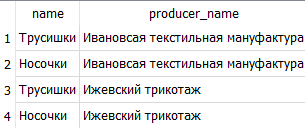


1. Вывести максимальную цену товара;
2. Вывести ID заказа, который содержит максимальное количество товаров;
3. Вывести адрес склад, в котором находится товар с максимальным количеством в наличии.

**MIN.**

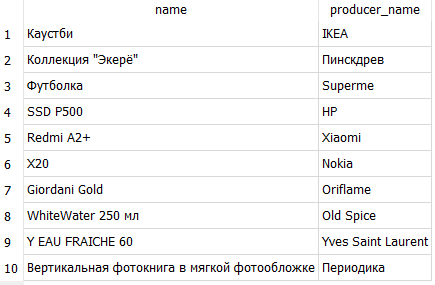
1. Найти самые дешёвые товары:

|  |  |
| --- | --- |
| 1  2  3  4  5  6  7  8 | **SELECT** name,  (**SELECT** name  **FROM** producer  **WHERE** producer.id = product.producer\_id) **AS** producer\_name  **FROM** product  **WHERE** price = (  **SELECT** **MIN**(price)  **FROM** product); |



1. Найти товар, которого меньше всего на складе:

|  |  |
| --- | --- |
| 1  2  3  4  5  6  7  8  9 | **SELECT** name,  (**SELECT** name  **FROM** producer  **WHERE** producer.id = product.producer\_id) **AS** producer\_name  **FROM** product  **WHERE** quantity = (  **SELECT** **MIN**(quantity)  **FROM** product  **WHERE** availability = **true**); |



1. Вывести минимальную цену товара;
2. Вывести категорию, которая содержит минимальное количество товаров;
3. Вывести адрес склад, в котором находится товар с минимальным количеством в наличии.

**SUM.**

1. Вывести сумму цен всех товаров:

|  |  |
| --- | --- |
| 1  2 | **SELECT** **SUM**(price \* quantity) **AS** total\_cost  **FROM** product; |



1. Вывести выручку (сумма всех выполненных заказов):

|  |  |
| --- | --- |
| 1  2  3  4  5 | **SELECT** **SUM**(p.price \* op.quantity) **AS** revenue  **FROM** product p, order\_product op, 'order' o  **WHERE** p.ID = op.product\_id  **AND** op.order\_id = o.id  **AND** status = 'выполнен'; |



1. Вывести сумму цен всех товаров от производителя «Milka»;
2. Вывести сумму цен всех товаров в наличии из категории «Одежда и обувь»;
3. Вывести заказы, где количество товаров от 10 до 30.

**AVG.**

1. Вывести среднюю цену на товар от производителя «Chertilla Italiano»:

|  |  |
| --- | --- |
| 1  2  3  4 | **SELECT** **AVG**(product.price)  **FROM** product, producer  **WHERE** producer.name = 'Chertilla Italiano'  **AND** producer.id = product.producer\_id; |



1. Вывести среднюю цену товаров в наличии на складе по адресу «г. Казань, ул. Пушкина, д. Колотушкина»:

|  |  |
| --- | --- |
| 1  2  3  4  5  6  7  8 | **SELECT** ROUND(**AVG**(price), 2) **AS** avg\_coast  **FROM** product  **WHERE** store\_id = (  **SELECT** id  **FROM** store  **WHERE** city = 'Казань'  **AND** address = 'ул. Пушкина, д. Колотушкина')  **AND** availability = **true**; |



1. Вывести среднее количество товаров в заказе;
2. Вывести среднюю цену на товары из категории «Иные товары»;
3. Вывести среднюю стоимость заказов.

**Группировка.**

1. Вывести количество уникальных товаров по каждому производителю. Условие: вывести только тех производителей, количество уникальных товаров которых превышает 4. Результат отсортировать по количеству в порядке убывания:

|  |  |
| --- | --- |
| 1  2  3  4  5  6  7  8 | **SELECT** name,  (**SELECT** **COUNT**(**DISTINCT** name)  **FROM** product  **WHERE** product.producer\_id = producer.id) **AS** **count**  **FROM** producer  **WHERE** **count** > 4  **GROUP** **BY** producer.name  **ORDER** **BY** **count** **DESC;** |



1. Вывести количество уникальных товаров по каждой категории. Результат отсортировать по количеству в порядке возрастания:

|  |  |
| --- | --- |
| 1  2  3  4  5 | **SELECT** **c**.name,  **COUNT**(**DISTINCT** (p.name || p.producer\_id)) **AS** counter  **FROM** product p, category **c**  **WHERE** p.category\_id = **c**.id  **GROUP** **BY** **c**.id; |



1. Вывести среднюю цену товаров для каждого производителя (не дороже 2500):

|  |  |
| --- | --- |
| 1  2  3  4  5 | **SELECT** producer.name, ROUND(**AVG**(product.price), 2) **AS** average\_price  **FROM** product, producer  **WHERE** product.producer\_id = producer.id  **GROUP** **BY** producer.id  **HAVING** average\_price < 2500 |



1. Вывести город, в котором больше всего складов:

|  |  |
| --- | --- |
| 1  2  3  4  5  6  7  8 | **SELECT** store.city, **COUNT**(\*) **AS** total\_count  **FROM** store  **GROUP** **BY** store.city  **HAVING** **COUNT**(\*) = (  **SELECT** **MAX**(**count**)  **FROM** (**SELECT** **COUNT**(\*) **AS** **count**  **FROM** store  **GROUP** **BY** city)); |



1. Вывести наиболее популярного производителя по каждой категории;
2. Вывести общее количество заказов для каждого покупателя;
3. Вывести суммарную стоимость всех заказов для каждого покупателя;
4. Вывести суммарное количество всех товаров по каждому магазину.

**\*Дополнительно для заинтересованных.**

Сейчас мы покажем вам запрос, в котором использовано всё, что мы изучили в этой теме.

Запрос: Вывести ФИО покупателя, который купил максимальное количество товаров:

|  |  |
| --- | --- |
| 1  2  3  4  5  6  7  8  9  10  11  12  13  14  15  16  17  18  19 | **SELECT** name  **FROM** buyer  **WHERE** id **IN** (  **SELECT** buyer\_id  **FROM** 'order'  **WHERE** id **IN** (  **SELECT** order\_id  **FROM** order\_product  **GROUP** **BY** order\_id  **HAVING** **SUM**(quantity) = (  **SELECT** **MAX**(order\_count)  **FROM** (  **SELECT** **SUM**(quantity) **as** order\_count  **FROM** order\_product  **GROUP** **BY** order\_id  ) **as** total\_count  )  )  ); |

Попробуйте понять его работу. В следующих темах мы будем совершенствовать этот запрос по мере изучения новых команд.