

# SQL Задачи и решения

Учебник. Сергей Моисеенко.



◀ назад листать вперед ▶

Новое в стандарте и реализациях языка SQL

## CROSS APPLY / OUTER APPLY стр. 2

А теперь представьте, что нам нужно, помимо максимальной цены, вывести минимальную, среднюю цены и т.д. Поскольку коррелирующий подзапрос в предложении SELECT должен возвращать только одно значение, в первом варианте решения нам придется фактически дублировать код для каждого агрегата:



Выполнить



Консоль

```
1. SELECT *, (SELECT MAX(price) FROM Laptop L2
2. JOIN Product P1 ON L2.model=P1.model
3. WHERE maker = (SELECT maker FROM Product P2 WHERE P2.model= L1.model)) max_price,
4. (SELECT MIN(price) FROM Laptop L2
5. JOIN Product P1 ON L2.model=P1.model
6. WHERE maker = (SELECT maker FROM Product P2 WHERE P2.model= L1.model)) min_price
7. FROM Laptop L1;
```

и т.д.

А при использовании CROSS APPLY мы просто добавим в подзапрос требуемую агрегатную функцию:



Выполнить



Консоль

```
1. SELECT *
2. FROM laptop L1
3. CROSS APPLY
4. (SELECT MAX(price) max_price, MIN(price) min_price FROM Laptop L2
5. JOIN Product P1 ON L2.model=P1.model
6. WHERE maker = (SELECT maker FROM Product P2 WHERE P2.model= L1.model)) X;
```

Рассмотрим еще один пример.

Соединить каждую строку из таблицы Laptop со следующей строкой в порядке, заданном сортировкой (model, code).

Столбец code в сортировке используется для того, чтобы задать однозначный порядок для строк, имеющих одинаковые значения в столбце model. С помощью CROSS APPLY мы можем передать в подзапрос параметры текущей строки и выбрать первую строку из тех, которые идут ниже текущей в заданном сортировкой порядке. Итак,



Выполнить



Консоль

```
1. SELECT * FROM laptop L1
2. CROSS APPLY
3. (SELECT TOP 1 * FROM Laptop L2
4. WHERE L1.model < L2.model OR (L1.model = L2.model AND L1.code < L2.code)
5. ORDER BY model, code) X
6. ORDER BY L1.model;
```

Попробуйте решить эту задачу традиционными средствами, чтобы сравнить трудозатраты.

### Оператор OUTER APPLY

Как показывают результаты предыдущего запроса, мы "потеряли" последнюю (шестую) строку из таблицы Laptop, поскольку ее не с чем соединять. Другими словами, CROSS APPLY ведет себя как внутренне соединение. Аналогом же внешнего (левого) соединения является оператор OUTER APPLY. Он отличается от CROSS APPLY только тем, что выводит все строки из левой таблицы, заменяя отсутствующие значения из правой таблицы NULL-значениями.

Замена CROSS APPLY на OUTER APPLY в предыдущем запросе иллюстрирует сказанное.



Выполнить



Консоль

```
1. SELECT * FROM laptop L1
```



```
2. OUTER APPLY
3. (SELECT TOP 1 *
4. FROM Laptop L2
5. WHERE L1.model < L2.model OR (L1.model = L2.model AND L1.code < L2.code)
6. ORDER BY model, code) X
7. ORDER BY L1.model;
```

Еще одной популярной задачей является вывод по N строк из каждой группы. Примером может служить вывод 5 наиболее популярных товаров в каждой категории. Рассмотрим следующую задачу.

Вывести из таблицы Product по три модели с наименьшими номерами из каждой группы, характеризующей типом продукции.

Дополним [решения](#), предложенные на сайте sql-ex.ru, решением, использующим CROSS APPLY. Идея заключается в соединении уникальных типов (первый запрос) с запросом, выводящим по 3 модели каждого типа из первого запроса в соответствии с требуемой сортировкой.



Выполнить



Консоль

```
1. SELECT X.* FROM
2. (SELECT DISTINCT type FROM product) Pr1
3. CROSS APPLY
4. (SELECT TOP 3 * FROM product Pr2 WHERE Pr1.type=Pr2.type ORDER BY pr2.model) x;
```

Страницы: 1 2 3 4

коррелирующие подзапросы CROSS APPLY OUTER APPLY



[Предыдущая \[COUNT DISTINCT и оконные функции\]](#)

[\[Функция CONCAT\]](#) [Следующая](#) 

#### Последние изменения:

Метод наименьших квадратов

Вставка строк в таблицу, содержащую автоинкрементируемое поле  
стр. 5

Сортировка по дням рождения  
стр. 2

Функция STRING\_AGG стр. 2

Функция TRANSLATE

Упражнение 151 (подсказки и решения)

Предикат LIKE

Приложение 2. Список задач  
стр. 2

Приложение 2. Список задач  
Футбол



#### Тэги:

поиск по тэгам

ALL AND AUTO\_INCREMENT AVG  
battles CASE CAST CHAR  
CHARINDEX CHECK classes  
COALESCE CONSTRAINT Convert  
COUNT CROSS APPLY CTE  
DATEADD DATEDIFF DATENAME  
DATEPART DATETIME DDL  
DEFAULT DELETE DISTINCT DML  
EXCEPT EXISTS EXTRACT  
FOREIGN KEY FROM FULL JOIN

GROUP BY Guadalcanal HAVING  
IDENTITY IN  
INFORMATION\_SCHEMA INNER  
JOIN insert INTERSECT IS NOT  
NULL IS NULL ISNULL laptop LEFT  
LEFT OUTER JOIN LEN maker

[Больше тэгов](#)

Учебник обновлялся  
месяц назад  
<https://exchangesumo.com/obmen/RN>  
NIXEUR/ . ремонт ноутбуков

---

©SQL-EX,2008 [Развитие] [Связь] [О проекте] [Ссылки] [Team]  
Перепечатка материалов сайта возможна только с разрешения автора.

