

SQL Задачи и решения

Учебник. Сергей Моисеенко.



◀ назад листать вперед ▶

Новое в стандарте и реализациях языка SQL

CROSS APPLY / OUTER APPLY стр. 4

PostgreSQL обладает аналогичной функциональностью. Синтаксические отличия незначительны и состоят в замене CROSS APPLY на CROSS JOIN LATERAL. Сравните три примера, которые рассматривались на предыдущих страницах этой главы.

Пример 1

SQL Server



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;
```

PostgreSQL

```
1. SELECT *
2.     FROM laptop L1
3.     CROSS JOIN LATERAL
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;
```

Пример 2

SQL Server



Выполнить



Консоль

```
1. SELECT code, name, value FROM Laptop
2.     CROSS APPLY
3.     (VALUES('speed', speed)
4.      ,('ram', ram)
5.      ,('hd', hd)
6.      ,('screen', screen)
7.      ) spec(name, value)
8.     WHERE code < 4
9.     ORDER BY code, name, value;
```

PostgreSQL

```
1. SELECT code, name, value FROM Laptop
2.     CROSS JOIN LATERAL
3.     (VALUES('speed', speed)
4.      ,('ram', ram)
5.      ,('hd', hd)
6.      ,('screen', screen)
7.      ) spec(name, value)
8.     WHERE code < 4
9.     ORDER BY code, name, value;
```

Пример 3

SQL Server



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;
```

PostgreSQL

Дополнительное отличие в этом примере связано не с реализацией CROSS APPLY, а с тем, что для ограничения выборки PostgreSQL вместо конструкции TOP(n) использует LIMIT n в предложении ORDER BY.

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

OUTER APPLY

Для данного "внешнего" соединения в PostgreSQL используется LEFT JOIN LATERAL. Сравните запросы в примере 4.



Пример 4



SQL Server



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;
```

PostgreSQL

Обратите внимание на предикат ON TRUE. Поскольку синтаксис соединения [LEFT|RIGHT [OUTER]] JOIN требует предиката, то для единообразия используется "фиктивный" предикат, имеющий значение ИСТИНА.

```
1. SELECT * FROM laptop L1
2. LEFT JOIN LATERAL
3. (SELECT *
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 LIMIT 1) X ON TRUE
7. ORDER BY L1.model;
```

Страницы: 1 2 3 4

SQL Server PostgreSQL CROSS APPLY cross join lateral lateral



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

ремонт ноутбука . Допустимые
погрешности микрометра

