Соединения

Иногда данных одной таблицы недостаточно, чтобы получить те данные, которые вам необходимы. В этом случае используются соединения таблиц. В рамках данной работы будет объяснено:

1. соединение (join)
2. левое внешнее соединение LEFT OUTER JOIN
3. правое внешнее соединение — RIGHT OUTER JOIN
4. полное внешнее соединение — FULL OUTER JOIN.

**Соединение (join)**

В тех случаях, когда информации, содержащейся в одной таблице, недостаточно для получения требуемого результата, используют соединение (join) таблиц. Покажем способ выполнения соединения на примере следующего запроса: выбрать все места, предусмотренные компоновкой салона самолета Cessna 208 Caravan.

**SELECT a.aircraft\_code, a.model, s.seat\_no, s.fare\_conditions**

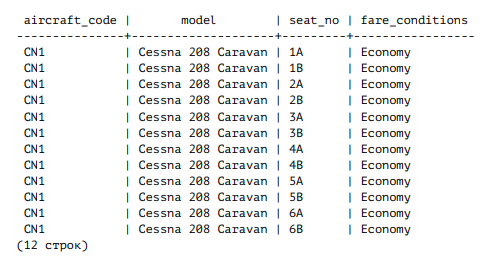
**FROM seats AS s**

**JOIN aircrafts AS a**

**ON s.aircraft\_code = a.aircraft\_code**

**WHERE a.model ~ '^Cessna'**

**ORDER BY s.seat\_no;**



Данная команда иллюстрирует соединение двух таблиц на основе равенства значений атрибутов. В этой команде в предложении FROM указаны две таблицы — aircrafts и seats, причем каждая из них получила еще и псевдоним с помощью ключевого слова AS (заметим, что оно не является обязательным). Конечно, псевдонимы могут состоять не только из одной буквы, как в нашем примере. Псевдонимы удобны в тех случаях, когда в соединяемых таблицах есть одноименные атрибуты. В таких случаях в списке атрибутов, следующих за ключевым словом SELECT, необходимо указывать либо имя таблицы, из которой выбирается значение этого атрибута, либо ее псевдоним, но псевдоним может быть коротким, что удобнее при написании команды. Псевдоним и атрибут соединяются символом «.».

Итак, как мы рассуждали? Если бы в качестве исходных сведений мы получили сразу код самолета — CN1, то запрос свелся бы к выборке из одной таблицы «Места». Он был бы таким:

**SELECT \* FROM seats WHERE aircraft\_code = 'CN1';**

Но нам дано название модели, а не ее код, поэтому придется подключить к работе и таблицу «Самолеты» (aircrafts), в которой хранятся наименования моделей. Для того чтобы решить, удовлетворяет ли строка таблицы seats поставленному условию, нужно узнать, какой модели самолета соответствует эта строка.

Как это можно узнать? В каждой строке таблицы seats есть атрибут aircraft\_code, такой же атрибут есть и в каждой строке таблицы aircrafts. Если с каждой строкой таблицы seats соединить такую строку таблицы aircrafts, в которой значение атрибута aircraft\_code такое же, как и в строке таблицы seats, то сформированная комбинированная строка, составленная из атрибутов обеих таблиц, будет содержать не только номер места, класс обслуживания и код модели, но — что важно — и наименование модели. Поэтому с помощью условия WHERE можно будет отобрать только те результирующие строки, в которых значение атрибута model будет «Cessna 208 Caravan».

А какие столбцы оставлять в списке столбцов предложения SELECT, решать нам. Даже если мы соединяем две таблицы (или более), то совершенно не обязательно в результирующий список столбцов включать столбцы всех таблиц, перечисленных в предложении FROM. Мы могли бы оставить только атрибуты таблицы seats:

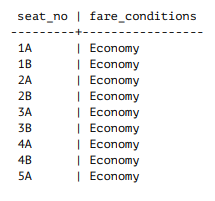
**SELECT s.seat\_no, s.fare\_conditions**

**FROM seats s**

**JOIN aircrafts a ON s.aircraft\_code = a.aircraft\_code**

**WHERE a.model ~ '^Cessna'**

**ORDER BY s.seat\_no;**

****

Если подвести итог, то можно упрощенно объяснить механизм построения соединения следующим образом.

Сначала формируются все попарные комбинации строк из обеих таблиц, т. е. декартово произведение множеств строк этих таблиц. Эти комбинированные строки включают в себя все атрибуты обеих таблиц.

Затем в дело вступает условие s.aircraft\_code = a.aircraft\_code. Это означает, что в результирующем множестве строк останутся только те из них, в которых значения атрибута aircraft\_code, взятые из таблицы aircrafts и из таблицы seats, одинаковые. Строки, не удовлетворяющие этому критерию, отфильтровываются.

Это означает на практике, что каждой строке из таблицы «Места» мы сопоставили только одну конкретную строку из таблицы «Самолеты», из которой мы теперь можем взять значение атрибута «Модель самолета», чтобы включить ее в итоговый вывод данных.

**Левое внешнее соединение LEFT OUTER JOIN**

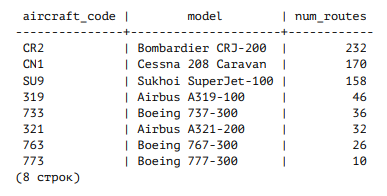
Теперь обратимся к так называемым внешним соединениям. Зададимся вопросом: сколько маршрутов обслуживают самолеты каждого типа? Если не требовать вывода наименований моделей самолетов, тогда всю необходимую информацию можно получить из материализованного представления «Маршруты» (routes). Но мы все же будем выводить и наименования моделей, поэтому обратимся также к таблице «Самолеты» (aircrafts). Соединим эти таблицы на основе атрибута aircraft\_code, сгруппируем строки и просто воспользуемся функцией count. В этом запросе внешнее соединение еще не используется.

**SELECT r.aircraft\_code, a.model, count( \* ) AS num\_routes**

**FROM routes r JOIN aircrafts a ON r.aircraft\_code = a.aircraft\_code**

**GROUP BY 1, 2**

**ORDER BY 3 DESC;**

****

Обратите внимание, что таблица «Самолеты» содержит 9 моделей, а в этой выборке лишь 8 строк. Значит, какая-то модель самолета не участвует в выполнении рейсов. Как ее выявить? С помощью такого запроса:

**SELECT a.aircraft\_code AS a\_code,**

**a.model,**

**r.aircraft\_code AS r\_code,**

**count( r.aircraft\_code ) AS num\_routes**

**FROM aircrafts a LEFT OUTER JOIN routes r ON r.aircraft\_code = a.aircraft\_code**

**GROUP BY 1, 2, 3**

**ORDER BY 4 DESC;**