

ТЕМА 9. ОПЕРАТОР CREATE TABLE

Синтаксис

```
CREATE TABLE [ IF NOT EXISTS ] имя_таблицы ( [  
    { имя_столбца тип_данных [ ограничение_столбца [ ... ] ]  
    | ограничение_таблицы }  
    [, ... ]  
]
```

Здесь *ограничение_столбца*:

```
[ CONSTRAINT имя_ограничения ]  
{ NOT NULL |  
  NULL |  
  CHECK ( выражение ) |  
  DEFAULT выражение_по_умолчанию |  
  UNIQUE параметры_индекса |  
  PRIMARY KEY параметры_индекса |  
  REFERENCES целевая_таблица [ ( целевой_столбец ) ] [ MATCH FULL | MATCH  
PARTIAL | MATCH SIMPLE ]  
  [ ON DELETE действие ] [ ON UPDATE действие ] }
```

и *ограничение_таблицы*:

```
[ CONSTRAINT имя_ограничения ]  
{ CHECK ( выражение ) |  
  UNIQUE ( имя_столбца [, ... ] ) |  
  PRIMARY KEY ( имя_столбца [, ... ] ) |  
  FOREIGN KEY ( имя_столбца [, ... ] ) REFERENCES целевая_таблица [ ( целевой_столбец [, ... ] ) ]  
  [ MATCH FULL | MATCH PARTIAL | MATCH SIMPLE ] [ ON DELETE действие ] [ ON  
UPDATE действие ] }
```

Описание

CREATE TABLE создаёт новую, изначально пустую таблицу в текущей базе данных. Владелец таблицы будет пользователем, выполнивший эту команду.

Необязательные предложения ограничений задают ограничения (проверки), которым должны удовлетворять добавляемые или изменяемые строки, чтобы операция добавления или изменения была выполнена успешно. Ограничение представляет собой SQL-объект, помогающий некоторым способом определить множество допустимых значений в таблице.

Определить ограничения можно двумя способами: в виде ограничений таблицы и в виде ограничений столбца. Ограничение столбца определяется как часть определения столбца, а ограничение таблицы не привязывается к конкретному столбцу и может задействовать несколько столбцов. Любые ограничения столбцов можно также записать в виде ограничения таблицы, они введены просто для удобства записи в случаях, когда ограничение затрагивает только один столбец.

Параметры

IF NOT EXISTS

Не считать ошибкой, если отношение с таким именем уже существует. В этом случае будет выдано замечание. Заметьте, что нет никакой гарантии, что существующее отношение как-то соотносится с тем, которое могло бы быть создано.

имя_таблицы

Имя создаваемой таблицы (возможно, дополненное схемой).

имя_столбца

Имя столбца, создаваемого в новой таблице.

тип_данных

Тип данных столбца (может включать определение массива с этим типом).

CONSTRAINT имя_ограничения

Необязательное имя столбца или ограничения таблицы. При нарушении ограничения его имя будет выводиться в сообщении об ошибках, так что имена ограничений вида `столбец` должен быть положительным могут сообщить полезную информацию об ограничении клиентскому приложению. (Имена ограничений, включающие пробелы, необходимо заключать в двойные кавычки.) Если имя ограничения не указано, система генерирует имя автоматически.

NOT NULL

Данный столбец не принимает значения NULL.

NULL

Данный столбец может содержать значения NULL (по умолчанию).

Это предложение предназначено только для совместимости с нестандартными базами данных SQL. Использовать его в новых приложениях не рекомендуется.

CHECK (выражение)

В ограничении **CHECK** задаётся выражение, возвращающее булевский результат, по которому определяется, будет ли успешна операция добавления или изменения для конкретных строк. Операция выполняется успешно, если результат выражения равен TRUE или UNKNOWN. Если же для какой-нибудь строки, задействованной в операции добавления или изменения, будет получен результат FALSE, возникает ошибка, и эта операция не меняет ничего в базе данных. Ограничение-проверка, заданное как ограничение столбца, должно ссылаться только на значение самого столбца, тогда как ограничение на уровне таблицы может ссылаться и на несколько столбцов.

В настоящее время выражения **CHECK** не могут содержать подзапросы или ссылаться на переменные, кроме как на столбцы текущей строки.

Когда для таблицы задано несколько ограничений **CHECK**, они будут проверяться для каждой строки в алфавитном порядке имён после проверки ограничений **NOT NULL**.

DEFAULT выражение_по_умолчанию

Предложение **DEFAULT** задаёт значение по умолчанию для столбца, в определении которого оно присутствует. Значение задаётся выражением без переменных (подзапросы и перекрёстные ссылки на другие столбцы текущей таблицы в нём не допускаются). Тип данных выражения, задающего значение по умолчанию, должен соответствовать типу данных столбца.

Это выражение будет использоваться во всех операциях добавления данных, в которых не задаётся значение данного столбца. Если значение по умолчанию не определено, таким значением будет NULL.

UNIQUE (ограничение столбца)

UNIQUE (имя_столбца [, ...]) (ограничение таблицы)

Ограничение **UNIQUE** определяет, что группа из одного или нескольких столбцов таблицы может содержать только уникальные значения. Ограничение уникальности для таблицы ведёт себя точно так же, как ограничение для столбца, но может охватывать несколько столбцов.

При проверке ограничения уникальности значения NULL не считаются равными.

В каждом ограничении уникальности для таблицы должен задаваться набор столбцов, отличный от набора любого другого ограничения уникальности или первичного ключа в данной таблице. (Иначе это будет одно и то же ограничение, определённое многократно.)

PRIMARY KEY (ограничение столбца)

PRIMARY KEY (имя_столбца [, ...]) (ограничение таблицы)

Ограничение **PRIMARY KEY** определяет, что столбец или столбцы таблицы могут содержать только уникальные (без повторений) значения, отличные от NULL. Для таблицы может быть задан только один первичный ключ, будь то ограничение столбца или ограничение таблицы.

В определении первичного ключа должен задаваться набор столбцов, отличный от набора любого другого ограничения уникальности, установленного для данной таблицы. (В противном случае уникальное ограничение оказывается избыточным и будет отброшено.)

PRIMARY KEY устанавливает для данных те же ограничения, что и сочетание **UNIQUE** и **NOT NULL**, но образование первичного ключа из набора столбцов также добавляет метаданные о конструкции схемы, так как первичный ключ подразумевает, что другие таблицы могут ссылаться на этот набор столбцов, как на уникальный идентификатор строк.

REFERENCES внешняя_таблица [(внешний_столбец)] [

MATCH тип_совпадения] [ON DELETE действие] [ON

UPDATE действие] (ограничение столбца)

FOREIGN KEY (имя_столбца [, ...]) REFERENCES внешняя_таблица [(внешний_столбец [, ...])] [MATCH тип_совпадения] [ON DELETE действие] [ON UPDATE действие] (ограничение таблицы)

Эти предложения определяют ограничение внешнего ключа, требующее, чтобы указанная группа из одного или нескольких столбцов новой таблицы содержала только такие значения,

которым соответствуют значения в заданных внешних столбцах некоторой строки во внешней таблице. Если список *внешних_столбцов* опущен, в качестве него используется первичный ключ *внешней_таблицы*. В качестве внешних столбцов должны указываться столбцы неоткладываемого уникального ограничения или ограничение первичного ключа во внешней таблице. Заметьте, что ограничения внешнего ключа, связывающие временные и постоянные таблицы, определить нельзя.

Значения, вставляемые в ссылающиеся столбцы, сверяются со значениями во внешних столбцах внешней таблицы с учётом заданного типа совпадения. Возможны три типа совпадения: `MATCH FULL` (полное совпадение), `MATCH PARTIAL` (частичное совпадение) и тип по умолчанию, `MATCH SIMPLE` (простое совпадение). С `MATCH FULL` ни один из столбцов составного внешнего ключа не может содержать `NULL`, кроме случая, когда все внешние столбцы `NULL`; в этом случае, строка может не иметь соответствия во внешней таблице. С `MATCH SIMPLE` любой из столбцов внешнего ключа может содержать `NULL`; при этом строка с `NULL` в одном из таких столбцов может не иметь соответствия во внешней таблице. Тип `MATCH PARTIAL` ещё не реализован. (Разумеется, чтобы вопросы со сравнением `NULL` не возникали, к столбцам, ссылающимся на внешние, можно применить ограничения `NOT NULL`.)

Кроме того, при изменении значений во внешних столбцах с данными в столбцах этой таблицы могут производиться определённые действия. Предложение `ON DELETE` задаёт действие, производимое при удалении некоторой строки во внешней таблице. Предложение `ON UPDATE` подобным образом задаёт действие, производимое при изменении значения в целевых столбцах внешней таблицы. Если строка изменена, но это изменение не затронуло целевые столбцы, никакое действие не производится. Ссылочные действия, кроме `NO ACTION`, нельзя сделать откладываемыми, даже если ограничение объявлено как откладываемое. Для каждого предложения возможные следующие варианты действий:

`NO ACTION`

Выдать ошибку, показывающую, что при удалении или изменении записи произойдёт нарушение ограничения внешнего ключа. Для отложенных ограничений ошибка произойдёт в момент проверки ограничения, если строки, ссылающиеся на эту запись, по-прежнему будут существовать. Этот вариант действия подразумевается по умолчанию.

`RESTRICT`

Выдать ошибку, показывающую, что при удалении или изменении записи произойдёт нарушение ограничения внешнего ключа. Этот вариант подобен `NO ACTION`, но эта проверка будет неоткладываемой.

`CASCADE`

Удалить все строки, ссылающиеся на удаляемую запись, либо поменять значения в ссылающихся столбцах на новые значения во внешних столбцах, в соответствии с операцией.

`SET NULL`

Установить ссылающиеся столбцы равными `NULL`.

`SET DEFAULT`

Установить в ссылающихся столбцах значения по умолчанию. (Если эти значения не равны `NULL`, во внешней таблице должна быть строка, соответствующая набору значений по умолчанию; в противном случае операция завершится ошибкой.)

Если внешние столбцы меняются часто, будет разумным добавить индекс для ссылающихся столбцов, чтобы действия по обеспечению ссылочной целостности, связанные с ограничением внешнего ключа, выполнялись более эффективно.