

Тема 8. Операторы DDL

Демонстрацию операторов DDL языка SQL будем проводить на примере базы данных СНАБЖЕНИЕ, описывающей взаимодействие поставщиков и покупателей товаров.

База данных состоит из трех отношений со схемами:

- Покупатели (Фамилия, Адрес, Баланс),
- Поставщики (Название, Адрес, Товар, Цена),
- Заказы (Фамилия, Товар, Количество).

В приведенных далее описаниях операторов используются следующие соглашения: конструкция [текст] означает необязательность заключенного в скобки фрагмента оператора, конструкция {текст},... - одно или несколько повторений фрагмента, конструкция {вариант1|вариант2|вариант3...} - выбор одного из вариантов.

Оператор CREATE TABLE

```
CREATE TABLE имя_таблицы ( {имя_столбца тип [NOT NULL]},... )
```

Создается новая таблица с именем имя_таблицы, включающая столбцы с указанными именами и типами данных. Необязательная фраза NOT NULL запрещает помещать в базу данных кортежи с незадаанными значениями соответствующих компонентов.

Пример.

```
CREATE TABLE Покупатели  
    (Фамилия CHAR (10) NOT NULL,  
     Адрес CHAR (20) NOT NULL,  
     Баланс NUMERIC (10,2)).
```

Оператор создает новую пустую таблицу Покупатели со столбцами Фамилия, Адрес, Баланс. Фамилии и адреса покупателей должны присутствовать во всех строках таблицы. Баланс указывать не обязательно.

Оператор ALTER TABLE

```
ALTER TABLE имя_таблицы  
    ( { {ADD|MODIFY|DROP} имя_столбца [тип] [NOT NULL]},... )
```

Изменяет указанную таблицу. Варианты ADD,MODIFY, DROP означают соответственно добавление, модификацию, удаление указанного столбца.

Пример.

```
ALTER TABLE Покупатели  
      (ADD Телефон CHAR (7))
```

Добавляет к таблице Покупатели столбец Телефон типа CHAR (7).

Оператор CREATE INDEX

Индекс используется для ускорения поиска в таблицах, состоит из последовательности пар (ключ кортежа, адрес кортежа), заполняется и обслуживается СУБД.

```
CREATE [UNIQUE] INDEX имя_индекса ON имя_таблицы  
      ({имя_столбца [ASC|DESC]},...)
```

Создает индекс с указанным именем для таблицы имя_таблицы. Ключ индекса состоит из всех перечисленных атрибутов (имен столбцов).

Необязательное слово UNIQUE запрещает дублирование ключей, слова ASC, DESC задают тип сортировки значений ключа (по возрастанию или убыванию атрибута). Если тип сортировки не задан, то предполагается ASC.

Пример.

```
CREATE INDEX Инд_Покуп ON Покупатели (Фамилия).
```

Создает индекс Инд_Покуп для таблицы Покупатели. Ключ индекса содержит единственный атрибут Фамилия. Разрешено дублирование ключей, значения ключа упорядочены по возрастанию.

Оператор CREATE VIEW

```
CREATE VIEW имя_представления  
      [(список_столбцов)] AS оператор-SELECT
```

Создает таблицу - представление имя_представления, основанную на результате выполнения оператора SELECT. Параметр список_столбцов определяет структуру таблицы. Если список не задан, то используются имена столбцов из запроса.

Операторы DML

Оператор UPDATE

```
UPDATE имя_таблицы  
    {SET имя_столбца = {выражение | NULL}},...  
    [WHERE условие]
```

Изменяет значения указанных столбцов таблицы в строках, удовлетворяющих условию. Слово NULL означает отсутствие значения для столбца. Если условие не задано, то обновляются все строки таблицы.

Пример.

```
UPDATE Покупатели  
    SET Баланс = Баланс+100  
    WHERE Баланс<0
```

Обновляется таблица Покупатели. К каждой строке таблицы с отрицательным балансом добавляется 100.

Оператор INSERT

Имеются две формы оператора.

```
INSERT INTO имя_таблицы (1)  
    [(список_столбцов)] VALUES (список_значений)
```

```
INSERT INTO имя_таблицы (2)  
    [(список_столбцов)] предложение-SELECT
```

Первая форма вводит в таблицу новые строки с указанными значениями полей. Если столбцы не указаны, то используются все столбцы таблицы.

Вторая форма добавляет в таблицу строки, отобранные предложением SELECT. Если имена столбцов не указаны, то используются имена из предложения.

Пример.

```
INSERT INTO Покупатели VALUES («Петров», «Адрес», 100.00)
```

Добавляет к таблице Покупатели одну запись с атрибутами покупателя Петрова.

Оператор DELETE

DELETE FROM имя_таблицы [WHERE условие]

Удаляет удовлетворяющие условию строки. Если условие не указано, то удаляются все строки таблицы.