

Практическая работа №10(4)

Создание запросов SQL на добавление, изменение и удаление записей

Цель занятия: Научиться создавать SQL запросы на добавление, изменение и удаление записей.

Теоретическая часть

Запросы на добавление

Добавление осуществляется командой INSERT. Синтаксис команды INSERT:

INSERT INTO <имя_таблицы>

[(<список столбцов>)]

VALUES

(DEFAULT | NULL | <выражение>) | <результатирующая_таблица>

Рассмотрим назначение каждого из аргументов.

<имя_таблицы> — имя таблицы, в которую необходимо вставить данные;

<список_столбцов> — содержит список столбцов, в которые будет производиться вставка данных. Если он опущен, то данные будут вставляться последовательно во все столбцы, начиная с первого.

VALUES (DEFAULT / NULL | <выражение>) - определяет набор данных, которые будут вставлены в таблицу. Количество аргументов **VALUES** определяется количеством столбцов в таблице или количеством столбцов в списке (если таковой имеется). Для каждого столбца таблицы можно указать один из трех возможных вариантов:

- **DEFAULT** — будет вставлено значение по умолчанию, определенное для столбца. Если для столбца разрешено хранение значений NULL, а значение по умолчанию не определено, то в столбец будет вставлено значение NULL.
- **NULL** — в столбец будет вставлено значение NULL. Естественно, вставка таких значений будет успешной, если для столбца была разрешена возможность хранения значений NULL. Следует помнить, что для столбцов, входящих в первичный ключ, возможность хранения значений NULL не предусмотрена.
- **<выражение>** — задает значение, которое будет вставлено в столбец таблицы. Этот параметр должен иметь тот же тип данных, что и столбец, а также удовлетворять ограничениям целостности, определенным для соответствующего столбца.

<результатирующая_таблица> — этот параметр подразумевает указание запроса SELECT, с помощью которого будет формироваться набор данных, вставляемых в таблицу.

Пример 1. Добавить информацию о преподавателе.

INSERT INTO Преподаватели (Фамилия, Имя, Отчество, Стаж, Разряд, Категория)

VALUES ('Петров', 'Петр', 'Петрович', 10, 4, 'Высшая');

Создание запроса на добавление с помощью конструктора

ЗЕМЕЧАНИЕ: Access не имеет средств для создания типовых запросов на добавления, он позволяет только добавлять информацию из уже существующих таблиц, поэтому типовые запросы на добавления нужно формировать в ручном режиме, что учтено в приведенном алгоритме его создания.

Создания запроса на добавление осуществляется следующим образом:

1. На ленте **Создание**, в разделе **Запросы** нажмите кнопку **Конструктор запросов**.
2. Отказаться от добавления таблицы (возможности конструктора не используются).
3. Перейти в режим SQL.
4. Ввести текст запроса на добавление.
5. Сохраните полученный запрос и дайте ему имя.

Запросы на изменение

Для внесения изменений в данные таблиц служит команда **UPDATE**, позволяющая выполнять как простое обновление данных в столбце, так и сложные операции модификации данных во множестве строк таблицы. Рассмотрим синтаксис этой команды:

```
UPDATE <имя_таблицы>  
SET { <имя_столбца> = { <выражение> | DEFAULT | NULL } }  
{ [ FROM { <имя_таблицы> ] }  
[ WHERE <условие_отбора> ] }
```

Рассмотрим назначение каждого из аргументов.

<имя_таблицы> — имя таблицы, в которой необходимо произвести изменение данных.

SET — с этого ключевого слова начинается блок, в котором определяется список изменяемых столбцов. За один вызов **UPDATE** можно изменить данные в нескольких столбцах множества строк одной таблицы.

<имя_столбца> = {<выражение> | DEFAULT | NULL} — для каждого изменяемого столбца нужно задать значение, которое он примет после выполнения изменения. С помощью ключевого слова **DEFAULT** можно присвоить столбцу значение, определенное для него по умолчанию. Можно также установить для столбца значение **NULL**. Изменению подвергнутся все строки, удовлетворяющие критериям ограничения области действия запроса **UPDATE**, которые задаются с помощью раздела **WHERE**.

FROM {<имя_таблицы>} — если при изменении данных в таблице необходимо учесть состояние данных в других таблицах, то эти источники данных необходимо указать в разделе **FROM**.

WHERE <условие_отбора> — назначение раздела **WHERE**, используемого в запросе **UPDATE**, полностью соответствует назначению, которое раздел имеет в запросе **SELECT**. Если в контексте значений строки указанное логическое условие выполняется, т. е. возвращает значение **TRUE**, то данные этой строке будут изменены. В противном случае изменение не выполняется. Предполагается, что логическое условие включает имена столбцов изменяемой таблицы, однако это необязательно.

Пример 2. Увеличить стаж преподавателей на 1.

```
UPDATE Преподаватели  
SET Стаж=Стаж+1
```

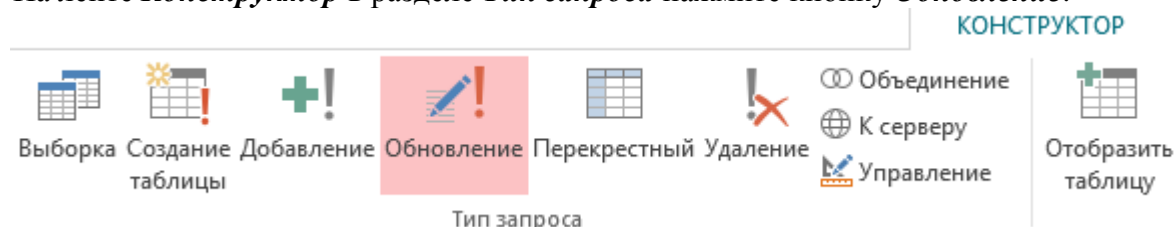
Пример 3. Изменить ФИО преподавателя Петрова на ФИО Сидорова.

```
UPDATE Преподаватели  
SET Фамилия="Сидорова"  
WHERE Фамилия="Петрова"
```

Создание запроса на обновление с помощью конструктора

Создания запроса на обновление осуществляется также как и обычного запроса и выполняется следующим образом:

1. На ленте **Создание**, в разделе **Запросы** нажмите кнопку **Конструктор запросов**.
2. Добавьте таблицу, которую будете использовать в запросе.
3. Двойным щелчком мыши выберите из таблицы поля, которые будут использоваться в запросе.
4. На ленте **Конструктор** в разделе **Тип запроса** нажмите кнопку **Обновление**.



5. В строке **Обновление** укажите новое значение поля. В примере ниже изменяется название специальности, которая вводится с клавиатуры как параметр.
6. В строке **Условия отбора** укажите условие для нужного поля. Обновляются только те записи, которые соответствуют всем условиям в строке **Условия отбора**. В примере ниже изменять специальность будет только для заданной группы.
7. Сохраните полученный запрос и дайте ему имя.

Поле:	НомГруппы	КодСпециальности	
Имя таблицы:	Группы	Группы	
Обновление:		[Новая специальность]	
Условие отбора:	[Группа]		
или:			

**UPDATE Группы SET Группы.КодСпециальности = [Новая специальность]
WHERE (((Группы.НомГруппы)=[Группа]));**

Запросы на удаление записей

При удалении записей с помощью запроса на удаление отменить операцию невозможно. Поэтому, прежде чем выполнить такой запрос, необходимо просмотреть выбранные для удаления данные. Для этого просмотрите запрос в режиме таблицы. Рекомендуется всегда делать резервные копии данных. Таким образом, если были удалены не те записи, их можно восстановить из резервных копий. Иногда при выполнении запроса на удаление также могут быть удалены и некоторые записи из связанных таблиц, не включенных в запрос. Это случается, если в запрос включена только таблица, находящаяся на стороне «один» отношения «один-ко-многим», и для этого отношения было установлено каскадное удаление. При удалении записей из данной таблицы будут также удаляться записи из таблицы, находящейся на стороне «многие».

Чаще всего удаление данных выполняется с помощью команды DELETE, удаляющей строки таблицы.

Синтаксис команды, чаще всего использующийся на практике, следующий:

DELETE <Имя_таблицы>
[WHERE <Условие_отбора>]

Таким образом, в большинстве случаев требуется указание лишь имени таблицы, из которой необходимо удалить данные, и логического условия, ограничивающего диапазон удаляемых строк. Причем последнее вовсе не обязательно, и при отсутствии условия из таблицы будут удалены все имеющиеся строки. Как и при выборке и изменении строк, диапазон удаляемых строк формируется с помощью раздела WHERE.

Пример 4. Удалить преподавателя Петрова

DELETE Преподаватели
WHERE Фамилия="Петров"

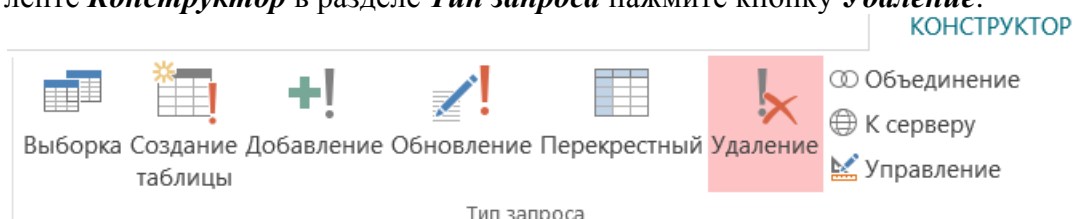
Пример 5. Удалить всю информацию из таблицы преподаватели.

DELETE Преподаватели

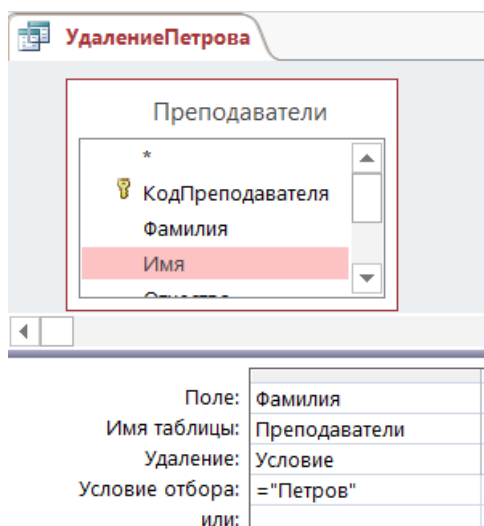
Создание запроса на удаление с помощью конструктора

Создания запроса на удаление осуществляется также, как и обычного запроса и выполняется следующим образом:

1. На ленте **Создание**, в разделе **Запросы** нажмите кнопку **Конструктор запросов**.
2. Добавьте таблицу, которую будете использовать в запросе.
3. Двойным щелчком мыши выберите из таблицы поля, которые будут использоваться для условия отбора при удалении.
4. На ленте **Конструктор** в разделе **Тип запроса** нажмите кнопку **Удаление**.



5. В строке **Условия отбора** укажите условие для нужного поля. Удаляются только те записи, которые соответствуют всем условиям в строке **Условия отбора**. В примере ниже изменять специальность будет только для заданной группы.
6. Сохраните полученный запрос и дайте ему имя.



ЗАМЕЧАНИЕ: При использовании конструктора SQL код запроса на удаление использует альтернативную форму задания оператора удаления.

DELETE Преподаватели.Фамилия
FROM Преподаватели
WHERE (((Преподаватели.Фамилия)="Петров"));

Если вы желаете, то можете запрос в режиме SQL набрать самостоятельно используя классическую форму оператора удаления.

DELETE Преподаватели
WHERE Фамилия="Петров"

Практическое задание

1. Создайте копию файла БД предыдущей работы и дайте ему имя Работа№10.
2. Удалите элементы запросов из БД.
3. Создайте копию таблицы Экзамен, назовите Экзамен1.
4. Создайте запрос на удаление записей из таблицы «Экзамен1» тех студентов, которые не имеют допусков. Сохраните запрос и полученный список в текстовый файл. Имя запроса – УдалениеБезДопускаЭкзамен.
5. Создайте копию таблицы Зачет, назовите Зачет1.
6. Создайте запрос на удаление записей из таблицы Зачет1» тех студентов, которые сдавали 2 года назад. Сохраните запрос и полученный список в текстовый файл. Имя запроса – УдалениеЗачетов2года.
7. Создать запрос на добавление сданных зачетов – Данилов, Химия, 4 семестр, допущен. Сохраните запрос и полученный список в текстовый файл. Имя запроса – ДобавитьДанилова.
8. Создать запрос на добавление информации о родителях – ФИО, Место работы, телефон. Сохраните запрос и полученный список в текстовый файл. Имя запроса – ДобавитьРодителя.
9. Создайте копии таблиц Группы, Учащиеся, Студенты. В названии копий добавьте цифру 1.
10. Создайте связи между таблицами Группы1, Учащиеся1 и Студенты1.
11. Создайте запрос на удаление учащихся заданной группы из таблицы Учащиеся1. Распустите учащихся группы П-41. Сохраните запрос и полученный список в текстовый файл. Имя запроса – РоспускГруппы.
12. Создайте запрос на удаление студентов, которые не учатся ни в одной из групп в таблице Студенты1. Сохраните запрос и полученный список в текстовый файл. Имя запроса – ВыпускСтудентов.
13. Создайте запрос на перевод учащихся из одной группы в другую группу в таблице Учащиеся1. Переведите учащихся группы П-31 в группу П-41. Сохраните запрос и полученный список в текстовый файл. Имя запроса – ПереводМеждуГруппами.
14. Оформите отчет.

КОНТРОЛЬНЫЕ ВОПРОСЫ

1. В чем отличие запросов на добавление, изменение, удаление данных?
2. При помощи Справки Access опишите запросы на объединение данных?
3. Как уберечь базу данных от некорректно построенных запросов на удаление, изменение, добавление данных?
4. В каких случаях можно применять запросы на удаление и добавление данных?