Лабораторная работа №4

Тема: Изменение данных и структуры БД. Клиентский интерфейс для БД.

Цель: Развитие навыков программирования приложений, использующих БД, знакомство с частями SDL и DML языка SQL.

Навыки и умения: модификация данных и определение структуры БД с помощью SQL, использование инструментария MS Access (редактор макросов, VBA модули, конструктор форм), написание клиентского интерфейса.

Теоретическая часть

Структура языка SQL

Все операторы языка SQL можно условно разделить на три группы операторов. Оператор языка запросов – SELECT (был рассмотрен в предыдущих лабораторных работах), операторы языка манипуляции данными (Insert, Update, Delete) и операторы языка определения данных (Create, Drop, Alter).

Запросы DML (ЯМД)

К запросам языка манипуляции данными (Data Manipulation Language) относятся запросы на добавление, удаление и модификацию кортежей.

Добавление кортежа производится командой:

INSERT INTO имя_таблицы [(<список столбцов>)] VALUES (<список значений>)

Список столбцов и список значений указываются через запятую, а значения добавляются в соответствующие столбцы. Если необходимо добавить кортеж целиком (т.е. значения есть для всех полей и их порядок совпадает с порядком полей в отношении), то описание списка столбцов можно опустить.

Пример 1:

Три следующих запроса будут верно исполнены для отношения R1:

INSERT INTO R1(ФИО, Дисциплина, Оценка) VALUES («Попова», «БД», 3);

INSERT INTO R1 VALUES («Попова», «Моделирование», 3);

INSERT INTO R1(ФИО, Дисциплина) VALUES («Бурковский», «Сети ЭВМ»);

Оператор **удаления данных** DELETE позволяет удалить одну или несколько строк из таблицы в соответствии с условиями, которые задаются для удаляемых строк. Синтаксис оператора DELETE следующий:

DELETE FROM <имя_таблицы> [WHERE <условия_отбора>]

Если условия отбора не задаются, то из таблицы удаляются все строки. Операция **обновления данных** UPDATE требуется тогда, когда происходят изменения данных, которые надо отразить в базе данных.

Запрос на обновление может изменить сразу целую группу записей. Этот запрос состоит из трех частей:

- Предложение **UPDATE**, которое указывает на обновляемую таблицу;
- Предложение **SET**, задающее данные для обновления;
- Необязательный критерий **WHERE**, ограничивающий число записей, на которые воздействует запрос на обновление.

Пример 2:

Изменить на 3 оценку по дисциплине «БД» у студента Миронова в таблице R1:

UPDATE R1 SET R1.Оценка = 3
WHERE R1.ФИО = «Миронов» AND R1.Дисциплина = «БД»;

Запросы SDL (ЯОД)

Команды языка определения схемы данных (Schema Definition Language – SDL) представляют собой инструкции SQL, которые позволяют создавать и модифицировать элементы структуры базы данных. Например, используя SDL, можно создавать, удалять таблицы и изменять их структуру, создавать и удалять индексы.

Создание таблицы. Оператор создания таблицы имеет следующий вид:

CREATE TABLE <имя таблицы> (<имя столбца> <тип данных> [NOT NULL] [,<имя столбца> <тип данных> [NOT NULL]] …)

Обязательными операндами оператора являются имя создаваемой таблицы и имя хотя бы одного столбца (поля) с указанием типа данных, хранимых в этом столбце.

При создании таблицы для отдельных полей могут указываться некоторые дополнительные правила контроля вводимых в них значений. Например, конструкция NOT NULL (не пустое) служит для определения обязательного поля.

В табл. 1 перечислены типы данных, которые можно использовать при создании таблиц, используя Microsoft Jet SDL и предложение CREATE (СУБД Access).

Тип данных	SQL тип
Счетчик	COUNTER
Текстовый	TEXT
Memo	LONGTEXT
Денежный	CURRENCY
Дата/время	DATETIME
Числовой (одинарное с плавающей точкой)	SINGLE
Числовой (двойное с плавающей точкой)	DOUBLE
Числовой (целое)	INTEGER
Числовой (длинное целое)	LONG
Числовой (байт)	BYTE

С помощью конструкции **CONSTRAINT** можно задать первичный ключ таблицы.

Пример 3:

Создание таблицы TABL1:

```
CREATE TABLE TABL1 (
[FIL1] COUNTER,
[FIL2] TEXT (10),
[FIL3] CURRENCY, [FIL4] DATETIME,
[FIL5] BYTE, [FIL6] INTEGER,
[FIL7] SINGLE, [FIL8] LONG,
[FIL9] DOUBLE,
CONSTRAINT PrimaryKey PRIMARY KEY ([FIL1]) );
```

В примере 3 поле FIL1 объявлено ключевым, для данного поля создан индекс с именем PrimaryKey.

Похожим образом задается внешний ключ:

Пример 4:

Создание таблицы TABL2:

```
CREATE TABLE TABL2 (
[FIL1] INTEGER,
[FIL2] TEXT (10) NOT NULL,
[FIL3] CURRENCY, [FIL4] LONGTEXT,
CONSTRAINT PrimaryKey PRIMARY KEY ([FIL1],[FIL2]),
CONSTRAINT ForeignKey FOREIGN KEY ([FIL1])
REFERENCES TABL1 ([FIL1]));
```

В данной таблице поле FIL1 объявлено внешним ключом. Между таблицами TABL1 и TABL2 устанавливается связь «один-ко-многим» по полю FIL1.

Для **удаления таблиц** служит инструкция

DROP TABLE <имя таблицы>

Для **модификации структуры таблицы** (добавление, удаление полей, изменения типов полей) используется оператор ALTER TABLE изменения структуры таблицы имеет следующий вид:

ALTER TABLE <имя таблицы>MODIFY | ADD | DROP <имя поля> [<тип данных>]

Создание индексов. Помимо создания индексов в процессе формирования таблицы (с помощью предложения CONSTRAINT), можно также создавать индексы уже после того, как таблица сформирована:

CREATE [UNIQUE] INDEX <имя индекса>ON <имя таблицы> (<имя столбца> [ASC | DESC] [, <имя столбца> [ASC | DESC]…)

Этот оператор позволяет создать индекс для одного или нескольких столбцов заданной таблицы с целью ускорения выполнения запросных и поисковых операций с таблицей. Для одной таблицы можно создать несколько индексов.

Для **удаления индексов** служит инструкция

DROP INDEX <имя индекса> ON<имя таблицы>

Создание форм и отчетов в среде MS Access

СУБД MS Access предоставляет программисту инструментарий для создания форм и отчетов (для пользователя). Соответствующие объекты можно найти среди объектов БД. Доступно как создание форм (отчетов) по определенным таблицам (запросам), так и самостоятельное создание в режиме дизайнера.

Встроенный язык Visual Basic for Application

СУБД MS Access предоставляет возможность описания процедур на языке высокого уровня Visual Basic for Application (VBA). Этот язык встроен

во все программные средства, относящиеся к MS Office, и он позволяет работать с объектами БД через выполнение SQL-запросов. Язык VBA является родственником VB и Basic. Также этот язык является процедурным, поддерживает деление на модули, поддерживает дизайнер форм. Обеспечивает обработку исключительных ситуаций и выполнение транзакций. Процедуры и функции, описанные в области видимости public, могут также быть использованы при построении SQL-запросов, подобно встроенным функциям СУБД.

Создать модули можно на соответствующей странице объектов MS Access.

Задание на лабораторную работу

1. Создать базу данных по предметной области своего варианта, которая должна минимум содержать таблицу, состоящую минимум из 6 полей. Реализовать кодовые поля в основной таблице и справочник(и) для расшифровки этих полей. Для создания таблиц БД **использовать скриптовый файл или макрокоманду**, содержащую набор SQL-команд из части языка SDL;
2. Реализовать процедуры Добавления, Удаления, Поиска и Изменения, с помощью SQL;
3. Организовать оконный интерфейс для функций, созданных на предыдущем этапе (добавления, удаления, поиска и изменения);
4. Поиск должен осуществляться с использованием индексов, т.е. поля, по которым осуществляется поиск, должны быть проиндексированы.
5. Организовать вывод результатов в виде отчетов.

Прием работы

Прием происходит при наличии оформленного отчета и работающей БД, созданной в среде MS Access.

Вопросы

1. На какие части можно разделить язык SQL, какие команды им соответствуют?
2. Для чего используются индексы?
3. Как обновить несколько полей для нескольких кортежей таблицы одним запросом?
4. Что определяет ключевое слово Constraint?
5. Что такое VBA?
6. Можно ли выполнить добавление данных без указания названия полей, в которые добавляются значения? (почему нельзя или как можно)
7. Как создать форму в MS ACCESS?
8. Как создать отчет в MS ACCESS?