



Министерство образования  
Республики Беларусь

БЕЛОРУССКИЙ НАЦИОНАЛЬНЫЙ  
ТЕХНИЧЕСКИЙ УНИВЕРСИТЕТ

---

---

Кафедра «Экономика и управление научными  
исследованиями, проектированием и производством»

# СЕТЕВЫЕ ТЕХНОЛОГИИ И БАЗЫ ДАННЫХ

Лабораторный практикум

Часть 1

СУБД MS ACCESS

Минск  
БНТУ  
2011

Министерство образования Республики Беларусь  
БЕЛОРУССКИЙ НАЦИОНАЛЬНЫЙ ТЕХНИЧЕСКИЙ  
УНИВЕРСИТЕТ

---

Кафедра «Экономика и управление научными  
исследованиями, проектированием и производством»

## СЕТЕВЫЕ ТЕХНОЛОГИИ И БАЗЫ ДАННЫХ

Лабораторный практикум

В 3 частях

Часть 1

СУБД MS ACCESS

Минск  
БНТУ  
2011

УДК 004.65 (076.5)

ББК 32.97я7

С 33

Составитель *А.Г. Ляхевич*

Рецензенты:

*А.А. Дудкин, П.В. Мелюшин*

С 33 Сетевые технологии и базы данных: лабораторный практикум: в 3 ч. / сост. А.Г. Ляхевич. – Минск: БНТУ, 2011. – Ч. 1: СУБД MS ACCESS. – 45 с.

Практикум предназначен для студентов специальности 1-27 01 01 «Экономика и организация производства» направления 1-27 01 01-08 «Приборостроение» и специальности 1-26 02 02 «Менеджмент» направления 1-26 02 02-03 «Инновационный менеджмент». Первая часть практикума знакомит студентов с основами работы с СУБД MS ACCESS и языком SQL.

ISBN 978-985-525-529-2 (Ч. 1)

ISBN 978-985-525-530-8

© БНТУ, 2011

## Лабораторная работа № 1

### ОСНОВНЫЕ ПРИЕМЫ РАБОТЫ С MS ACCESS. СОЗДАНИЕ ТАБЛИЦ И СХЕМЫ ДАННЫХ

#### Цели работы:

- изучение интерфейса СУБД MS Access;
- получение навыков создания базы данных в СУБД MS Access;
- получение навыков создания и изменения таблиц базы данных;
- получение навыков создания связей «первичный – внешний ключ» между таблицами реляционной базы данных.

#### Теоретический материал

Создание таблицы возможно одним из следующих способов:

- **режим таблицы** удобнее всего. Создание таблицы осуществляется путем ввода данных в пустую таблицу. После сохранения таблицы остаются только те столбцы, которые были переименованы или в которые были введены данные (рис. 1.1);

Фамилия	Имя	Отчество	Адрес	Поле5
Иванов	Иван	Иванович	МКАД	
Кейрос	Диего			

Рис. 1.1. Создание таблиц в режиме таблицы

- **режим конструктора** предназначен для создания таблицы при помощи графического интерфейса. Приводится список столбцов таблицы: указывается имя поля и тип данных, длина поля, является ли поле первичным ключом таблицы, допустимо ли указание в поле пустых значений и т. д. (рис. 1.2).

Тип данных «Поле объекта OLE» позволяет хранить в базе данных объекты, обрабатываемые другими приложениями, например, документы Word, Excel, изображения, видеофайлы и т. п. Для добавления объекта необходимо переключиться в режим отображения данных таблицы и в контекстном меню выбрать пункт «Добавить объект» (рис. 1.3).

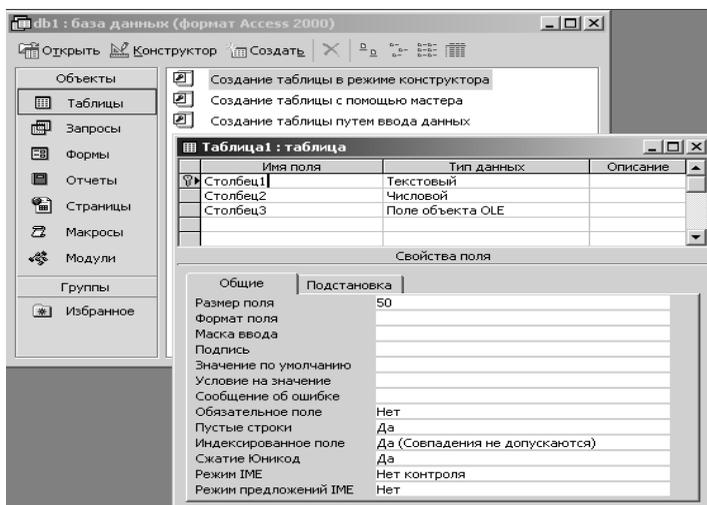


Рис. 1.2. Создание таблиц в режиме конструктора

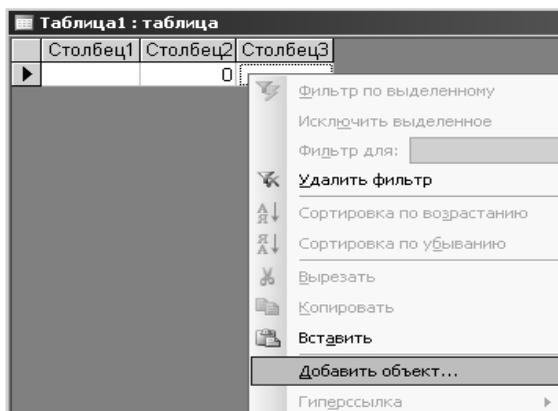


Рис. 1.3. Добавление объекта OLE

- **мастер таблиц** – это создание таблицы при помощи мастера на основе шаблонов других таблиц;

- **импорт таблиц** – импорт таблиц из других файлов Access или файлов других форматов (MS Excel, СУБД Visual FoxPro и др.). При импорте в текущей базе данных просто создается копия соответствующих таблиц, а связь с ними отсутствует. Для выполнения импорта используется команда меню «Файл/Внешние данные/Импорт». Импорт таблицы Excel можно осуществить и просто вставив ее через контекстное меню (вызывается на белом фоне в списке таблиц);

- **связь с таблицами** – позволяет выполнить те же действия, что и «Импорт таблиц» с той разницей, что сама таблица фактически останется в другой базе данных, а в текущей БД Access будет просто ссылка на эту таблицу. Используя «Связь с таблицами», можно превратить MS Access в клиентскую программу для доступа к СУБД MS SQL Server, Oracle или другой базе данных, имеющей драйвер ODBC (меню «Файл/Внешние данные/Связь с таблицами/Тип файлов/Базы данных ODBC/кнопка New»).

При создании таблицы в режиме конструктора или при последующем редактировании таблицы можно создать столбец подстановок. Столбец подстановок – это столбец, в который значения вводятся не вручную, а из раскрывающегося списка. В качестве источника данных списка может выступать столбец другой таблицы или фиксированный набор значений пользователя. Создание столбца подстановок: открыть таблицу в режиме конструктора, выбрать нужный столбец и вкладку «Подстановка». Указать «Тип элемента управления» – «Поле со списком», указать «Тип источника строк» – «Таблица или запрос», рядом с полем «Источник строк» нажать кнопку с тремя точками, сформировать запрос в появившемся диалоговом окне «Инструкция SQL:построитель запросов», закрыть его, в ответ на запрос сохранения инструкции SQL указать «Да» (рис. 1.4). Для ввода фиксированного набора значений в поле «Тип источника строк» указывается «Список значений»,

а сами значения вводятся в поле «Источник строк», разделяя разные значения точкой с запятой. После вставки столбца подстановок в него можно будет вводить значения вручную или выбирать их из раскрывающегося списка. Для запрета ручного ввода значений, не совпадающих со списком, необходимо открыть таблицу в режиме конструктора, выбрать столбец подстановки и на вкладке «Подстановка» в поле «Ограничиться списком» указать «Да».

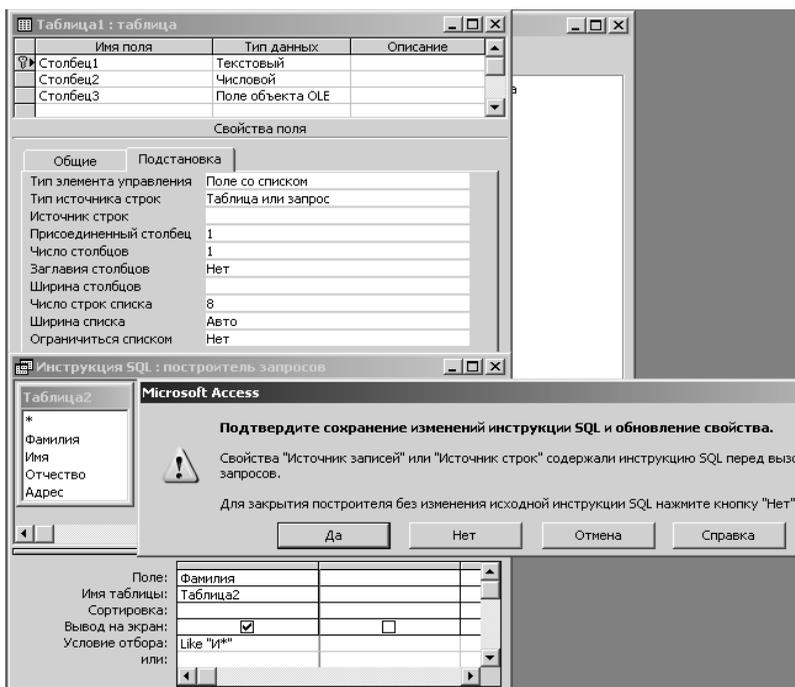


Рис. 1.4. Создание столбца подстановок

**Схема данных** позволяет определить связи между таблицами, используя понятия «первичный ключ» и «внешний ключ» реляционной модели данных. Она также позволяет установить правила каскадного удаления и обновления строк в связанных таблицах. Для создания схемы данных – «Сервис/Схема дан-

ных» – при помощи графического интерфейса установить связи между таблицами, перетянув поле одной таблицы на связываемое поле другой таблицы (рис. 1.5).

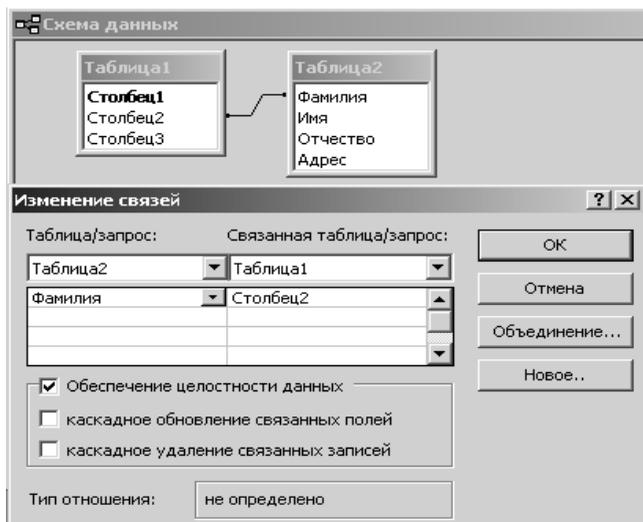


Рис. 1.5. Схема данных

Если установить флажок «Обеспечение целостности данных», то Microsoft Access не позволяет добавлять в связанную таблицу записи, для которых нет соответствующих записей в главной таблице или же изменять (удалять) записи в главной таблице, для которых имеются подчиненные записи в связанной таблице. Если дополнительно установить флажок «Каскадное обновление» связанных полей, любое изменение значения в ключевом поле главной таблицы приведет к автоматическому обновлению соответствующих значений во всех связанных записях. Постановка флажка «Каскадное удаление» приведет к автоматическому удалению связанных записей в подчиненной таблице при удалении записи в главной таблице. Создавая связи, необходимо помнить о реляционной модели данных: связи устанавливаются не между любыми полями, а

только между ключевым полем таблицы и внешним ключом в другой таблице (некоторым полем во второй таблице, содержащим те же значения, что и ключевое поле первой таблицы, не обязательно все значения). Тип создаваемой в схеме данных связи зависит от полей, для которых определяется связь:

1) отношение «один-ко-многим» создается в том случае, когда только одно из полей является ключевым или имеет уникальный индекс;

2) отношение «один-к-одному» создается в том случае, когда оба связываемых поля являются ключевыми или имеют уникальные индексы.

### Задания для выполнения

**Задание 1.** Создать в MS Access таблицы следующего содержания (табл. 1.1–1.3):

Таблица 1.1

#### Склад

Изделие	Цена	Наличие	Поставщик	Примечание
F117A	10000000	1	д. Гадюкино	–
F14	1000000	20	корпорация Макдоннел Дуглас	–
F15	3000000	15	корпорация Макдоннел Дуглас	–
F16	2000000	10	корпорация Макдоннел Дуглас	–
Палочки для воды	100	5	–	–
Палочки для еды	1	10000	Бьенхоа	–

Первичным ключом таблицы является столбец «Изделие». Не допускается вводить в столбцы «Наличие» отрицательные значения (выводится предупреждающее сообщение).

Таблица 1.2

### Заказ

№ заказа	Заказчик	Изделие	Количество
1	ВВС США	F14	1
2	Ильич Рамирес Санчес	F117A	2
3	ВВС США	F14	130
4	ВВС Ирана	F16	1
5	Петрович	Палочки для воды	15
6	Ричард Роу	Палочки для еды	1000

Первичным ключом таблицы является столбец «№ заказа». Столбец «Изделие» является столбцом подстановки, данные для которого берутся из столбца «Изделие» таблицы «Склад» (запрещается вводить в столбец значения, не перечисленные в столбце «Изделие» таблицы «Склад»). В столбец «Количество» запрещается вводить отрицательные значения.

Таблица 1.3

### Оплата

№ заказа	Форма расчетов	Оплата
1	Безналичный перевод	1000000
2	Наличные	0
3	Безналичный перевод	10000000
4	Казначейские облигации США	2000000
5	Бартер	1500

Первичным ключом таблицы является столбец «№ заказа». Столбец «Форма оплаты» является столбцом подстановки, данные для которого представляют список фиксированных значений (примеры всех значений приведены в таблице). В столбец «Форма оплаты» разрешается вводить и значения, не указанные в списке.

**Задание 2.** Выполнить следующие изменения структуры таблицы «Склад»:

- удалить из таблицы «Склад» столбец «Примечание»;
- добавить в таблицу «Склад» поле «Документация», имеющее тип данных «Поле объекта OLE». Используя меню «Вставить/Объект», вставить по строчкам этого столбца документы Word и Excel. Должны быть вставлены как документы уже имеющиеся на диске (со связью и без установления связи с документом на диске), так и заново созданные документы;
- установить для столбца «Цена» таблицы «Склад» значение по умолчанию 0.25, а также запрет на ввод отрицательных чисел с выводом соответствующего предупреждения при неправильном вводе данных.

**Задание 3.** При помощи меню «Сервис/Схема данных» установить связь между таблицами:

- столбец «Заказ» таблицы «Изделие» является внешним ключом для таблицы «Склад». При попытке удалить изделие, по которому существует заказ, удаление изделия блокируется;
- столбец «№ заказа» таблицы «Оплата» является внешним ключом для таблицы «Заказ». При обновлении или удалении записей в таблице «Заказ» соответствующие изменения происходят в таблице «Оплата».

## Лабораторная работа № 2

### ОСНОВНЫЕ ПРИЕМЫ РАБОТЫ С MS ACCESS. СОЗДАНИЕ ЗАПРОСОВ, ИСПОЛЬЗОВАНИЕ ЯЗЫКА SQL В ЗАПРОСАХ

#### Цели работы:

- получение навыков создания запросов в СУБД MS Access;
- получение навыков использования языка SQL в реляционной базе данных.

#### Теоретический материал

Наиболее гибким вариантом является создание запроса в режиме конструктора: задается имя таблицы, имена отбираемых столбцов, условия отбора, сортировки и вывода на экран. Условия, находящиеся в одной строке конструктора, объединяются логическим оператором «и», а в разных строках – оператором «или» (рис. 2.1). Если флажок «Вывод на экран» снят, то сам столбец на экран не выводится, но условие, записанное в нем, продолжает действовать. Возможно создание вычисляемых столбцов в формате «Название столбца: Вычисления». Названия столбцов заключаются в квадратные скобки, если в названии содержится хотя бы один пробел. Не следует выносить на панель запроса более одной таблицы, если вы не собираетесь объединять их. Таблицы объединяются только по полям с совпадающими данными. Для успешного объединения рекомендуется непосредственно указать эти столбцы, протянув мышью связь от одного столбца к другому.

Используя команду меню «Вставка/Запрос», можно также создать запрос в режиме мастера запросов по одному из следующих вариантов:

- *простой запрос* позволяет указать таблицы и имена столбцов, включаемые в результаты запроса;
- *перекрестный запрос* напоминает сводную таблицу Excel (необходимо указать строки, столбцы, данные и функцию об-

работки данных). Для создания перекрестного запроса на основании нескольких таблиц надо сначала создать простой запрос, содержащий все необходимые столбцы из этих таблиц;

- **повторяющиеся записи** выводит повторяющиеся записи из таблицы: указывается столбец, по которому надо искать повторы, а также другие столбцы. Если дополнительные столбцы не указаны, то выводится количество повторов;

- **записи без подчиненных** – поиск записей, не имеющих связей с записями в другой таблице (например, клиенты, не имеющие заказов).

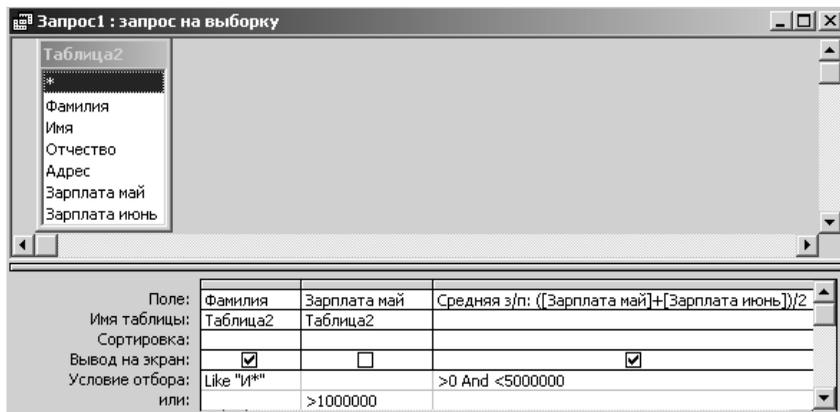


Рис. 2.1. Запрос в режиме конструктора

Помимо создания запроса одним из вышеперечисленных способов можно задать запрос и непосредственно, в виде инструкций SQL. SQL (Structured Query Language – структурированный язык запросов) – это язык, предназначенный для выборки и обработки информации, содержащейся в реляционной базе данных. Для перехода из режима конструктора запроса в режим SQL необходимо воспользоваться командой меню «Вид/Режим SQL».

Запросы на выборку позволяют получить информацию из одной или нескольких таблиц и оформляются ключевым сло-

вом SELECT. Поскольку инструкции SQL могут читаться как обычные предложения английского языка, то дальнейшее пояснение инструкций будет даваться путем перевода соответствующих ключевых слов на русский язык. В большинстве случаев это достаточно для понимания инструкции.

### *Простой запрос на выборку*

```
SELECT DISTINCT столбец1, (столбец2+столбец3)*5
FROM Таблица1
WHERE (столбец1>10) OR (столбец4 LIKE 'М_нск%')
AND NOT (столбец3 IN 10,20,30)
OR (столбец5 IS NULL) OR (столбец6 BETWEEN 100
AND 200)
OR ((столбец3 - столбец2) IS UNKNOWN) OR (столбец7 IS
FALSE)
ORDER BY столбец1, столбец2 ASC
```

Русский перевод ключевых слов инструкции:

```
ОТОБРАТЬ ОТЛИЧАЮЩИЕСЯ_ЗНАЧЕНИЯ столбец1,
(столбец2+столбец3)*5
ИЗ Таблица1
ГДЕ (столбец1>10) ИЛИ (столбец4 ПОХОЖ_НА 'М_нск%')
И НЕ_ВЕРНО_ЧТО (столбец3 СРЕДИ_ЗНАЧЕНИЙ 10,20,30)
ИЛИ (столбец5 НЕ_ОПРЕДЕЛЕН) ИЛИ (столбец6
МЕЖДУ 100 И 200)
ИЛИ ((столбец3- столбец2) РЕЗУЛЬТАТ_НЕ_ОПРЕДЕЛЕН)
ИЛИ (столбец7 СОДЕРЖИТ_ЗНАЧЕНИЕ ЛОЖЬ)
УПОРЯДОЧИТЬ ПО столбец1, столбец2 ПО_ВОЗРАСТ-
ТАНИЮ
```

В шаблоне LIKE подчеркивание означает любой одиночный символ, а «%» – произвольное число любых символов (в MS Access вместо «\_» используют «?», а вместо «%» используют «\*»). Так, под шаблон «М\_нск%» подходят слова

«Минск», «Менский» и т. п. Результат вычислений может быть неопределен (unknown), если хотя бы один столбец содержит неопределенное значение (null). Сортировка по двум столбцам выполняется следующим образом: сначала по одному столбцу, а в рамках повторяющихся значений первого столбца – по второму столбцу. Так можно отсортировать сотрудников по должностям, а в рамках должности бухгалтера – по стажу. Сортировка по возрастанию может не указываться (сортируется по умолчанию), либо задаваться ключевым словом ASC. При сортировке по убыванию вместо ASC указывается DESC.

**Многотабличные запросы.** Информация выбирается из двух или более таблиц, имеющих общее поле, т. е. в каждой из таблиц имеется столбец, содержащий такие же значения (не обязательно все), как и в другой таблице. На рис. 2.2 показан пример многотабличного запроса, а ниже указана соответствующая инструкция.

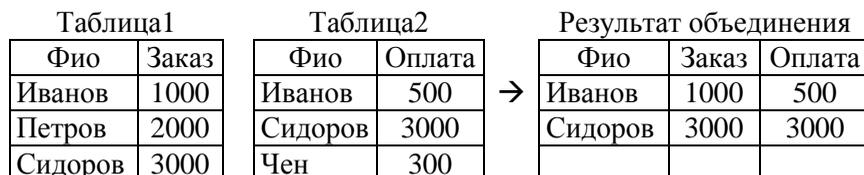


Рис. 2.2. Внутреннее объединение таблиц

```

SELECT Таблица1.Фино, Заказ, Оплата
FROM Таблица1 INNER JOIN Таблица2
ON Таблица1.Фино= Таблица2.Фино
Русский перевод ключевых слов инструкции:
ОТОБРАТЬ Таблица1.Фино, Заказ, Оплата
ИЗ Таблица1 ПРОВЕДЯ_ВНУТРЕННЕЕ
ОБЪЕДИНЕНИЕ_С Таблица2
ПО Таблица1.Фино= Таблица2.Фино

```

При внутреннем объединении остаются только строки со значениями, совпадающими в обеих таблицах. При этом часть данных «теряется» (например, строки «Чен» и «Петров»). Если по

логике запроса этого необходимо избежать, то можно воспользоваться «полным», «левым» или «правым» внешним объединением. Инструкция INNER JOIN заменяется соответственно на FULL OUTER JOIN (полное), LEFT OUTER JOIN (левое) или RIGHT OUTER JOIN (правое объединение). В случае полного внешнего объединения (рис. 2.3) в результирующей таблице сохраняются все строки, а отсутствующие данные заполняются значениями NULL (неопределенное значение). В случае левого внешнего объединения (рис. 2.4) сохраняются значения только первой таблицы в запросе (находится слева от слова JOIN), а несовпадающие строки второй таблицы удаляются. В случае правого внешнего объединения сохраняются значения только второй таблицы в запросе (находится справа от слова JOIN).

Таблица1		Таблица2		Результат объединения		
Фио	Заказ	Фио	Оплата	Фио	Заказ	Оплата
Иванов	1000	Иванов	500	Иванов	1000	500
Петров	2000	Сидоров	3000	Сидоров	3000	3000
Сидоров	3000	Чен	300	Петров	2000	NULL
SELECT Таблица1.Фио, Заказ, Оплата FROM Таблица1 FULL OUTER JOIN Таблица2 ON Таблица1.Фио= Таблица2. Фио				Чен	NULL	300

Рис. 2.3. Полное внешнее объединение таблиц

Таблица1		Таблица2		Результат объединения		
Фио	Заказ	Фио	Оплата	Фио	Заказ	Оплата
Иванов	1000	Иванов	500	Иванов	1000	500
Петров	2000	Сидоров	3000	Сидоров	3000	3000
Сидоров	3000	Чен	300	Петров	2000	NULL
SELECT Таблица1.Фио, Заказ, Оплата FROM Таблица1 LEFT OUTER JOIN Таблица2 ON Таблица1.Фио = Таблица2. Фио						

Рис. 2.4. Левое внешнее объединение таблиц

**Объединение результатов запросов.** Несколько однотипных запросов могут быть объединены в общий список при помощи инструкции UNION. Объединяемые столбцы должны иметь одинаковый тип данных (но не обязательно логически однородное содержимое). Инструкция DISTINCT для удаления повторяющихся строк должна присутствовать только в последнем из объединяемых запросов.

```
SELECT Фιο, Заказ, Оплата FROM Таблица1 WHERE Заказ>0  
UNION
```

```
SELECT DISTINCT Фιο, Оплата FROM Таблица2 WHERE  
Оплата>300
```

Русский перевод ключевых слов инструкции:

```
ОТОБРАТЬ Фιο, Заказ ИЗ Таблица1 ГДЕ Заказ>0  
ОБЪЕДИНИВ РЕЗУЛЬТАТЫ С  
ОТОБРАТЬ ОТЛИЧАЮЩИЕСЯ ЗНАЧЕНИЯ Фιο, Оплата  
ИЗ Таблица2  
ГДЕ Оплата>300
```

**Итоговые запросы (запрос с группировкой).** Позволяют сгруппировать данные по какому-либо столбцу и рассчитать итоговые значения по каждой группе. Ниже приведен пример инструкции, в котором данные сначала группируются по столбцу отдела, а в рамках отдела – по должностям. Затем по каждой группе считается число сотрудников (фамилий), средняя зарплата и т. п. Условие WHERE накладывается на данные ДО группировки, а условие HAVING – ПОСЛЕ группировки. Если не выполнить группировку (не указать GROUP BY), то среднее значение, минимум, максимум и прочие функции будут рассчитаны в целом по столбцу.

```
SELECT Отдел, Должность, COUNT(Фιο), AVG(Зарплата),  
MAX(Зарплата),
```

```
MIN(Зарплата), SUM(Продажи), COUNT (DISTINCT Фи-  
лиалы)
```

```
FROM Таблица1 WHERE столбецГород='Минск'  
GROUP BY Отдел, Должность  
HAVING COUNT (Фео) > 2  
ORDER BY Отдел DESC
```

Русский перевод ключевых слов инструкции:

```
ОТОБРАТЬ Отдел, Должность, ЧИСЛО_СТРОК(Фео),  
СРЕДНЯЯ (Зарплата),  
МАКСИМУМ (Зарплата), МИНИМУМ(Зарплата),  
СУММА (Продажи), ЧИСЛО_СТРОК  
(ОТЛИЧАЮЩИЕСЯ_ЗНАЧЕНИЯ Филиалы)  
ИЗ Таблица1 ГДЕ столбецГород='Минск'  
СГРУППИРОВАТЬ ПО Отдел, Должность  
ИМЕЮЩИЕ_ПОСЛЕ_ГРУППИРОВКИ  
ЧИСЛО_СТРОК(Фео) > 2  
УПОРЯДОЧИТЬ ПО Отдел ПО_УБЫВАНИЮ
```

**Подчиненные запросы (вложенные запросы).** Позволяют создать несколько уровней вложенности запросов. В обычных запросах условия отбора строк WHERE и HAVING выглядят примерно так:

WHERE столбец = значение.

В подчиненных запросах эта схема модифицируется следующим образом:

WHERE столбец = (результаты еще одного вложенного запроса).

Таким образом, в условиях вместо констант используются результаты других запросов, причем число уровней вложенности запросов может быть достаточно велико. По этой причине в таких запросах во избежание путаницы с одинаковыми названиями столбцов целесообразно использовать полные имена столбцов в формате «ИмяТаблицы.ИмяСтолбца». Ниже приведен пример инструкции, отбирающей из списка сотрудников из отдела маркетинга фамилию работника, зарплата которого больше, чем у любого из работников филиала предприятия.

```
SELECT Фио FROM таблицаСотрудники
WHERE Зарплата > ALL
      (SELECT таблицаФилиал.Зарплата
      FROM таблицаФилиал WHERE отдел = 'маркетинг' )
```

Русский перевод ключевых слов инструкции:

```
ОТОБРАТЬ Фио ИЗ таблицаСотрудники
ГДЕ Зарплата > ВСЕХ (ОТОБРАТЬ таблицаФилиал.Зарплата
ИЗ таблицаФилиал ГДЕ отдел = 'маркетинг' )
```

Вместо ключевого слова ALL (все) в обработке результатов подчиненных запросов могут также использоваться ключевые слова ANY (кто-либо), IN (среди значений), NOT EXISTS (не существует), MAX (максимум), MIN (минимум) и другие стандартные функции.

### **Задания для выполнения**

**Задание 1.** Создать запросы в конструкторе MS Access. Приведенные ниже задания выполняются в следующем порядке: запрос создается при помощи конструктора запросов MS Access на основании таблиц, созданных в лабораторной работе № 1. Затем отдельно (в блокноте) студентом записывается SQL-инструкция, соответствующая данному запросу. После чего конструктор MS Access переводится в режим просмотра SQL-инструкции (команда меню «Вид/Режим SQL») и полученная SQL-инструкция сопоставляется с инструкцией, написанной студентом.

Задания для выполнения:

- создать запрос «Наличие изделий». В запрос включить столбцы «Изделие» и «Наличие» из таблицы «Склад»;
- создать запрос «Заказчики F14». В запрос включить заказчиков «F14», отсортировав их по убыванию;

- создать запрос «Мелкие заказчики», содержащий столбцы «Заказчик» и «Количество» из таблицы «Заказ» и включающий заказчиков, у которых количество заказанных изделий находится в пределах от 1 до 10 или заказанное изделие начинается на «Палочки». Заказчики в таблице не должны повторяться;
- создать запрос «Форма заказа», содержащий столбцы «№ заказа», «Заказчик», «Изделие», «Количество» из таблицы «Заказ», столбец «Цена» из таблицы «Склад» и вычисляемый столбец «Сумма заказа» (=Цена\*Количество). Отсортировать результаты запроса по полю «№ заказа» по возрастанию;
- создать запрос «Закупить», включающий изделия, для которых в соответствии с имеющимися заказами необходимо произвести дополнительные закупки. В запрос включить столбец «Изделие» и «Закупить» (в столбце рассчитывается необходимый объем закупок).

**Задание 2.** Создать запросы при помощи инструкций SQL. При выполнении данного задания инструкции необходимо непосредственно вводить в режиме SQL конструктора запросов (меню «Вид/Режим SQL»):

- написать инструкцию SQL и создать запрос «Заказ-оплата», включающий в себя столбцы «№ заказа», «Заказчик», «Сумма заказа» из запроса «Форма заказа», и столбец «Оплата» из таблицы «Оплата». В результирующем запросе не должны быть потеряны заказчики, от которых не поступила оплата (использовать левое внешнее объединение);
- написать инструкцию SQL и создать запрос «Контакты», представляющий общий список заказчиков (таблица «Заказ») и поставщиков (таблица «Склад»), имевших дело с данным предприятием. Заказчики и поставщики должны быть перечислены в один столбец и не должны повторяться;
- написать инструкцию SQL и создать запрос «Максимальная партия», подсчитывающий максимальную партию (столбец

«Количество» таблицы «Заказы») по заказам, у которых название изделия не начинается со слова «Палочки»;

- написать инструкцию SQL и создать запрос «Средняя партия по крупным заказам». В запрос включить только изделия, для которых средняя партия превышает 1 и название изделия не начинается со слова «Палочки»;

- написать инструкцию SQL и создать запрос «Изделия без спроса», перечисляющий все изделия из таблицы «Склад», на которые отсутствуют заказы в таблице «Заказ». Изделия не должны повторяться;

- написать инструкцию SQL и создать запрос «Партия больше средней», включающий столбцы «Изделие» и «Количество». В результаты запроса должны попасть только те изделия, по которым партия (столбец «Количество») превышает средний размер партии по всем изделиям (без учета изделий, начинающихся на «Палочки», при определении среднего значения), и название самого изделия не начинается со слова «Палочки».

## Лабораторная работа № 3

### ИСПОЛЬЗОВАНИЕ SQL В MS ACCESS. СОЗДАНИЕ ТАБЛИЦ И ИЗМЕНЕНИЕ ДАННЫХ ПРИ ПОМОЩИ SQL

#### Цели работы:

- получение навыков использования языка SQL в СУБД MS Access;
- изучение возможностей использования языка SQL для создания таблиц и изменения данных.

#### Теоретический материал

Язык SQL позволяет не только строить запросы к базе данных, но и изменять хранящуюся в ней информацию и даже структуру базы данных. Рассмотрим примеры соответствующих инструкций SQL.

#### *Однострочная инструкция добавления данных*

```
INSERT INTO Таблица1 (столбец1, столбец3) VALUES ('Иванов', 300000)
```

Русский перевод ключевых слов инструкции:

```
ВСТАВИТЬ В Таблица1 (столбец1, столбец3) ЗНАЧЕНИЯ ('Иванов', 300000)
```

Если при вставке строки данные указаны для **всех** столбцов по порядку, то инструкцию можно записать в сокращенном варианте:

```
INSERT INTO Таблица1 ('Иванов', NULL, 300000)
```

*Многострочная инструкция добавления данных* вставляет в таблицу строки, полученные в результате запроса.

```
INSERT INTO Таблица1 (столбец1, столбец3)  
SELECT Фио, Стипендия FROM Таблица2 WHERE курс>2
```

Русский перевод ключевых слов инструкции:

ВСТАВИТЬ В Таблица1 (столбец1, столбец3)  
ОТОБРАТЬ Фио, Стипендия ИЗ Таблица2 ГДЕ курс>2

### ***Удаление строк***

DELETE FROM Таблица1 WHERE Фио='Иванов'

Русский перевод ключевых слов инструкции:

УДАЛИТЬ\_СТРОКИ ИЗ Таблица1 ГДЕ Фио='Иванов'

Если не указать критерий отбора WHERE, то из таблицы будут удалены все строки. Инструкция удаляет строки, но не удаляет саму таблицу.

### ***Обновление данных***

UPDATE Таблица1 SET столбец1='Петров',  
столбец3=столбец3+100000  
WHERE столбец3<200000

Русский перевод ключевых слов инструкции:

ОБНОВИТЬ Таблица1 УСТАНОВИТЬ столбец1='Петров',  
столбец3=столбец3+100000  
ГДЕ столбец3<200000

***Создание таблиц*** осуществляется инструкцией CREATE TABLE. Необходимо помнить, что связь создается не между любыми столбцами, а только между первичным ключом таблицы и внешним ключом. Ниже приведен пример инструкции, создающей таблицу1 со связями с другими таблицами, приведенными на рис. 3.1.

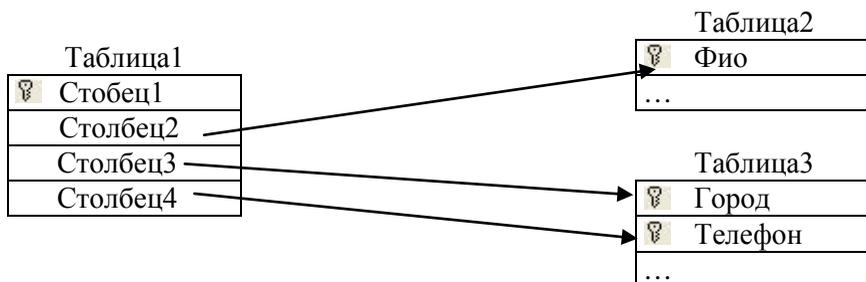


Рис. 3.1. Схема данных создаваемой таблицы

```

CREATE TABLE Таблица1 (
    Столбец1 INTEGER, Столбец2 VARCHAR(30),
    Столбец3 VARCHAR(30), Столбец4 VARCHAR(20),
    CONSTRAINT условие1
    PRIMARY KEY (Столбец1),
    CONSTRAINT условие2
    FOREIGN KEY (Столбец2) REFERENCES Таблица2
    ON DELETE CASCADE
    ON UPDATE RESTRICT,
    CONSTRAINT условие3
    FOREIGN KEY (Столбец3, Столбец4) REFERENCES
Таблица3
    ON DELETE SET DEFAULT
    ON UPDATE SET NULL,
    CONSTRAINT условие4
    CHECK (Столбец3 IN 'Минск', 'Пинск'))
  
```

Русский перевод ключевых слов инструкции:

```

СОЗДАТЬ ТАБЛИЦУ Таблица1 (
    Столбец1 ЦЕЛОЕ, Столбец2 СТРОКА_СИМВОЛОВ(30)
    Столбец3 СТРОКА_СИМВОЛОВ(30), Столбец4 СТРО-
КА_СИМВОЛОВ(20),
    ОГРАНИЧЕНИЕ условие1
    ПЕРВИЧНЫЙ КЛЮЧ (Столбец1),
  
```

ОГРАНИЧЕНИЕ условие2  
ВНЕШНИЙ КЛЮЧ (Столбец2) ССЫЛАЕТСЯ\_НА  
Таблица2  
ПРИ УДАЛЕНИИ КАСКАДНОЕ\_УДАЛЕНИЕ  
ПРИ ОБНОВЛЕНИИ ОБЕСПЕЧИТЬ\_ЦЕЛОСТНОСТЬ,  
ОГРАНИЧЕНИЕ условие3  
ВНЕШНИЙ КЛЮЧ (Столбец3, Столбец4)  
ССЫЛАЕТСЯ\_НА Таблица3  
ПРИ УДАЛЕНИИ УСТАНОВИТЬ ЗНАЧЕНИЕ\_ПО\_-  
УМОЛЧАНИЮ  
ПРИ ОБНОВЛЕНИИ УСТАНОВИТЬ НЕОПРЕДЕЛЕН-  
НОЕ\_ЗНАЧЕНИЕ,  
ОГРАНИЧЕНИЕ условие4  
ПРОВЕРЯТЬ\_ПРИ\_ВВОДЕ\_ДАННЫХ (Столбец3  
СРЕДИ\_ЗНАЧЕНИЙ 'Минск', 'Пинск') )

### ***Удаление таблиц***

DROP TABLE Таблица1 CASCADE *или* RESTRICT

Русский перевод ключевых слов инструкции:

УДАЛИТЬ ТАБЛИЦУ Таблица1  
С\_КАСКАДНЫМ\_УДАЛЕНИЕМ\_СВЯЗАННЫХ\_ЗАПИСЕ  
Й\_В\_ДРУГИХ\_ТАБЛИЦАХ

*или*

С\_ЗАПРЕТОМ\_УДАЛЕНИЯ\_ПРИ\_НАЛИЧАЕ\_СВЯЗАНН  
ЫХ\_ЗАПИСЕЙ

### ***Добавление столбцов таблиц***

ALTER TABLE Таблица1 ADD COLUMN Столбец5 INTEGER

Русский перевод ключевых слов инструкции:

ИЗМЕНИТЬ ТАБЛИЦУ Таблица1 ДОБАВИТЬ СТОЛБЕЦ  
Столбец5 ЦЕЛОЕ\_ЧИСЛО

## *Удаление столбцов таблиц*

ALTER TABLE Таблица1 DROP Столбец5 CASCADE *или*  
RESTRICT

Русский перевод ключевых слов инструкции:

ИЗМЕНИТЬ ТАБЛИЦУ Таблица1 УДАЛИТЬ Столбец5  
С\_КАСКАДНЫМ\_УДАЛЕНИЕМ\_СВЯЗАННЫХ\_ЗАПИС  
ЕЙ\_В\_ДРУГИХ\_ТАБЛИЦАХ

*или*

С\_ЗАПРЕТОМ\_УДАЛЕНИЯ\_ПРИ\_НАЛИЧИЕ\_СВЯЗАНН  
ЫХ\_ЗАПИСЕЙ

## **Задания для выполнения**

Создать запросы при помощи инструкций SQL на основании таблиц и запросов, созданных в лабораторных работах № 1 и № 2. При выполнении данного задания инструкции необходимо непосредственно вводить в режиме SQL конструктора запросов (меню «Вид/Режим SQL»):

- написать инструкцию SQL и создать запрос, вносящий в таблицу «Склад» информацию о закупке десяти дополнительных палочек для воды;
- написать инструкцию SQL и создать запрос, добавляющий в таблицу «Склад» информацию о поступлении нового изделия «M21» в количестве 25 штук по цене 500 (USD), поставщик не известен;
- написать инструкцию SQL и создать запрос, удаляющий из таблицы «Склад» запись об изделии «M21», сделанную в предыдущем задании;
- написать инструкцию SQL и создать запрос, создающий таблицу «Резерв», состоящую из столбцов «Изделие» (строка символов переменной длины до 50 символов, является первич-

ным ключом таблицы), «Наличие» (целое число), «Примечание» (строка символов переменной длины до 100 символов);

- написать инструкцию SQL и создать запрос, вставляющий в созданную таблицу «Резерв» данные из таблицы «Склад» по всем изделиям, начинающимся с «F»;

- написать инструкцию SQL и создать запрос, создающий таблицы «Склад1» и «Заказ1», по структуре аналогичные таблицам «Склад» и «Заказ». Столбец «Изделие» из таблицы «Заказ1» является внешним ключом для таблицы «Склад1». Если на изделие существуют заказы в таблице «Заказ1», то попытки удалить его из таблицы «Склад1» блокируются.

## **Лабораторная работа № 4**

### **ОСНОВНЫЕ ПРИЕМЫ РАБОТЫ С MS ACCESS. СОЗДАНИЕ ФОРМ, ОТЧЕТОВ И МАКРОСОВ**

#### **Цели работы:**

- получение навыков создания форм в СУБД MS Access;
- получение навыков создания отчетов в СУБД MS Access;
- получение навыков использования макросов в СУБД MS Access

#### **Теоретический материал**

Наиболее гибким инструментом является создание форм в режиме конструктора, однако создание форм может осуществляться и при помощи мастера. Мастер форм позволяет автоматически создать форму по указанной таблице, на основании которой строится форма. Возможны следующие варианты форм: в один столбец (одновременно на экране отображается только одна запись, слева – название поля, справа – данные), ленточная (отображается несколько записей в виде таблицы, сверху названия столбцов), табличная (аналогична ли-

сту Excel), выровненная. При помощи команды меню «Вставка/Форма» или кнопки «Создать» в окне базы данных (рис. 4.1) возможно также быстрое создание формы по всем полям таблицы с использованием автоформ.

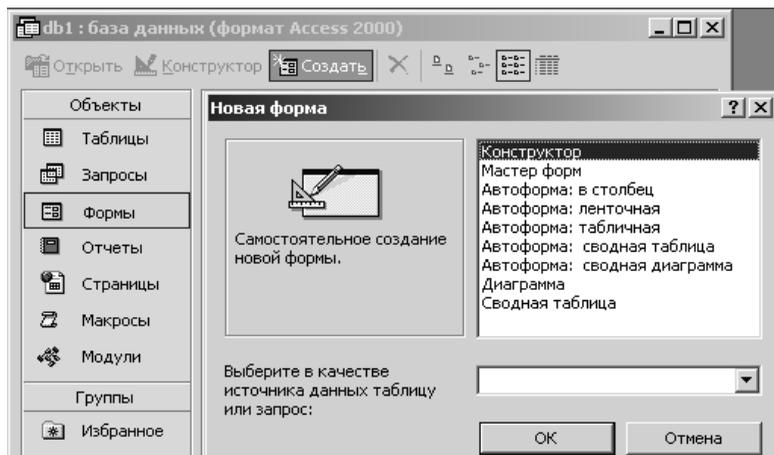


Рис. 4.1. Режимы создания новой формы

При создании формы в режиме конструктора в форме выделяется несколько областей (рис. 4.2 и рис. 4.3):

- заголовок/примечание (меню «Вид/Заголовок», «Вид/Примечание формы»);
- колонтитулы (меню «Вид/Колонтитулы»);
- область данных.

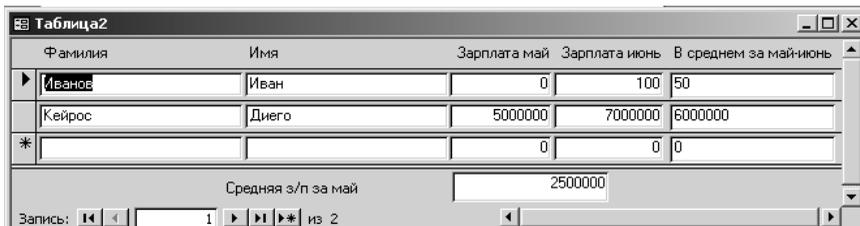


Рис. 4.2. Ленточная форма в режиме просмотра

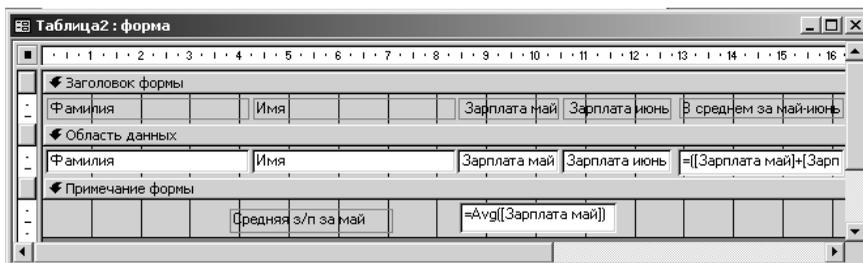


Рис. 4.3. Ленточная форма в режиме конструктора

В случае создания ленточной формы поля, размещенные в области данных, выводятся на экран столько раз, сколько присутствует записей в форме. Поля, размещенные в заголовке и примечании, выводятся на экран однократно. Таким образом, если необходимо создать поле с подсчетом среднего значения по каждой строке формы, его необходимо размещать в области данных, а если необходимо создать поле с подсчетом среднего значения по всему столбцу – в области примечания. В заголовке чаще всего создаются надписи, формирующие шапку с названиями столбцов.

Поля и надписи в форме создаются с помощью панели элементов. Кнопка «Надпись» **Aa** вставляет обычный текст, кнопка «Поле» **ab|** – поле данных или вычислений, кнопка «Кнопка»  создает кнопки формы. Если при создании новой кнопки включен режим мастера , то пользователю предлагается выбрать действия и внешний вид кнопки из стандартного списка. Если режим мастера не включен (отжат переключатель ) , то создается пустая кнопка, не производящая никаких действий. Впоследствии для кнопки правым щелчком мыши можно вызвать контекстное меню «Свойства/вкладка События» и в полях «Нажатие кнопки», «Двойное нажатие кнопки» и т. п. указать макрос или программу VBA, которые будут выполняться при нажатии на кнопку. Аналогично обработку событий можно описать и для любых

других элементов формы. Так, через контекстное меню «Свойства/ вкладка События/двойное нажатие кнопки» можно задать срабатывание программы при двойном щелчке по полю данных или изображению (рисунки вставляются с использованием кнопки  на панели элементов).

Задание или смена источника данных формы осуществляется также в меню «Свойства» (выделить всю форму целиком и нажать F4 либо вызвать свойства любого элемента формы и в верхней части диалога свойств в выпадающем списке выбрать «Форма») на вкладке «Данные/Источник записей». Для облегчения создания полей может быть вызван построитель выражений: выделить поле, контекстное меню «Свойства/Данные/Данные», нажать кнопку  рядом с полем «Данные». Построитель позволяет двойным щелчком заносить в поле название других полей формы, названия таблиц и т. п. Также построитель позволяет создавать вычисляемые поля с использованием встроенных функций Access (рис. 4.4). Пример создания вычисляемого поля приведен в табл. 4.1.

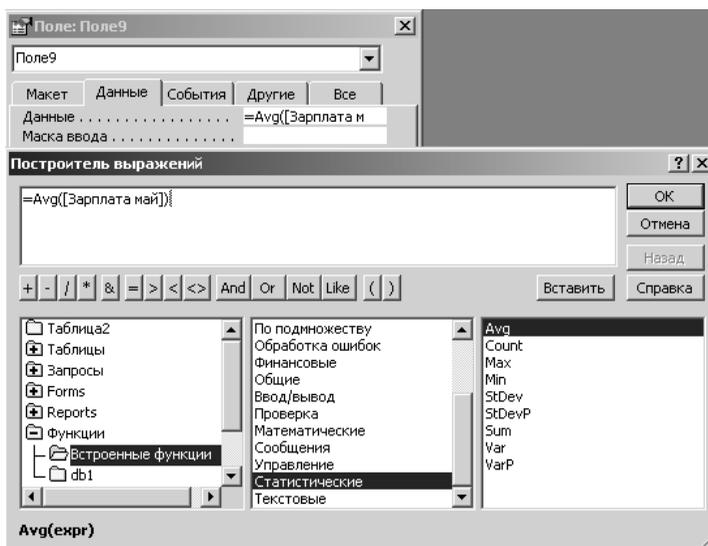


Рис. 4.4. Построитель выражений

## Создание вычисляемых полей

Вычисляемое поле	Описание
=Sum([Количество]*[Цена])	Сумма произведений столбцов «Количество» и «Цена» по всем строкам формы
=Avg([СтоимостьДоставки])	Среднее значение столбца «СтоимостьДоставки»
=Count([КодЗаказа])	Количество записей в столбце «КодЗаказа»
=DSum("[Столбец1]", "[Таблица1]", "[Столбец3] = 'Минск' ")	Использует функцию по подмножеству. Суммируются только те значения столбца «Столбец1» из таблицы «Таблица1», для которых в той же строке в столбце «Столбец3» содержится значение «Минск»

Внешний вид интерфейса формы (например, отображение кнопок оконного меню) настраивается на вкладке «Макет» свойств формы. Пользовательское контекстное меню в форме создается следующим образом:

- выбрать меню «Сервис/Настройка/вкладка Панели инструментов/кнопка Создать/указать имя панели меню/кнопка ОК»;
- с вкладки «Команды» перетащить на созданную панель необходимые кнопки;
- перейти на вкладку «Панели инструментов», выбрать созданную панель, нажать кнопку «Свойства», изменить тип на «Контекстное меню», нажать «ОК/Закрыть»;
- открыть форму в режиме конструктора, выделить всю форму, нажать F4 (свойства формы), на вкладке «Другие» в пункте «Контекстное меню» указать использование в форме созданного контекстного меню.

Для автоматического запуска формы при открытии базы данных используется меню «Сервис/Параметры запуска».

### Создание отчетов

Единственное отличие отчета от формы заключается в том, что отчет предполагает выведение информации на бумагу. В остальном создание отчетов аналогично созданию форм. В режиме конструктора выделяются те же области: заголовок отчета, верхний колонтитул, область данных, нижний колонтитул, примечание отчета (рис. 4.5).

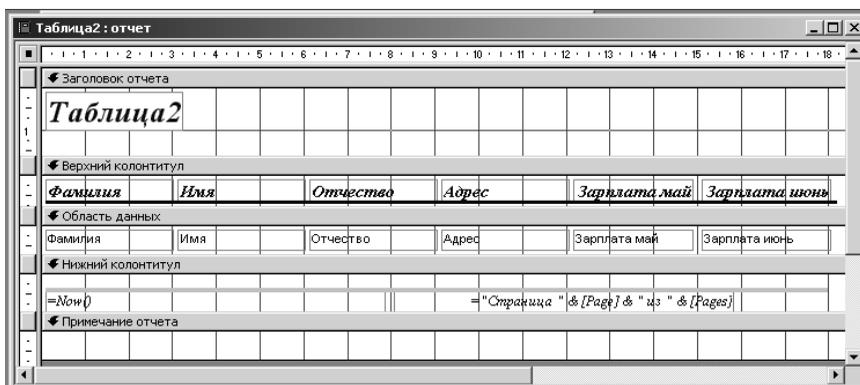


Рис. 4.5. Отчет в режиме конструктора

Колонтитулы отображаются на каждой странице отчета. Примечание и заголовок – в начале и конце отчета. Аналогичным образом создаются вычисляемые поля и надписи. Используя меню «Вид/Сортировка и группировка», можно создавать отчеты с группировкой, например, объем продаж товара, сгруппированный по регионам продаж и т. п. Также возможно создание отчетов и диаграмм с использованием мастера отчетов.

## Создание макросов

Макросы позволяют выполнить линейную последовательность стандартных команд Access. Задается сама команда и её параметры. На рис. 4.6 приведен пример выполнения инструкции SQL при помощи макроса.

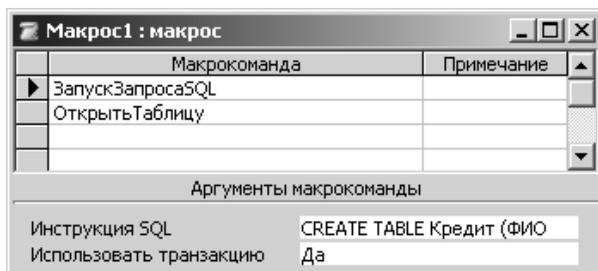


Рис. 4.6. Выполнение инструкции SQL при помощи макроса

### Задания для выполнения

Задания выполняются на основании таблиц и запросов, созданных в лабораторных работах № 1–3.

**Задание 1.** Создать на основании таблицы «Склад» ленточную форму и отчет. Предусмотреть в примечании формы/отчета подсчет средней цены по изделиям, находящимся на складе.

**Задание 2.** Создать ленточную форму «Форма заказа» на основании созданного ранее запроса «Форма заказа». В примечании формы предусмотреть поле «Суммарный объем продаж F14», в котором будет рассчитана общая сумма заказа по всем закупкам «F14» (использовать функции по подмножеству).

**Задание 3.** В форме «Форма заказа» создать: кнопку закрытия формы; кнопку просмотра отчета, аналогичного форме (предварительно необходимо создать соответствующий от-

чет); кнопку печати отчета, аналогичного форме; кнопку закрытия Access. При помощи меню «Вид/Программа» ознакомиться с текстом VBA-программ, соответствующих кнопкам.

**Задание 4.** Создать собственное контекстное меню формы «Форма заказа». Отключить отображение стандартных кнопок оконного меню (крестик на закрытие и т. д) в форме.

**Задание 5.** Установить параметры запуска базы данных так, чтобы при ее загрузке стандартное окно базы данных не загружалось, а вместо этого отображалась любая из созданных ранее форм. Проверить действие команды, перезагрузив базу данных. Предотвратить использование параметров запуска, удерживая клавишу <Shift> при загрузке базы данных, и изменить эти параметры.

**Задание 6.** Создать отчет по запросу «Заказ-Оплата», сгруппировав данные по заказчикам (меню «Вид/Сортировка и группировка») и поместив в заголовке группы надпись «Заказчик», а в примечании группы – среднюю сумму заказа по каждому заказчику.

**Задание 7.** Создать макрос, состоящий из следующих действий:

1) выполнение инструкции SQL вида: CREATE TABLE Кредит (ФИО VARCHAR(30) NOT NULL, Лимит NUMERIC, constraint ogr1 primary key (ФИО), constraint ogr2 UNIQUE (ФИО));

2) вывод созданной таблицы «Кредит» на экран.

Перейти на вкладку «Макросы», выделить созданный макрос и, воспользовавшись меню «Сервис/Макрос/Преобразовать макросы», преобразовать созданный макрос в модуль программы на VBA. Ознакомиться с содержимым полученного модуля на вкладке «Модули».

Воспользовавшись командой меню «Сервис/Настройка/вкладка Команды/ Категория: Все макросы», создать в меню и на панели управления кнопку, выполняющую данный макрос. В любой форме создать кнопку, выполняющую созданный макрос. Проверить работоспособность кнопки и проверить, что отображается в свойствах кнопки в разделе «События/ Нажатие кнопки». Воспользовавшись меню «Вид/Программа», перейти к просмотру VBA-программ формы и убедиться, что там отсутствует программа, соответствующая созданному макросу. В режиме конструктора формы выполнить команду «Сервис/Макрос/Преобразовать макросы формы», затем заново перейти к просмотру VBA-программ формы и найти там преобразованную VBA-программу, соответствующую макросу.

## Лабораторная работа № 5

### СИСТЕМА БЕЗОПАСНОСТИ MS ACCESS

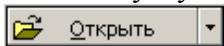
#### Цели работы:

- изучение системы безопасности и разграничения доступа в MS Access;
- получение навыков создания защищенной базы данных.

#### Теоретический материал

Access обеспечивает два способа защиты базы данных: установка пароля и защита на уровне пользователей. Кроме того, можно предотвратить изменения структуры форм, отчетов и модулей, сохранив базу данных как файл MDE (меню «Сервис/Служебные программы/Создать MDE-файл»).

Для установки пароля откройте файл базы данных монопольно (в однопользовательском режиме), выбрав в правом нижнем углу диалогового окна вместо кнопки «Открыть»



вариант «Монопольно». Выполните команду

меню «Сервис/Защита/Задать пароль базы данных». Введите и подтвердите пароль (с учетом регистра).

Защита на уровне пользователей позволяет установить разные права доступа на различные объекты базы данных для различных пользователей. При запуске Access пользователь указывает свое имя и пароль. Если данный пользователь есть в списке и пароль введен правильно, то пользователь подключается к Access под своим именем и как участник своей группы. Группы пользователей: Admins (администраторы), Users (пользователи) или любые другие определяемые вручную группы. Каждая группа имеет свой определяемый набор прав. Например, члены группы Users могут иметь разрешения на просмотр, ввод или изменение данных в таблицы, но им не будет разрешено изменять структуру таблиц. Группа Users может быть допущена только к просмотру данных в таблице «Заказы» и не иметь доступа к таблице «Платежная ведомость». Члены группы Admins имеют все разрешения на доступ ко всем объектам базы данных. Кроме того, каждый пользователь может иметь свой индивидуальный набор прав. Права пользователя суммируются с правами группы, в которую он входит.

Владелец базы данных или отдельного объекта базы данных – это тот пользователь, который был подключен к Access при создании базы данных или отдельного объекта. Владелец базы данных всегда может открыть базу данных и получить все разрешения на объекты для которых он является владельцем.

**Создание нового пользователя:** «Сервис/Защита/Пользователи и группы/вкладка Пользователи/кнопка Создать/ввести имя нового пользователя и его код». Код пользователя – это не пароль. Фактически, к базе данных имеет доступ не имя пользователя, а комбинация «имя+уникальный код». Для повторного создания идентичного пользователя необходимо указать не только правильное имя, но и правильный код. Используя кнопки «Добавить», «Удалить», можно переместить в окно «Участие в группе» те группы, в которых будет участвовать

пользователь. Группа Admins дает пользователю права администратора, Users – права рядового пользователя. Кнопка «Удалить» удаляет пользователя.

**Создание новой группы:** «Сервис/Защита/Пользователи и группы/вкладка «Группы»/кнопка «Создать»/ввести название группы и ее уникальный код».

**Изменение пароля пользователя:** «Сервис/Защита/Пользователи и группы/вкладка «Изменение пароля»/указать старый пароль, новый пароль и подтверждение пароля». Можно изменить только пароль того пользователя, от имени которого было произведено подключение к Access. Для того, чтобы изменить пароль любого пользователя, необходимо выйти из Access и снова подключиться к нему от имени этого пользователя. Пользователь, обладающий правами администратора, может отменить пароль любого пользователя, отображаемого на вкладке в окне «Пользователь имя:», нажав кнопку «Снять пароль».

**Подключение к Access от имени определенного пользователя:** можно подключиться из командной строки (Access.exe/ /User <имя >/Pwd <пароль>). Если в командной строке пароль не указан, то Access выведет диалоговое окно, где попросит его указать. Можно также воспользоваться следующей техникой: Access по умолчанию пытается подключить всех пользователей как Admin. Достаточно установить пароль для пользователя Admin и при стандартном запуске Access всегда будет выводить диалоговое окно с запросом на ввод имени пользователя и его пароля. Чтобы отменить вывод этого запроса, администратору достаточно снять пароль пользователю Admin кнопкой «Снять пароль».

**Установка защиты на уровне пользователей:** создать список групп и имен пользователей. Запустить Access без открытия базы данных как пользователь Admin или любой другой с правами администратора. Открыть защищаемую базу данных. В меню «Сервис/Разрешения/вкладка Разрешения/поле Тип объекта/выбрать База данных/выбрать имя пользователя или

название группы для которой устанавливаются разрешения», флажками отметить, что разрешено данному пользователю или группе (разрешения отдельных пользователей и разрешения групп, в которых они участвуют, суммируются). На вкладке «Разрешения» в поле «Тип объекта» выбрать остальные объекты базы данных «Таблицы», «Запросы», «Формы», «Отчеты», «Макросы», «Модули» и установить разрешения для них. При этом можно определять набор разрешений как для уже существующих объектов, так и для вновь создаваемых.

**Удаление защиты на уровне пользователей:** подключиться к Access как администратор. Загрузить защищенную базу данных. Предоставить группе Users разрешения на все объекты базы данных. Возвратить права владельца базы данных и ее объектов стандартной учетной записи пользователя Admin. Выйти из Access и снова подключиться к системе под именем Admin, создать пустую базу данных, импортировать все объекты из защищенной базы данных в новую. Снять пароль пользователя Admin.

**Смена владельца отдельных объектов базы данных:** меню «Сервис/Защита/Пользователи и группы/вкладка Смена владельца». Выбрать тип объектов в списке (таблица, запрос, форма, отчет и т. д.). Выделить один или несколько объектов, для которых требуется произвести смену владельца. В списке «Новый владелец» выбрать имя пользователя или группы, которым передаются права владельца объекта. Нажать кнопку «Сменить владельца».

**Смена владельца всей базы данных:** подключиться к Access от имени пользователя, который станет новым владельцем базы данных. Создать новую пустую базу данных. Импортировать в нее все объекты из исходной базы данных. Для импорта пользователь должен иметь разрешение «Открытие/запуск» для базы данных и «Чтение макета/Чтение данных» на ее объекты.

**Создание нового файла рабочей группы:** в файле рабочей группы содержится перечень имен пользователей и групп. Ме-

ню «Сервис/Защита/Администратор рабочих групп». В диалоговом окне нажать кнопку «Создать» и ввести свое имя и название организации. В поле «Код рабочей группы» ввести до 20 цифр/букв и нажать кнопку «ОК». Ввести имя нового файла рабочей группы, указать, куда сохранить и нажать кнопку «ОК».

**Присоединение к существующему файлу рабочей группы:** меню «Сервис/Защита/Администратор рабочих групп». В диалоговом окне нажать кнопку «Связать». Ввести путь и имя файла рабочей группы или нажать кнопку «Обзор». При следующем запуске Microsoft Access будет использован новый файл рабочей группы. Можно также указать файл рабочей группы при запуске Access с параметром командной строки Access.exe /wrkgrp <путь и имя файла рабочей группы>.

**Устранение слабых мест защиты на уровне пользователей:** проблему представляют стандартный пользователь Admin и стандартный код групп Admins/Users при использовании стандартного файла рабочей группы.

Учетные записи Admin одинаковы для всех экземпляров Microsoft Access. По этой причине Admin не должен быть владельцем защищенной базы данных или ее объектов. Пользователь Admin не должен иметь права администратора или входить в группу администраторов. В противном случае можно легко создать файл рабочей группы заново (создав тем самым стандартного пользователя Admin без пароля) и получить полный доступ к защищенной базе данных.

То же относится и к группе Admins, для обеспечения уникальности которой требуется создать файл рабочей группы с уникальным кодом (при установке Access «по умолчанию» это не сделано).

Последовательность действий для создания защищенной базы такова: создать файл рабочей группы с уникальным кодом; создать уникальную учетную запись администратора (не Admin); подключиться к Access от имени уникального администратора; создать пустую базу данных, проимпортировать в нее объекты

из старой базы данных. Удалить пользователя Admin из группы Admins. Отобразить у группы Users, в которой продолжает находиться Admin, все права на доступ к базе данных.

**Шифрование и дешифрование базы данных:** при шифровании базы данных ее файл сжимается и делается недоступным для чтения с помощью сторонних программ или текстовых редакторов (в Access база открывается обычным образом). Это предотвращает извлечение информации из базы данных в обход механизмов безопасности Access. Дешифрование базы данных отменяет результаты операции шифрования. Порядок шифрования: запустить Access без открытия базы данных (невозможно зашифровать или дешифровать открытую базу данных); меню «Сервис/Защита/Шифровать или Дешифровать/указать имя базы данных, указать имя зашифрованной базы данных».

### **Задания для выполнения**

Использовать базу данных, созданную в лабораторных работах № 1–4.

**Задание 1.** Установить пароль на базу данных, проверить его действие и снять пароль.

**Задание 2.** Выполнить шифрование базы данных, присвоив зашифрованной базе новое имя. Не открывая зашифрованную базу данных, запустить текстовый редактор WordPad («кнопка Пуск/Выполнить/набрать «WordPad»/ОК»), загрузить в него сначала незашифрованную базу данных, просмотреть ее, а затем загрузить в редактор зашифрованную базу данных и сравнить ее с незашифрованной. Дешифровать зашифрованную базу данных.

**Задание 3.** Установить защиту на уровне пользователей:

- создать новую группу пользователей «Работники». Создать нового пользователя «Директор» с правами администратора (участвует в группах Admins, Users и «Работники»). Со-

здать нового пользователя «Иванов» с правами работника предприятия и пользователя (участвует в группах Users и «Работники»). Создать пользователя «Новичок» с правами работника предприятия и пользователя. Создать пользователя «Гость» с правами пользователя (участвует только в группе Users). При создании пользователя «Гость» его индивидуальный «Код» задать в виде «Гость1234»;

- установить пароль «123456789» для пользователя Admin. Перезапустить Access. Войти в Access от имени пользователя «Директор» и установить ему пароль. Войти в Access от имени пользователей «Иванов» и «Новичок» и установить им пароли, для «Гость» пароль не устанавливать;

- войти в Access от имени «Директора». Открыть базу данных, созданную в предыдущих лабораторных. Запретить группе Users открывать базу данных. Разрешить группе «Работники» только просматривать существующие таблицы, запросы, формы и отчеты, а пользователю «Иванов» – еще и изменять введенные данные, но не изменять саму структуру базы данных как для уже существующих, так и для вновь создаваемых таблиц, запросов, форм и отчетов. Разрешить пользователю «Гость» открывать базу данных, но разрешить ему только просмотр формы «Форма заказа». Создать свою пустую базу данных с именем «Конфиденциально» и запретить пользователю «Гость» открывать эту базу данных (все остальные пользователи, входящие в группу «Работники», могут ее открывать). Для всех текущих или вновь создаваемых пользователей, кроме «Директора», запретить устанавливать или изменять пароль на базу данных;

- войти в Access последовательно как «Гость», «Новичок», «Иванов» и проверить действие ограничений на уже существующие таблицы. При подключении каждого из пользователей создавать произвольные новые таблицы, назвав их соответственно «Гость», «Новичок», «Иванов». Обратить внимание на то, что пользователь, создавший новую таблицу (запрос,

форму, отчет), является ее владельцем, а значит обладает любыми правами на нее, не зависимо от переданных ему администратором разрешений. В то же время другие пользователи относительно этой таблицы обладают лишь теми правами, которые им определил администратор для вновь создаваемых таблиц. Проверить это утверждение, попытавшись изменить структуру таблицы «Иванов», подключившись как «Новичок», а затем структуру таблицы «Гость», подключившись как «Гость». По вновь созданным таблицам передать права владельца «Директору»;

- войти в Access как «Директор», удалить пользователя «Гость» и снять пароль с пользователя «Новичок». Попытаться подключиться к Access как «Гость», подключиться как «Новичок» и заново задать для него пароль. Подключиться как «Директор» и заново создать пользователя «Гость» (входит только в группу Users), но с индивидуальным кодом «1234» (код не совпадает с кодом прежнего пользователя «Гость»). Подключиться как «Гость» и попытаться открыть базу данных. Подключиться как «Директор», удалить и заново создать пользователя «Гость» с индивидуальным кодом «Гость1234» (код совпадает с кодом прежнего пользователя «Гость»). Подключиться как «Гость» и открыть базу. Объяснить причины неудачи и успеха.

**Задание 4.** Изучить слабые места защиты на уровне пользователей:

- запустить администратора рабочих групп, определить используемый сейчас стандартный файл рабочей группы, название организации и имя пользователя. В своем каталоге под именем «Взломщик» создать пустую копию файла рабочей группы, введя такое же «Имя пользователя» и «Название организации» («Код» не вводить). Связать этот файл с Access (кнопка «Связать»). Подключиться к Access с файлом рабочей группы «Взломщик» как пользователь Admin без пароля, открыть и изменить базу данных, заново связать Access со старым файлом рабочей группы;

- устранить слабые места защиты. Создать новый файл рабочей группы с уникальным кодом и именем «Защищенный». Запустить Access с файлом рабочей группы «Защищенный», подключившись как пользователь Admin. Заново создать группы пользователей и имена пользователей, как это было сделано в предыдущем задании (пользователей «Новичок» и «Гость» можно не создавать). Создать пустую базу данных с именем «Временная» и импортировать в нее все таблицы, запросы, формы и отчеты из старой базы данных. Удалить пользователя Admin из группы Admins и установить ему пароль. Запустить Access с файлом рабочей группы «Защищенный». Подключившись как «Директор», создать пустую базу данных с именем «Нет лазеек» и импортировать в нее все таблицы, запросы, формы и отчеты из базы данных «Временная» (Внимание! Очень важно, чтобы при создании базы данных «Нет лазеек» был активен пользователь «Директор», т. к. если будет активен Admin, то он станет владельцем базы данных, а значит сможет открывать и изменять любые ее объекты, что сделает защиту бессмысленной). Заново определить все разрешения на доступ к базе данных «Нет лазеек» для пользователей и групп пользователей так, как это было сделано в предыдущем задании (Внимание! Важно отобрать у группы Users все права на доступ к базе данных, т. к. в этой группе все еще продолжает находиться пользователь Admin, а удалить его из этой группы невозможно). Удалить старую базу данных и базу данных «Временная» без использования корзины;

- подключиться к Access с файлом рабочей группы «Взломщик» как пользователь Admin без пароля и попытаться открыть базу данных «Нет лазеек». Подключиться к Access с файлом рабочей группы «Защищенный» как пользователь «Директор» и открыть базу данных «Нет лазеек». Объяснить причины неудачи и успеха.

## ЛИТЕРАТУРА

1. Вейскас, Дж. Microsoft Office Access 2003 / Дж. Вейскас; пер. В. Ахмадуллина, А. Маркова, А. Раздобарина. – СПб.: Питер, 2005. – 1167 с.
2. Грофф, Дж. Энциклопедия SQL / Дж. Грофф, Р.П. Вайнберг. – 3-е изд. – СПб.: Питер; Киев: BHV, 2004. – 895 с.
3. Официальный учебный курс Microsoft: Microsoft Office Access 2003 / пер. Л. Сазонова. – М.: ЭКОМ, БИНОМ. Лаборатория знаний, 2006. – 526 с.

## **СОДЕРЖАНИЕ**

Лабораторная работа № 1

**ОСНОВНЫЕ ПРИЕМЫ  
РАБОТЫ С MS ACCESS. СОЗДАНИЕ ТАБЛИЦ  
И СХЕМЫ ДАННЫХ..... 3**

Лабораторная работа № 2

**ОСНОВНЫЕ ПРИЕМЫ  
РАБОТЫ С MS ACCESS. СОЗДАНИЕ ЗАПРОСОВ,  
ИСПОЛЬЗОВАНИЕ ЯЗЫКА SQL В ЗАПРОСАХ ..... 11**

Лабораторная работа № 3

**ИСПОЛЬЗОВАНИЕ SQL  
В MS ACCESS. СОЗДАНИЕ ТАБЛИЦ  
И ИЗМЕНЕНИЕ ДАННЫХ ПРИ ПОМОЩИ SQL..... 21**

Лабораторная работа № 4

**ОСНОВНЫЕ ПРИЕМЫ РАБОТЫ С MS ACCESS.  
СОЗДАНИЕ ФОРМ, ОТЧЕТОВ И МАКРОСОВ ..... 26**

Лабораторная работа № 5

**СИСТЕМА БЕЗОПАСНОСТИ MS ACCESS ..... 34**

**ЛИТЕРАТУРА ..... 43**

Учебное издание

СЕТЕВЫЕ ТЕХНОЛОГИИ  
И БАЗЫ ДАННЫХ

Лабораторный практикум

В 3 частях

Часть 1

СУБД MS ACCESS

Составитель  
ЛЯХЕВИЧ Александр Генрихович

Редактор Т.А. Подолякова  
Компьютерная верстка Д.К. Измайлович

---

Подписано в печать 04.07.2011.

Формат 60×84<sup>1</sup>/<sub>16</sub>. Бумага офсетная.

Отпечатано на ризографе. Гарнитура Таймс.

Усл. печ. л. 2,62. Уч.-изд. л. 2,05. Тираж 100. Заказ 1179.

---

Издатель и полиграфическое исполнение:  
Белорусский национальный технический университет.

ЛИ № 02330/0494349 от 16.03.2009.

Проспект Независимости, 65. 220013, Минск.