

**Методические указания для самостоятельной работы студентов
по курсу «Системы управления базами данных»**

**Иваново
2019**

Министерство науки и высшего образования Российской Федерации
Ивановский государственный химико-технологический университет

**Методические указания для самостоятельной работы студентов
по курсу «Системы управления базами данных»**

Составитель Е.Я. Князева

УДК 658.512.011.56

Составитель Е.Я. Князева

Методические указания для самостоятельной работы студентов по курсу «Системы управления базами данных»/ сост. Е.Я. Князева; Иван. гос. хим.-технол. ун-т. – Иваново: ИГХТУ, 2019. – 60 с.

В методических указаниях рассматриваются основные сведения о работе с системами управления баз данных на примере Access, входящего в стандартный набор прикладных программ пакета Microsoft Office. А также включены примеры и задания для самостоятельной работы студентов.

Предназначены для студентов, изучающих дисциплины «Системы управления базами данных» по направлениям: «Автоматизация технологических процессов и производств», «Управление в технических системах» очной и заочной форм обучения.

Ил. 44. Табл. 5. Библиогр.: 9 назв.

Рецензент – кандидат технических наук А.А. Липин (Ивановский государственный химико-технологический университет)

Оглавление

Введение.....	4
1. Общие сведения о MS Access.....	5
1.1. Структура таблицы.....	5
1.2. Объекты базы данных.....	5
1.3. Типы данных.....	5
2. Работа с базой данных.....	6
Задание 1. Создание однотабличной базы данных.....	6
Задание 2. Заполнение базы данных.....	9
Задание 3. Размещение новых объектов в таблице.....	13
Задание 4. Создание новых таблиц.....	16
Задание 5. Ввод и просмотр данных посредством формы.....	19
Панель элементов и список полей.....	20
Задание 6. Создание схемы данных.....	24
Задание 7. Автоматический запуск формы из файла-приложения Access.....	27
Задание 8. Создание многотабличной формы.....	28
Задание 9. Создание вычисляемых полей в форме.....	30
Задание 10. Создание других кнопок на форме.....	32
Задание 11. Формирование запросов на выборку.....	34
Задание 12. Формирование запросов на обновление и удаление.....	39
Задание 13. Создание перекрестного запроса.....	41
Задание 14. Создание сводных таблиц и диаграмм.....	42
Задание 15. Создание отчетов.....	46
Задание 16. Создание макросов.....	47
Вопросы для самостоятельной работы.....	48
Самостоятельная работа № 1.....	50
Самостоятельная работа № 2.....	51
Самостоятельная работа № 3.....	51
Самостоятельная работа № 4.....	51
Задание на разработку базы данных.....	52
Варианты заданий на создание таблиц.....	54
Темы рефератов.....	57
Библиографический список.....	58

ВВЕДЕНИЕ

Реляционная модель данных была предложена Е.Ф. Коддом (Dr. E.F. Codd), известным исследователем в области баз данных, в 1969 году.

Реляционная база данных представляет собой хранилище данных, содержащее набор двумерных таблиц.

Система управления базами данных Access (СУБД Access) входит в стандартный набор прикладных программ пакета Microsoft Office, который используется практически в каждой организации нашей страны.

Создание базы данных включает в себя:

- теоретическое продумывание, проектирование структуры будущих таблиц и их взаимных связей в базе данных;
- создание структуры этих таблиц на компьютере;
- установление связей таблиц базы данных;
- заполнение таблиц реальными данными (с использованием форм).

Система управления базами данных (СУБД) – это комплекс языковых и программных средств, предназначенный для создания, ведения и совместного использования баз данных несколькими пользователями. Назначение любой СУБД – создание, ведение и обработка баз данных. СУБД предоставляет значительные возможности по работе с хранящимися данными, их обработке и совместному использованию.

Большинство современных СУБД, используемых на персональных компьютерах, ориентированы на реляционные модели данных, представляющие собой совокупность двумерных таблиц.

Одной из наиболее популярных систем управления реляционными базами данных является Microsoft Access, входящая в состав пакета Microsoft Office. Достоинствами программного средства являются удобный интерфейс, понятные инструменты и различные мастера для создания таблиц, форм и отчетов. На создание таблиц с данными, их редактирование, разработку запросов и поиск нужных данных затрачивается минимум усилий.

СУБД Access предоставляет пользователю широкие возможности ввода, обработки и представления данных. Эти средства не только удобны, но и высокопродуктивны, что обеспечивает высокую скорость разработки приложений. Эти возможности совершенствуются от версии к версии, и сейчас СУБД Microsoft Access стала инструментом, который может удовлетворить самые разные категории пользователей: от новичка до профессионального разработчика.

1. ОБЩИЕ СВЕДЕНИЯ О MS ACCESS

MS Access является приложением Windows.

В СУБД Access предусмотрено много дополнительных сервисных возможностей. *Мастера* помогут создать таблицы, формы или отчеты из имеющихся заготовок. *Выражения* используются в Access, например, для проверки допустимости введенного значения. *Макросы* позволяют автоматизировать многие процессы без программирования, тогда как встроенный в Access язык VBA (Visual Basic for Applications) дает возможность опытному пользователю программировать сложные процедуры обработки данных.

1.1. Структура таблицы

При разработке структуры таблицы, прежде всего, необходимо определить названия полей, из которых она должна состоять, типы полей и их размеры. Каждому полю таблицы присваивается уникальное имя, которое не может содержать более 64 символов. Имя желательно делать таким, чтобы функция поля узнавалась по его имени. Далее надо решить, данные, какого типа будут содержаться в каждом поле. В MS Access можно выбирать любые из основных типов данных. Один из этих типов данных должен быть присвоен каждому полю. Значение типа поля может быть задано только в режиме конструктора.

В Access существует несколько способов создания пустой таблицы:

- **мастер** таблиц;
- ввод данных непосредственно в пустую таблицу в **режиме таблицы**;
- определение всех параметров макета таблицы в **режиме конструктора**.

1.2. Объекты базы данных

Объектами базы данных являются:

Таблицы – совокупность записей, где хранится основная информация.

Форма – представляет собой специальный формат экрана, используются для ввода данных в таблицу и просмотра одной записи.

Запрос – это инструмент для анализа, выбора и изменения данных. С помощью Access могут создаваться несколько видов запросов.

Отчеты – это средство организации данных при выводе на печать.

Из всех типов объектов только таблицы предназначены для хранения информации. Остальные используются для просмотра, редактирования, обработки и анализа данных – иначе говоря, для обеспечения эффективного доступа к информации.

1.3. Типы данных

Текстовый – наиболее чисто используемый в Access тип данных. Этот тип данных подходит для хранения адресов, для полей с кратким описанием, для числовых данных, не требующих расчетов, таких как телефонные номера и почтовые индексы. Длина – 255 символов.

Поле Мемо – предназначен для полей, длина которых превосходит 255 символов. Пример: длинное поле описания. Поле Мемо может хранить до

65 535 символов, что приближенно равно 32 страницам текста.

Числовой – данные, используемые для математических вычислений, за исключением финансовых расчетов (для них следует использовать тип «**Денежный**»).

Дата/время – значения дат и времени. Сохраняет 8 байтов. Можно вводить даты с 1 января 100 года по 31 декабря 9999 года. Access предлагает несколько различных форматов дат.

Денежный – используется для денежных значений и для предотвращения округления во время вычислений, для выполнения вычислений над полем, которое содержит числа, в левой части которых не более 15 знаков, а справа от запятой не более четырех знаков.

Счетчик – автоматическая вставка уникальных последовательных (увеличивающихся на 1) или случайных чисел при добавлении записи с использованием этого типа данных либо, выбрав соответствующий пункт в свойстве **Новое значение** этого поля. Если удалить одну из последовательных записей, этот тип поля не запомнит и не перенумерует удаленное значение. Это значение будет просто отсутствовать.

Логический (Да/нет) – данные, принимающие только одно из двух возможных значений, таких как «Да/Нет», «Истина/Ложь», «Вкл/Выкл». Значения Null не допускаются.

Поле объекта OLE – объекты OLE (такие как документы Microsoft Word, электронные таблицы Microsoft Excel, рисунки, звукозапись или другие данные в двоичном формате), созданные в других программах, использующих протокол OLE.

Гиперссылка – может иметь вид пути UNC либо адреса URL.

Мастер подстаново – создает поле, позволяющее выбрать значение из другой таблицы или из списка значений, используя поле со списком.

Необходимо соблюдать для одних наименований полей данных одинаковый тип данных.

2. РАБОТА С БАЗОЙ ДАННЫХ

ЗАДАНИЕ 1. СОЗДАНИЕ ОДНОТАБЛИЧНОЙ БАЗЫ ДАННЫХ

РАБОТА С ПОЛЯМИ

Цель работы. Приобрести навыки и умения при работе с таблицами: создавать таблицы с помощью конструктора, задавать тип данных, задавать маску ввода для поля, использовать свойства поля, создавать ключи и индексы для полей таблицы, заполнять таблицы данными и создавать формы.

1. Создание новой базы данных.

– загрузите MS Access, в появившемся окне выберите пункт **Новая база данных**;

– задайте имя вашей базы. По умолчанию MS Access предлагает вам имя

базы *db1* (Access 2007 – *Database1.accdb*), а тип файла – *Базы данных Access*.
Имя задайте *Стоматология*.

2. Создание таблицы базы данных.

В окне базы данных выберите вкладку *Создание*, а затем щелкните по кнопке «**Конструктор таблиц**» (рис. 1).

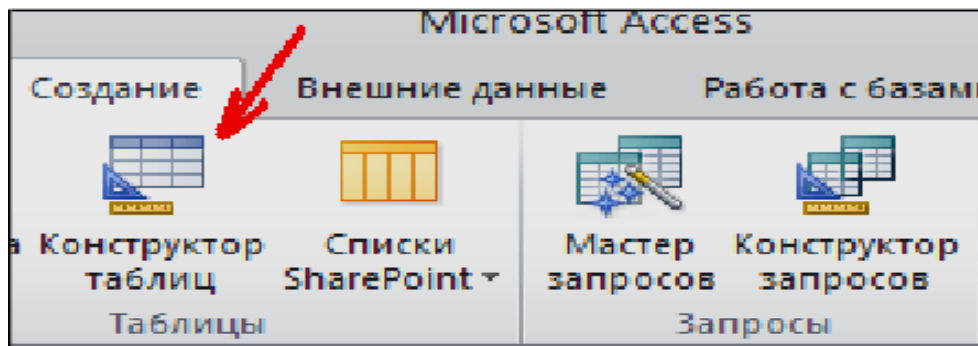


Рис. 1. Создание таблиц в режиме Конструктора

3. Определение полей таблицы.

– Введите в строку столбца **Имя поля** имя первого поля - *ID врача*.

– В строке столбца **Тип данных** щелкните по кнопке списка и выберите тип данных *Числовой*. Поля вкладки *Общие* оставьте такими, как предлагает Access.

– Для определения всех остальных полей таблицы базы данных «*Врачи*» выполните действия, аналогичные указанным выше в соответствии с табл. 1.

Таблица 1

Описание свойств полей таблицы *Врачи*

Имя поля	Тип данных	Размер поля (вкладка Общие)
ID врача	Числовой	
ФИО	Текстовый	20
Должность	Текстовый	20
№ кабинета	Числовой	Длинное целое (по умолчанию)
Время приема	Текстовый	12
ВидОказываемыхУслуг	Текстовый	50 (Максимальное значение)

4. Задайте условие для поля «№ кабинета».

Пусть по условию в поликлинике 2 этажа и значение номера кабинета находится в пределах от 100 до 130 и от 200 до 230.

В строке «*Условие на значение*» перейдите в режим **Построителя выражений** (рис. 2) и введите условие: $\geq 100 \text{ And } \leq 130 \text{ Or } \geq 200 \text{ And } \leq 230$. В строке «*Сообщение об ошибке*» введите: «Такого номера кабинета нет» (рис. 3).

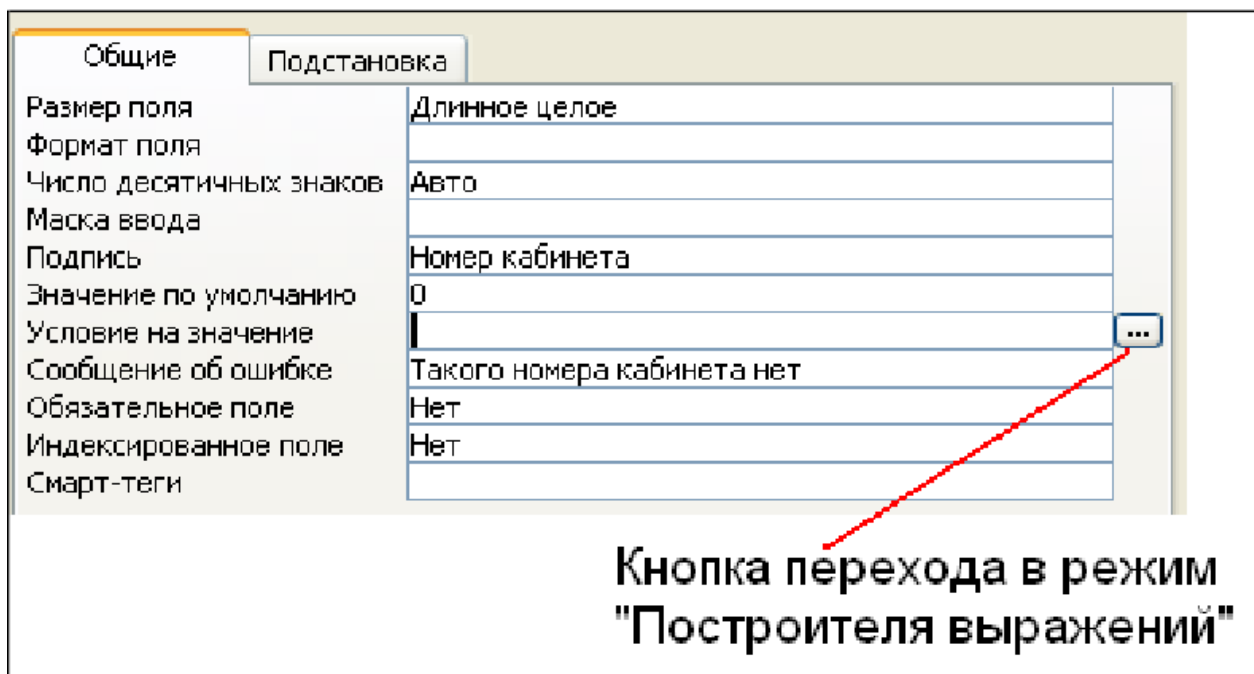


Рис. 2. Кнопка перехода в режим «Построителя выражений»

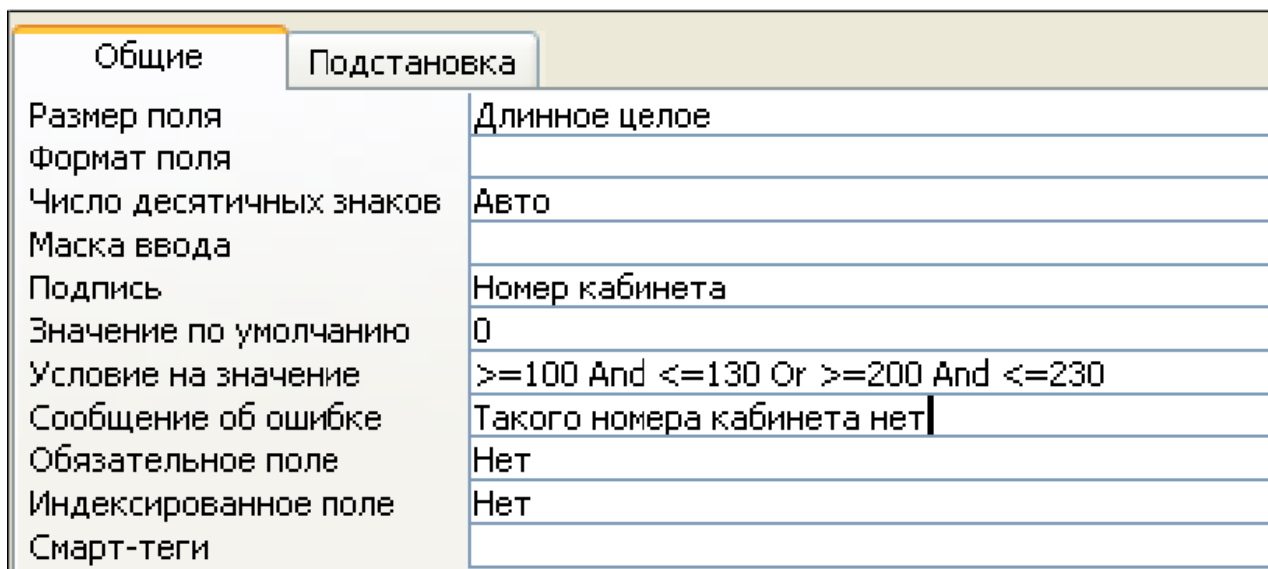



Рис. 3. Вкладка «Общие» с заданными условиями для поля «№ кабинета»

5. Создание ключевых полей.

Первичный ключ – это поле (или ряд полей), которое однозначно идентифицирует каждую запись в таблице. Access не допускает повторных значений в поле первичного ключа.

Сделайте поле **ID врача** ключевым. Для этого откройте таблицу «*Врачи*» в режиме **Конструктора** и, поместив курсор на имя этого поля, щелкните по кнопке  – **Ключевое поле** (рис. 4):

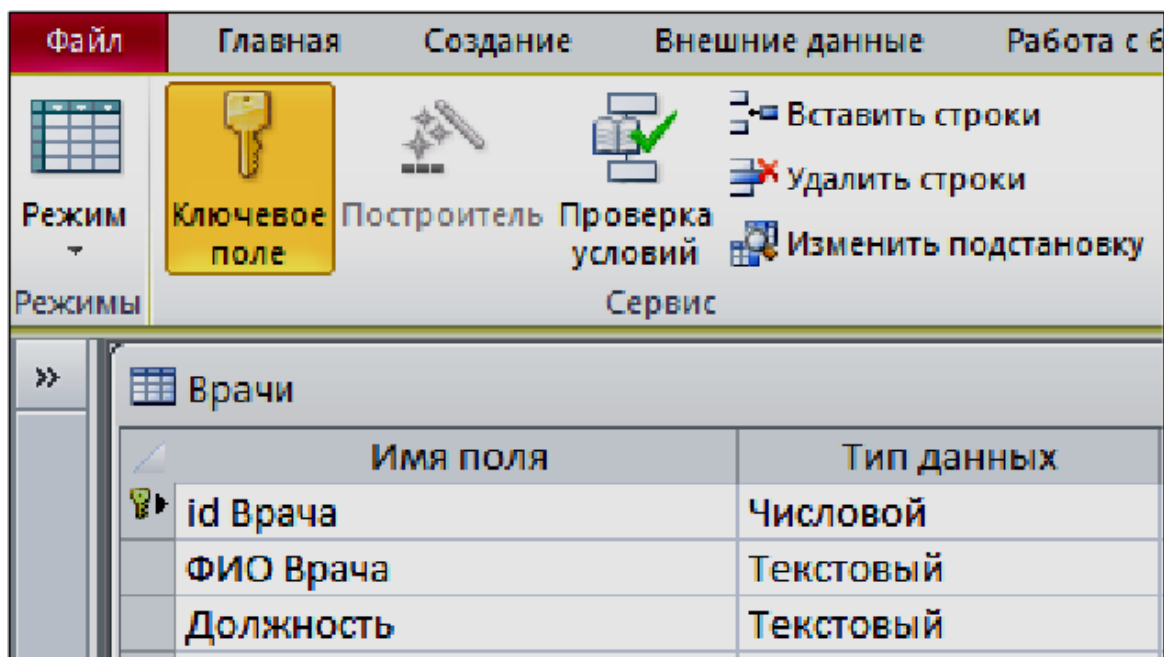


Рис. 4. Назначение поля ключевым

Закройте таблицу. MS Access предложит сохранить таблицу (по умолчанию «Таблица1»). Задайте имя таблицы «Врачи».

ЗАДАНИЕ 2. ЗАПОЛНЕНИЕ БАЗЫ ДАННЫХ

Цель работы. Приобрести навыки и умения при работе с таблицами: создавать таблицы-подстановки; заполнять таблицы данными; создавать подстановочное поле; вводить ограничения на данные в поле ID врача, эти данные не должны повторяться.

1. Создание таблиц-подстановок.

Для упрощения ввода данных в таблицу «Врачи» создайте таблицы-подстановки для полей: «Время Приема» и «Вид Оказываемых Услуг». Для поля «Должность» данные будут подставляться из таблицы «Штатное расписание». Подстановка упрощает режим ввода. При этом нет необходимости набирать длинные названия несколько раз. Достаточно будет выбрать необходимую запись из выпадающего списка.

2. Таблица «Смена_подстановка» используется для поля «Время Приема», поля таблицы представлены на рис. 5.

Смена_подстановка : таблица	
Имя поля	Тип данных
▶ НомерСмены	Текстовый
ВремяПриема	Текстовый

Рис. 5 Описание свойств полей таблицы «Смена_подстановка»

Примечание. В результате щелчка по кнопке **<ОК>** MS Access предложит вам задать ключевое поле. Для создаваемой таблицы щелкните по кнопке **<Нет>**.

3. Таблицу «*НаименованиеУслуги_подстановка*», используемую для поля «*Вид оказываемых услуг*», создайте в **Excel**, чтобы импортировать её в нашу базу данных. Поля таблицы представлены в табл. 2.

Таблица 2

Таблица с данными «*НаименованиеУслуги_подстановка*»

Наименование направления услуги
Терапевтическая
Хирургическая
Ортопедическая

Импорт как самих таблиц, так и полей таблиц можно производить из разных источников и файлов разных форматов.

- Определите формат ячеек: для поля *НаименованиеУслуги* – текстовый. Сохраните таблицу в той же папке, где находится ваша БД *Стоматология*.

- В базе данных *Стоматология* на панели инструментов выберите **Внешние данные → Excel**. В открывшемся диалоговом окне выберите созданную таблицу:

- В следующем окне «*Импорт электронной таблицы*» выберите *Лист 1*, нажмите **<Далее>**.

- Установите флажок «*Первая строка содержит заголовки столбцов*»; далее – «*Данные необходимо сохранить в новой таблице*», затем подтвердите описание и название полей; далее выберите «**Не создавать ключ**». Готово!

Таблица «*НаименованиеУслуги_подстановка*» будет импортирована в вашу БД. Посмотрите ее в режиме **Конструктора**. Проверьте все ли поля соответствуют **типу данных**.

4. Таблица «*Штатное расписание*». Данные поля «*Наименование Должности*» будут использоваться для подстановки в таблицу «*Врачи*» (рис. 6).

Имя поля	Тип данных
Должность	Текстовый
Оклад	Денежный
Надбавка	Денежный

Рис. 6. Описание свойств полей таблицы «*Штатное расписание*»

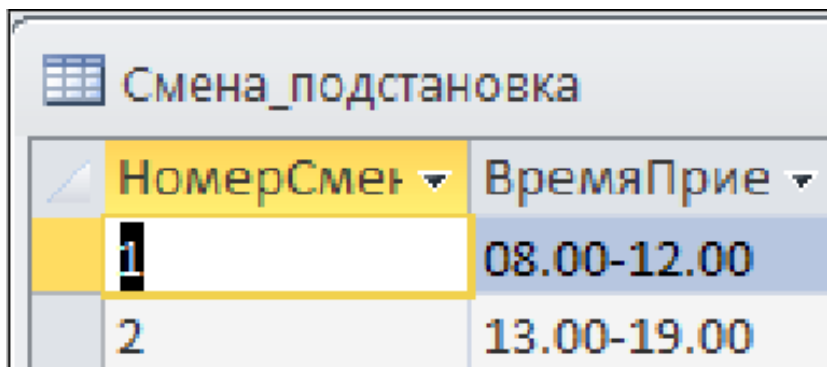
Для поля «Оклад» во вкладке **Общие** в строке «*Формат поля*» выбрать «*Денежный*». Для поля «Должность» задать значение **Ключевое поле**.

5. Заполнение *таблиц* данными.

Для заполнения таблиц данными необходимо перейти из режима **Конструктора** в режим **Таблицы** или открыть таблицу двойным щелчком.

Внимание! Данные в таблицу можно ввести только в режиме **Таблицы**

– Заполните таблицу «Смена_подстановка» данными (рис. 7).



НомерСмеи	ВремяПрие
1	08.00-12.00
2	13.00-19.00

Рис. 7. Таблица «Смена_подстановка» с данными

– Заполните таблицу «ШтатноеРасписание» данными (рис. 8).



	Должность	Оклад	Надбавка
+	врач-стоматолог	7 000,00р.	700,00р.
+	ортопед	14 000,00р.	1 200,00р.
+	хирург-терапевт	10 000,00р.	1 000,00р.


Рис. 8. Таблица «ШтатноеРасписание» с данными

6. Для создания *подстановочного поля* используется три способа:

- режим **Конструктора**. Этот способ используется в том случае, если поле для подстановки стоит первым в подстановочной таблице;
- использование **Мастера подстановок**;
- использование языка запросов SQL (с помощью построителя запросов).

Откройте таблицу «Врачи» в режиме **Конструктора**. Для поля «Должность» во вкладке **Подстановка** выберите в строке **Тип элемента управления**: «Поле со списком»; в строке **Тип источника строк**: «Таблица или запрос» (задается по умолчанию); в строке **Источник строк** из выпадающего списка выбрать таблицу «Штатное расписание» – то есть таблицу, из которой данные будут подставляться в таблицу «Врачи».

7. Для создания подстановки поля «*ВремяПриема*» в столбце **Тип данных** выберите из выпадающего списка «**Мастер подстановки**». В следующем окне нажмите кнопку <Далее>, в открывшемся диалоговом окне выберите «*Таблица: Смена_подстановка*». Нажмите <Далее>.

В следующем окне «*Создание подстановки*» из списка «*Доступные поля*» выберите при помощи кнопки  поле «*ВремяПриема*» (данные из этого поля будут подставляться в нашу таблицу), нажмите кнопку <Далее>, затем <Готово> (рис. 9).

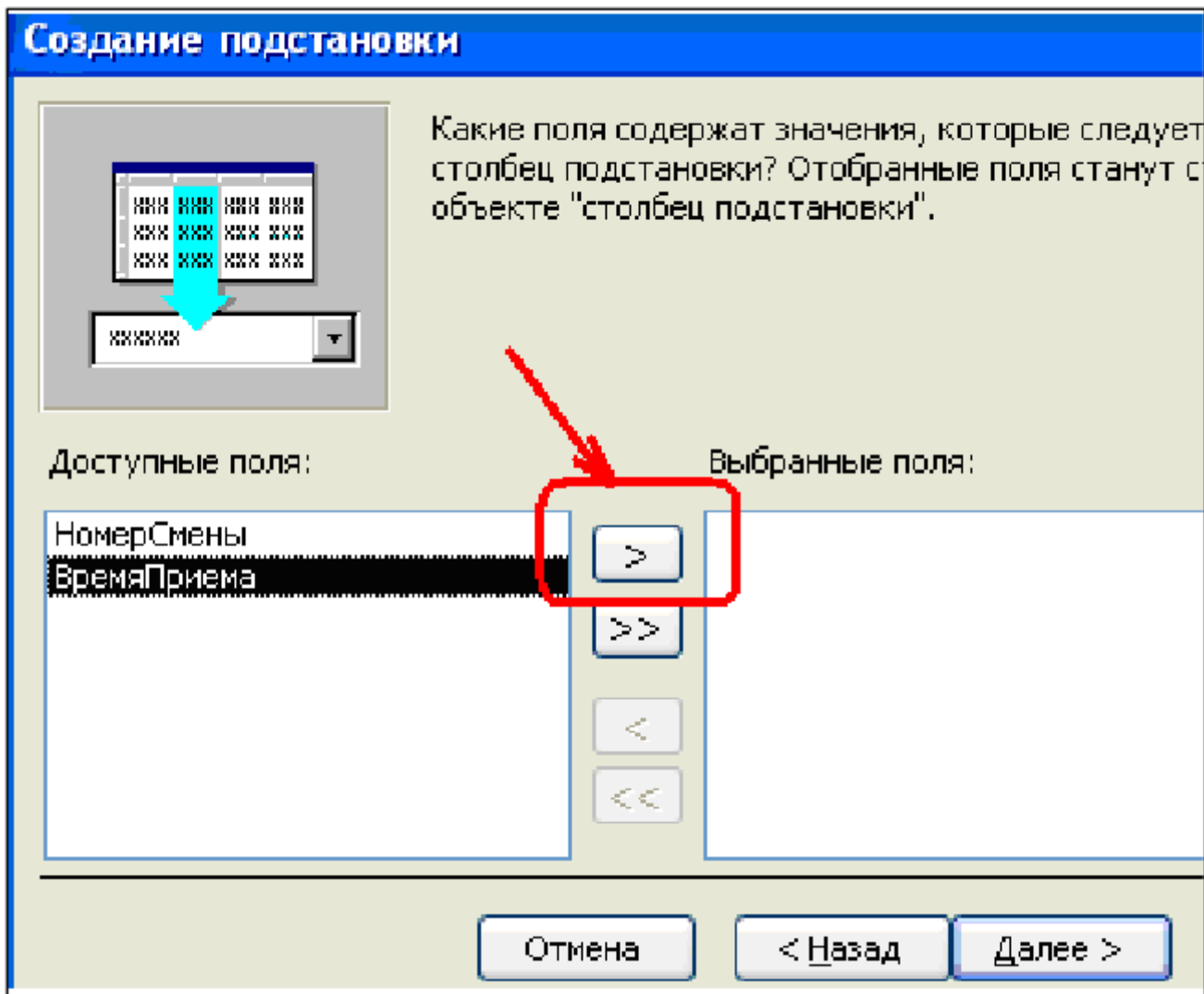



Рис. 9. Использование *Мастера подстановки*

8. Использование языка запросов *SQL*. Для использования построителя запросов для поля «*ВремяПриема*» выберите вкладку **Подстановка**:

- в строке **Тип элемента управления** - «*Поле со списком*»,
- в строке **Тип источника строк** – «*Таблица или запрос*»
- в строке **Источник строк** нажмите на кнопку **Построителя выражений** 

и в открывшемся диалоговом окне **Инструкция SQL: построитель запросов** на панели инструментов выберите: **Вид** → **Режим SQL**. Запрос бу-

дет иметь вид:

```
SELECT Смена_подстановка.ВремяПриема FROM Смена_подстановка.
```

САМОСТОЯТЕЛЬНО: для поля «ВидОказываемыхУслуг» создайте подстановочное поле.

Ввод ограничения на данные в поле «ID врача»

Для поля «ID врача» в таблице «Врачи» выполним условия:

- коды врачей не должны повторяться;
- должна быть обеспечена возможность их изменения (из-за последнего условия в этом поле нельзя использовать тип данных **Счетчик**, в котором данные не повторяются).

Для выполнения этих условий сделайте следующее:

- установите курсор на строке «ID врача»;
- во вкладке **Общие** щелкните по строке параметра **Индексированное поле**;
- выберите в списке пункт «Да (совпадения не допускаются)»;
- перейдите в режим **Таблица**.

***Примечание.** Индекс – это средство MS Access, ускоряющее поиск и сортировку данных в таблице.*

Заполните таблицу «Врачи» данными, используя подстановку.

ЗАДАНИЕ 3. РАЗМЕЩЕНИЕ НОВЫХ ОБЪЕКТОВ В ТАБЛИЦЕ

1. Размещение объекта OLE.

Рассмотрим размещение объекта OLE в таблице «Врачи» на примере поля «ФотоВрача». В нашем примере фотографии хранятся в формате графического редактора **Paint** в файлах с расширением **bmp**. (Создайте несколько таких объектов и сохраните их в той же папке, где хранится база данных).

– Откройте таблицу «Врачи» в режиме **Конструктора** и добавьте новое поле «ФотоВрача». Тип данных – **Поле объекта OLE**.

– Перейдите в режим **Таблицы**. В графе «Фото» выполните команду правой кнопкой мыши: **Добавить объект**.

– В окне **Вставка объекта** (рис. 10) отметьте переключатель «Создать из файла». Окно **Вставка объекта** позволяет ввести имя файла с фотографией. Для поиска файла воспользуйтесь кнопкой **Обзор**.

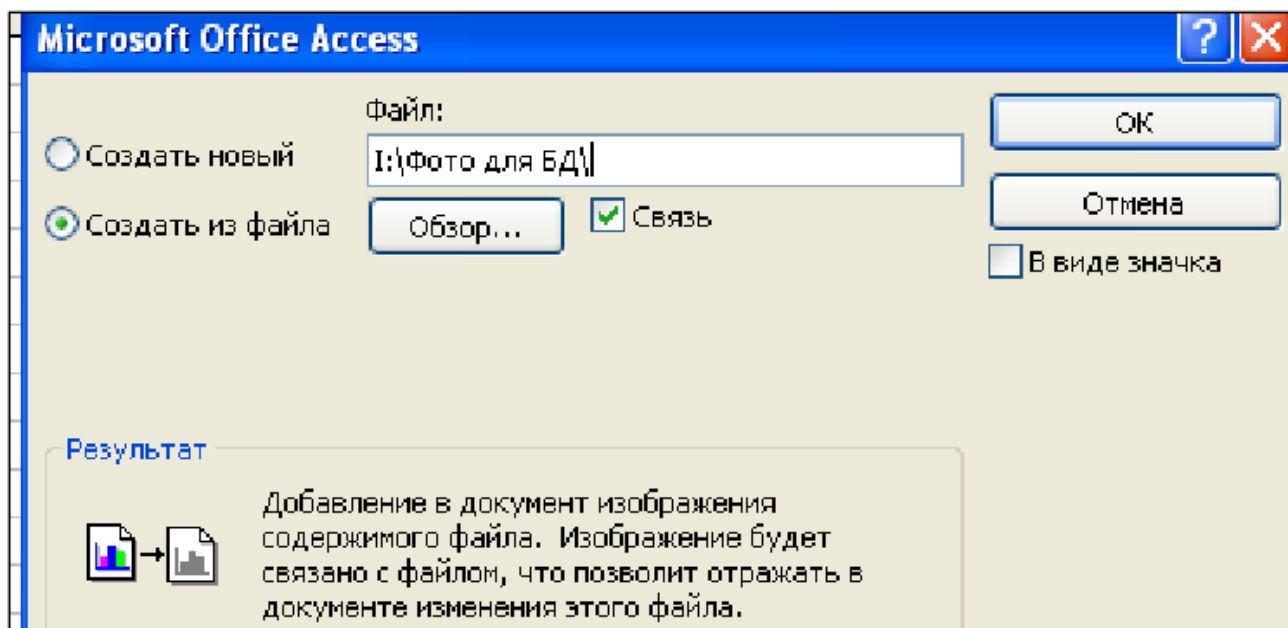



Рис. 10. Добавление объекта.

Для введения в поле связанного объекта надо установить флажок «Связь», что позволит в дальнейшем отражать в документе изменения этого файла.

2. Добавление нового поля в таблицу «СОЗДАНИЕ МАСКИ ВВОДА».

- Откройте таблицу «Врачи» в режиме Конструктора.
- Добавьте новое поле «Телефон». Тип данных – **Текстовый**.
- Задайте маску ввода для поля **Телефон** (в виде 22-22-222), для этого:
 - Щелкните на вкладке **Общие** в строке *Маска ввода* по кнопке .
 - Подтвердите сохранение таблицы.
 - В открывшемся диалоговом окне «Создание масок ввода» щелкните по кнопке <Список>. В появившемся диалоговом окне «Настройка масок ввода» выполните действия по образцу (рис. 11).
 - Нажмите кнопку <Закреть> и из появившегося списка выберите созданную маску ввода для поля **Телефон**.

В таблице 3 приведены допустимые знаки при работе с масками ввода.

Примечание. Маска ввода создается только для поля, имеющего тип данных – **текстовый!**

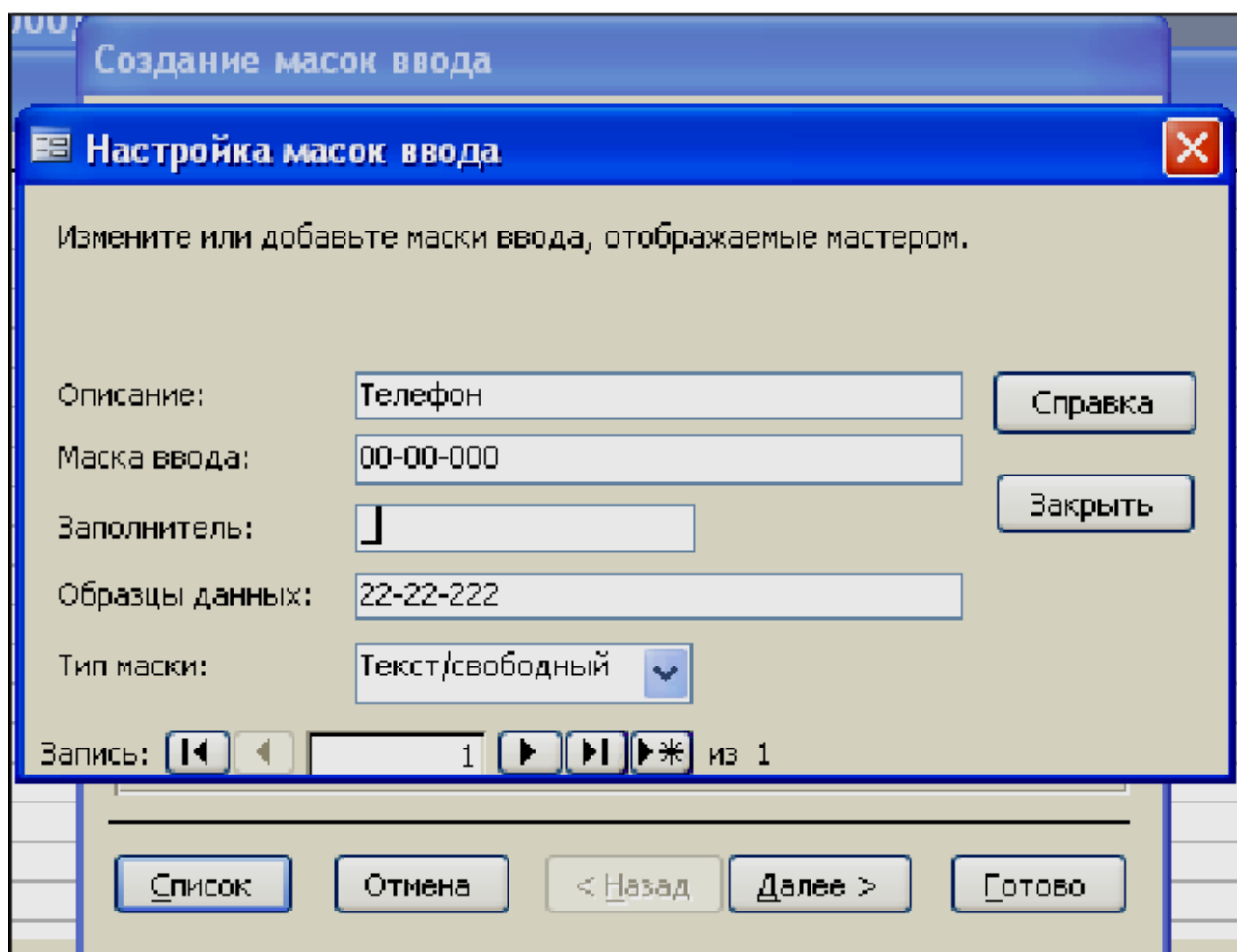


Рис. 11. Создание маски ввода


Таблица 3

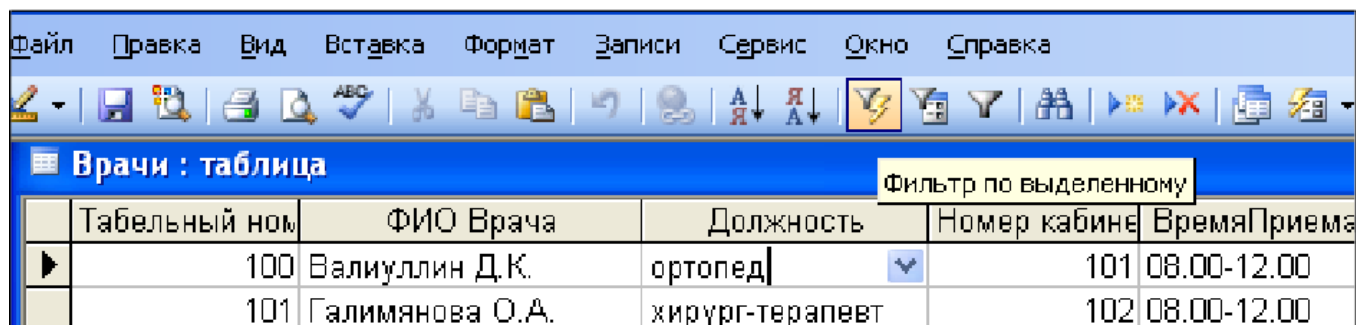
Некоторые допустимые знаки масок ввода

Знак	Описание
0	Цифра (от 0 до 9, ввод обязателен; знаки плюс [+] и минус [-] не допускаются).
9	Цифра или пробел (ввод не обязателен; знаки плюс [+] и минус [-] не допускаются).
#	Цифра или пробел (ввод не обязателен; пустые знаки преобразуются в пробелы, допускаются знаки плюс [+] и минус [-]).
L	Буква (от А до Z или от А до Я, ввод обязателен).
?	Буква (от А до Z или от А до Я, ввод не обязателен).
A	Буква или цифра (ввод обязателен).
&	Любой знак или пробел (ввод обязателен).

3. Фильтрация данных по полям.

В режиме **Таблица** в таблице «Врачи» щелкните по записи *Ортопед* поля «Должность»:


– щелкните по кнопке  или выполните команду **Записи, Фильтр, Фильтр по выделенному**. В таблице останутся только записи о врачах-ортопедах (рис. 12);



The screenshot shows a software window titled "Врачи : таблица". The menu bar includes "Файл", "Правка", "Вид", "Вставка", "Формат", "Записи", "Сервис", "Окно", and "Справка". The toolbar contains various icons, including a filter icon. The table has the following data:

	Табельный ном	ФИО Врача	Должность	Номер кабинё	ВремяПриема
▶	100	Валиуллин Д.К.	ортопед	101	08.00-12.00
	101	Галимянова О.А.	хирург-терапевт	102	08.00-12.00

Рис. 12. Фильтр данных

– для отмены фильтрации щелкните по кнопке  на панели инструментов или выполните команду **Записи, Удалить фильтр**. В таблице появятся все данные.

ЗАДАНИЕ 4. СОЗДАНИЕ НОВЫХ ТАБЛИЦ

1. Создайте таблицу «СписокПациентов», которая содержит следующие поля (рис. 13):

- поле «МедПолис» создайте маску ввода по образцу: «АБ-1234»;
- поле «Телефон» создайте маску ввода как в задании 3;
- поле «Пол» во вкладке **Общие** в строке «Значение по умолчанию» введите: *муж*. Это значение будет указано по умолчанию. В случае необходимости его можно заменить.



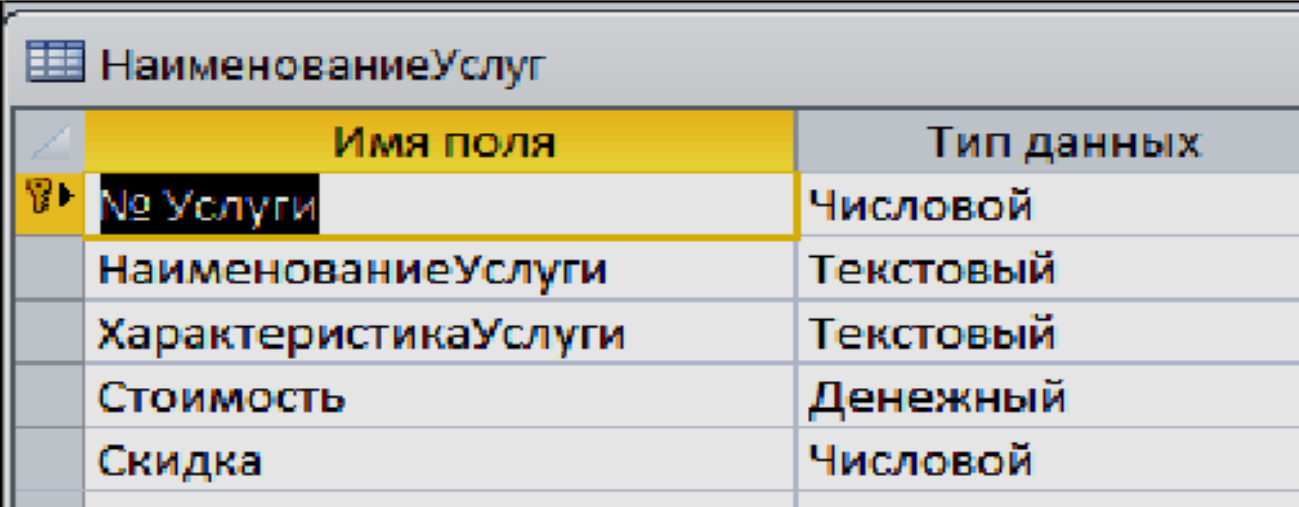
The screenshot shows a software window titled "СписокПациентов : таблица". The table has the following structure:

	Имя поля	Тип данных
?	НомерКарточки	Числовой
	ФИО Пациента	Текстовый
	ДомашнийАдрес	Текстовый
	МедПолис	Текстовый
	Телефон	Текстовый
▶	Пол	Текстовый

Рис. 13. Поля таблицы «СписокПациентов».

Заполните таблицу данными (не менее 15 записей).

2. Создайте таблицу «*НаименованиеУслуг*», которая содержит следующие поля (рис. 14):



	Имя поля	Тип данных
🔑	№ Услуги	Числовой
	НаименованиеУслуги	Текстовый
	ХарактеристикаУслуги	Текстовый
	Стоимость	Денежный
	Скидка	Числовой

Рис. 14. Поля таблицы «*НаименованиеУслуг*»

Заполните таблицу данными по следующему примеру, изображенному на рис. 15.

3. Создайте таблицу «*Журнал*» (рис. 16), которая содержит следующие поля:

– поле «*НомерКарточки*», используйте следующую подстановку:

```
SELECT СписокПациентов.НомерКарточки,СписокПациентов.  
[ФИО Пациента] FROM СписокПациентов;
```

– поле «*ФИО пациента*» и «*Должность врача*» не заполняйте. Для их заполнения будет использован **запрос на обновление** (см. Задание № 11);

– поле «*ФИО врача*», используйте для него подстановочное поле из таблицы «*Врачи*»;

– поле «*Услуги*», используйте в нём подстановку из таблиц услуг (таблица «*НаименованиеУслуг*»).

НаименованиеУслуг				
№ Услуги	НаименованиеУслуги	ХарактеристикаУслуг	Стоимость	Скидка
1	Лечение однокорневого пульпита	Терапевтическая	1 250,00р.	3
2	Лечение двухкорневого пульпита	Терапевтическая	1 650,00р.	1
3	Лечение трехкорневого пульпита	Терапевтическая	2 000,00р.	0
4	Лечение кариеса	Терапевтическая	500,00р.	5
5	Светополимерная композитная пломба (Рос	Терапевтическая	350,00р.	0
6	Лечение периодонтита	Терапевтическая	1 500,00р.	5
7	Удаление зубного камня – (один зуб)	Хирургическая	60,00р.	0
8	Чистка всех зубов одной челюсти (ультразву	Хирургическая	800,00р.	0
9	Профессиональное отбеливание - (один зуб)	Хирургическая	300,00р.	3
10	Удаление зуба	Хирургическая	150,00р.	0

Рис. 15. Примерные данные для таблицы «НаименованиеУслуг»

Журнал : таблица			
	Имя поля	Тип данных	
▶	ДатаПриема	Дата/время	
	НомерКарточки	Числовой	
	ФИО Пациента	Текстовый	
	ФИО Врача	Текстовый	
	ДолжностьВрача	Текстовый	
	Услуги	Текстовый	
	Количество	Числовой	

Общие	Подстановка
Формат поля	Краткий формат даты

Рис. 16. Таблица «Журнал»

ЗАДАНИЕ 5. ВВОД И ПРОСМОТР ДАННЫХ ПОСРЕДСТВОМ ФОРМЫ

- 1) Создать формы для ввода данных с последующей их модификацией.
- 2) Создать главную форму «Заставка».

ЭТАПЫ РАБОТЫ

Формы являются основным средством организации интерфейса пользователя в приложениях Access.

Для создания формы:

- Выберите таблицу, для которой будет создана форма, зайдите во вкладку **Создание**, выберите **Форма**.
- Автоматически созданная форма откроется в главном окне. Access предложит сохранить форму по названию таблицы.
- Создавать и редактировать формы любой степени сложности позволяет только **Конструктор** форм (рис. 17).

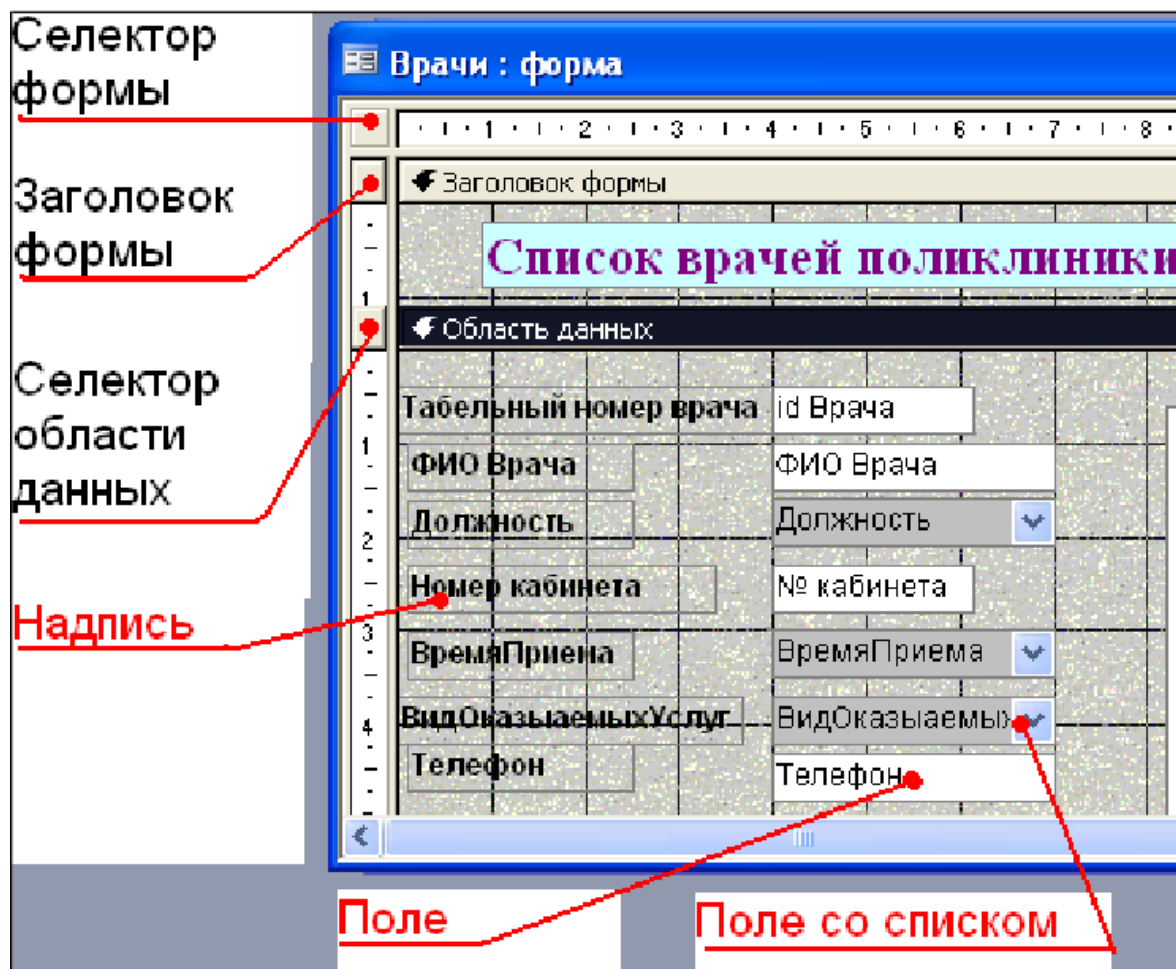


Рис. 17. Форма в режиме Конструктора

ПАНЕЛЬ ЭЛЕМЕНТОВ И СПИСОК ПОЛЕЙ

Панель элементов появляется в режиме **Конструктора** форм и используется для размещения объектов в форме. Ниже приведены наиболее используемые элементы формы и соответствующие им кнопки на Панели элементов.



– **Надпись.** Размещение в форме произвольного текста.



– **Поле.** Размещение в форме данных из соответствующего поля базовой таблицы/запроса, вывод результатов вычислений, а также прием данных, вводимых пользователем.



– **Выключатель.** Создание выключателя, кнопки с фиксацией.



– **Переключатель.** Создание селекторного переключателя.



– **Флажок.** Создание контрольного переключателя.



– **Поле со списком.** Размещение элемента управления, объединяющего поле и раскрывающийся список.



– **Список.** Создание списка, допускающего прокрутку. В режиме формы выбранное из списка значение можно ввести в новую запись или использовать для замены уже существующего значения.



– **Кнопка.** Создание командной кнопки, позволяющей осуществлять разнообразные действия в форме (поиск записей, печать отчета, установка фильтров и т.п.).



– **Рисунок.** Размещение в форме рисунка, не являющегося объектом OLE.



– **Подчиненная форма/отчет.**



– **Линия (Прямоугольник)** – элементы оформления. Размещение в форме линии для отделения логически связанных данных.



– **Свободная рамка объекта** - любой объект Windows-приложений, редактирование которого будет возможно вызовом соответствующего приложения.

Для создания «главной кнопочной формы» необходимо создать *управляющие кнопки*.

Кнопки используются в форме для выполнения определенного действия или ряда действий. Например, можно создать кнопку, которая будет открывать таблицу, запрос или другую форму. Можно создать набор кнопок для перемещения по записям таблицы.

• На панели инструментов выберите вкладку **Создание** → **Конструктор форм**. Появится пустая форма. Задайте мышкой размеры формы. (*Откроется вкладка **Конструктор** - панель элементов, которая позволяет создавать элементы управления на форме и осуществлять необходимые действия при конструировании* (рис. 18)).

• Выберите на панели инструментов и активируйте **Кнопку**.

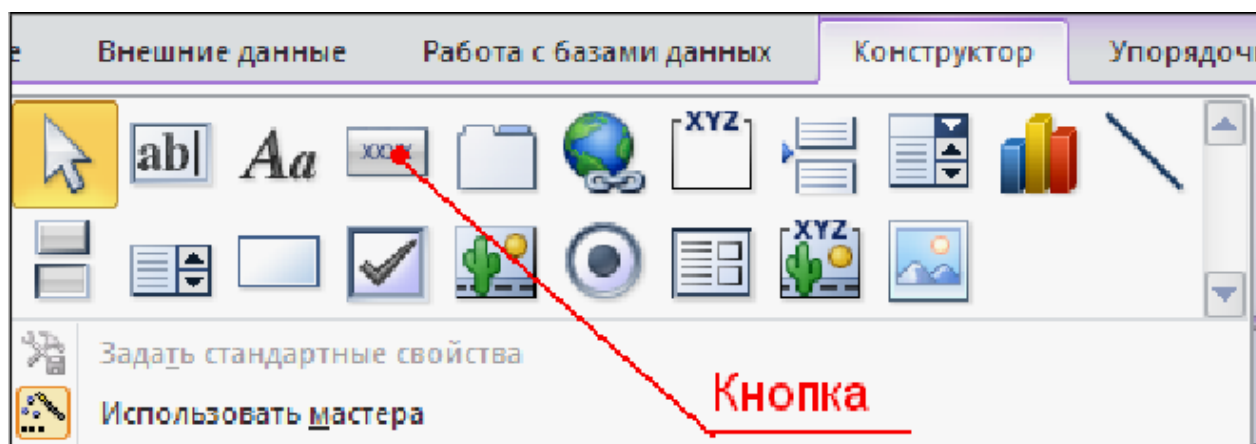



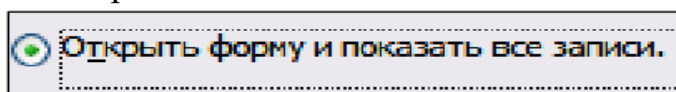
Рис. 18 Элементы управления

- Создайте заголовок формы. Для этого выберите кнопку  – **Надпись**, щелкнув по ней, расширьте область заголовка формы и введите в поле надпись **База данных «Врачи»**. Измените размер и цвет шрифта.

- Выберите на панели инструментов **Кнопку**. Щелкните мышкой по тому месту в области данных, где должна быть кнопка. Появится диалоговое окно **Создание кнопок**.

- Выберите категорию **Работа с формой**, а действие – **Открыть формы**. Нажмите кнопку <Далее>. Выберите форму **«Врачи»**, которая будет открываться этой кнопкой, нажмите кнопку <Далее>.

- Далее, оставьте переключатель в положении:



- В следующем окне поставьте переключатель в положение **Текст**, наберите **Список врачей**. Нажмите <Далее>.

- Задайте имя кнопки **Список** и нажмите <Готово>.

Выполните САМОСТОЯТЕЛЬНО: Добавьте кнопку выхода или закрытия главной формы.

- Изменить макет элементов управления можно, щелкнув правой кнопкой мыши по элементу и выбрав **Свойства** в режиме **Конструктор** формы. При этом открывается диалоговое окно свойств элементов управления (рис. 19):

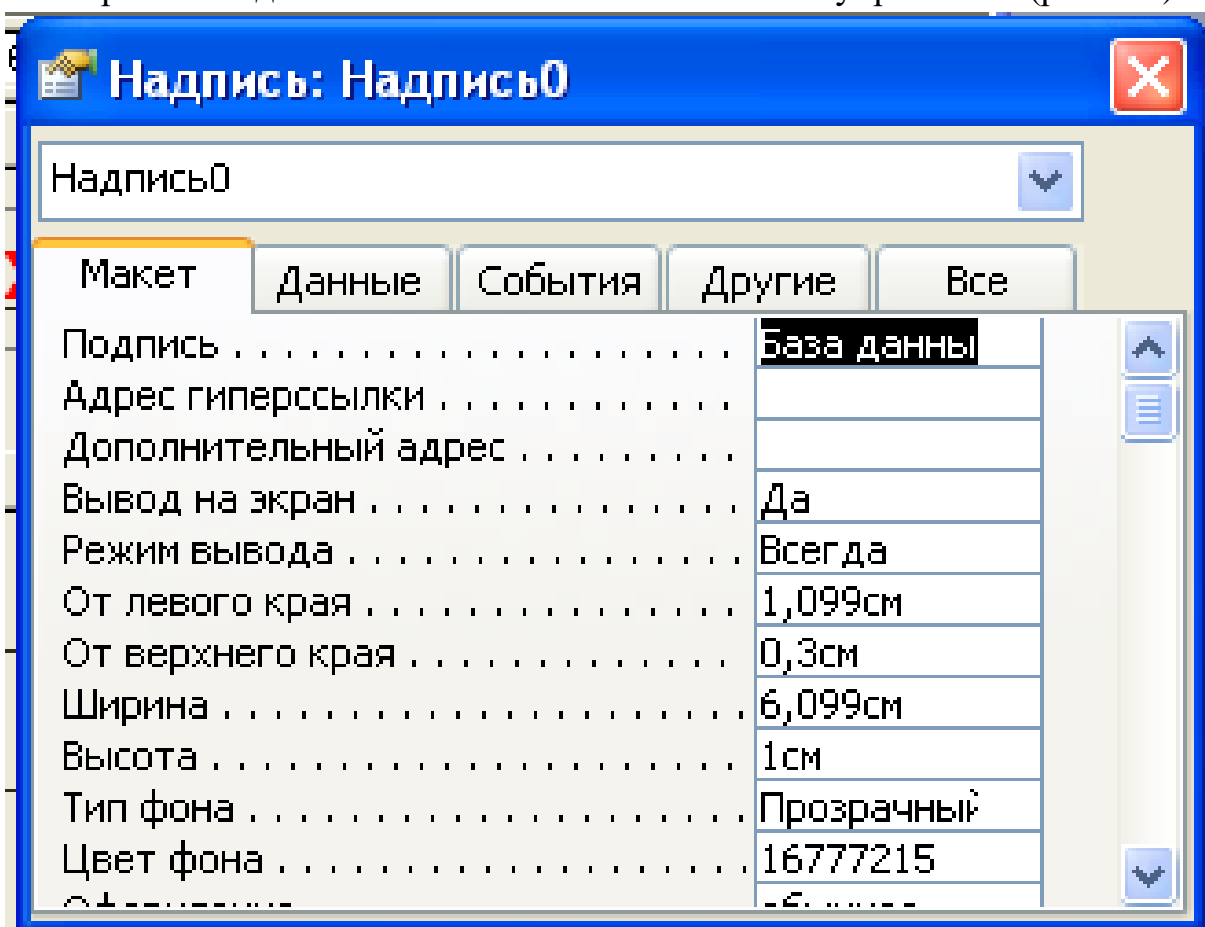



Рис. 19. Окно свойств полей данных

• Для того чтобы на форме «Заставка» убрать полосы прокрутки, выполните следующие действия: откройте форму в режиме **Конструктор**. Щелкните правой кнопкой мыши на форме и выберите «Свойства формы». В диалоговом окне «Окно свойств», во вкладке **Макет** выполнить следующие действия:

- Полосы прокрутки – выбрать **Отсутствуют**;
- Область выделения – выбрать **Нет**;
- Кнопки перехода (навигации) – **Нет**;
- Разделительные линии – **Нет**.

• Добавьте рисунок (логотип) на главную форму. Для этого щелкните на элементе **Рисунок** , щелкните в левой части заголовка и протащите указатель по форме, чтобы начертить прямоугольник. Отпустите кнопку мыши. Появится диалоговое окно **Выбор рисунка**, позволяющее выбрать графический файл, который будет вставлен в элемент управления.

• В режиме **Конструктора**, щелкнув правой кнопкой мыши по созданной кнопке **Список**, выберите **Свойства**. Открывается диалоговое окно **Кнопка: Кнопка0**. Во вкладке **Макет** добавьте фон или рисунок.

Сохраните изменения и переключитесь в режим формы, чтобы посмотреть на окончательный результат (рис. 20).

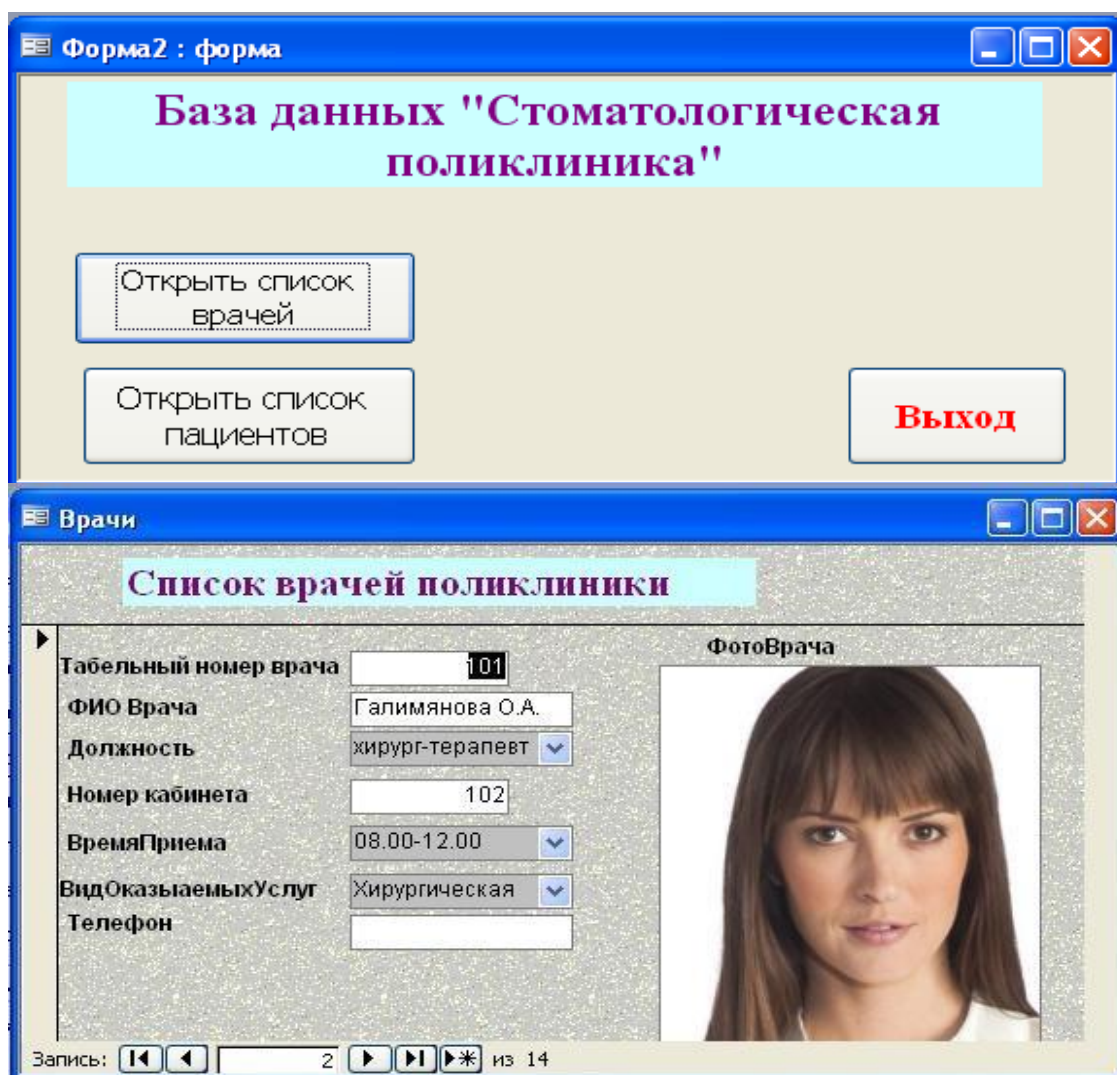


Рис. 20. Главная форма и форма «Врачи»

ЗАДАНИЕ 6. СОЗДАНИЕ СХЕМЫ ДАННЫХ

Одним из способов представления данных, независимо от реализующего его программного обеспечения, является модель «Сущность-связь».

Модель «Сущность-связь» (ER-модель) (англ. entity-relationship model, ERM) – модель данных, позволяющая описывать концептуальные схемы предметной области. С её помощью можно выделить ключевые сущности и обозначить связи, которые могут устанавливаться между этими сущностями.

Основными понятиями метода «Сущность-связь» являются: *сущность*, *атрибут сущности*, *ключ сущности*, *связь между сущностями*, *степень связи*.

Сущность представляет собой объект, информация о котором хранится в базе данных. Названия сущностей – существительные. Например: *ВРАЧИ*, *ПАЦИЕНТЫ*.

Атрибут представляет собой свойство сущности. Так, атрибутами сущности *ВРАЧИ* являются: *табельный номер*, *ФИО*, *должность* и т.д.

Ключ сущности – атрибут или набор атрибутов, используемый для идентификации экземпляра сущности.

Связь сущностей – зависимости между атрибутами этих сущностей. Название связи обычно представляется глаголом. Например: *врач ЗАНИМАЕТ должность*.

Важным свойством модели «Сущность-связь» является то, что она может быть представлена в графическом виде, что значительно облегчает анализ предметной области.

Будем использовать нотацию Чена-Мартина (табл. 4).

Таблица 4

Условные обозначения нотации Чена-Мартина

Обозначение	Описание
 Имя сущности	Набор независимых сущностей
 Имя сущности	Набор зависимых сущностей
 Имя атрибута	Значение атрибута
 Имя атрибута	Ключевой атрибут
 Имя связи	Набор связей

В качестве примера построим диаграмму, отражающую связь между пациентами и врачами, которые оказывают стоматологические услуги (рис. 21).

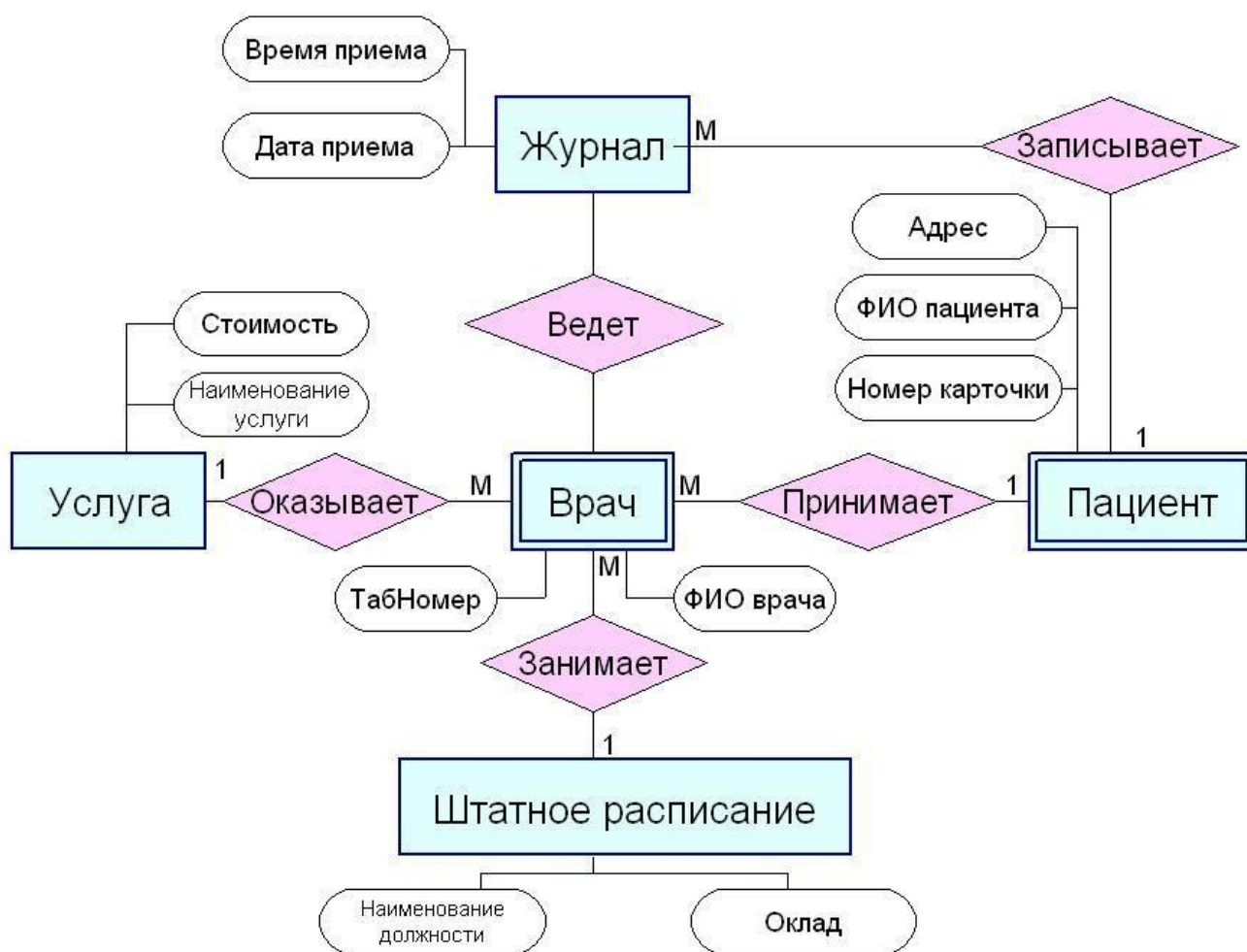


Рис.21. Пример ER-диаграммы («Сущность-связь»)

Во время проектирования баз данных происходит преобразование ER-модели в конкретную схему базы. Основными объектами схемы являются таблицы и связи. Рассмотрим создание схемы базы данных.

Этапы создания схемы данных.

1. Для создания схемы данных выполните действия: вкладка «Работа с базами данных» → «Схема данных».

2. Появится окно **Добавление таблицы**. Выберите вкладку Таблицы, выделите таблицы: «Врачи», «ШтатноеРасписание», «НаименованиеУслуг», «Журнал», «СписокПациентов» и разместите их с помощью кнопки <Добавить>. В окне **Схема данных** появится условный вид этих таблиц (рис. 22). Закройте окно.

Примечание. Жирным текстом в схеме будут выделены поля, имеющие ключевое значение.

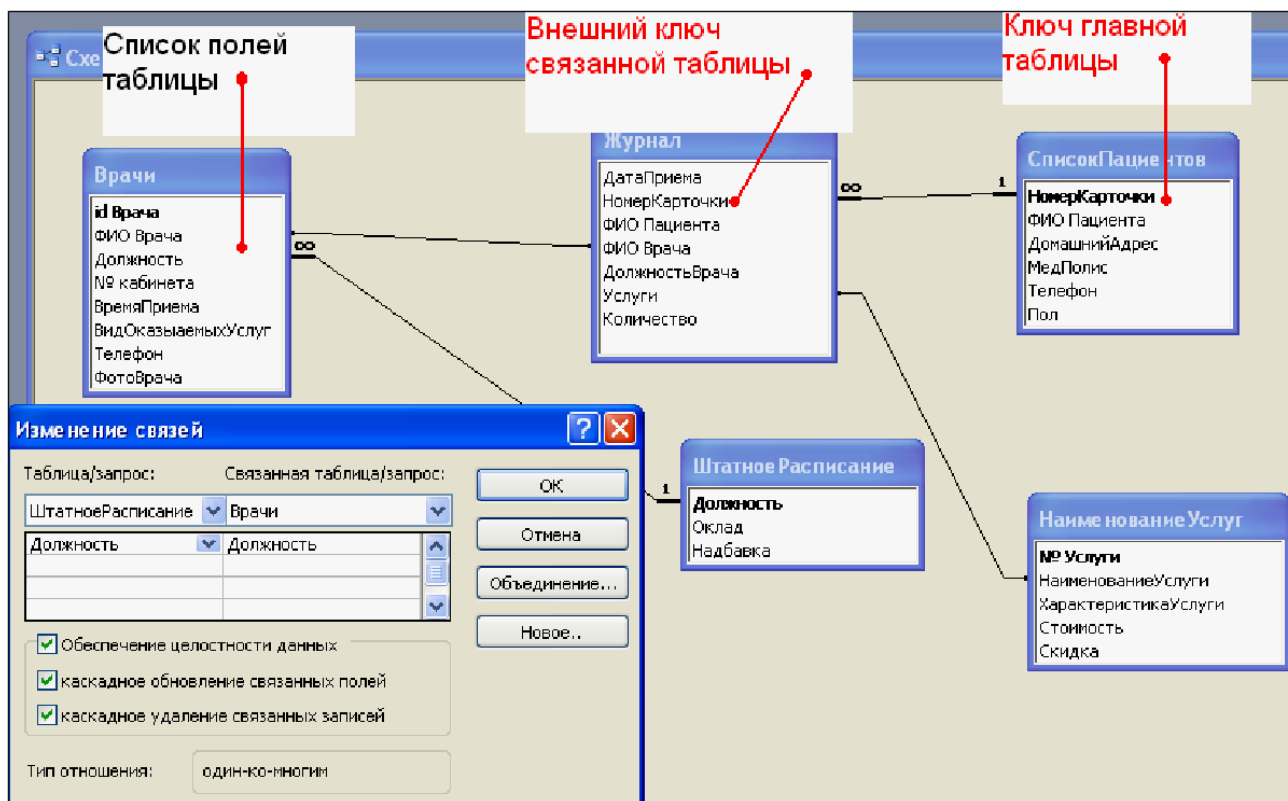


Рис. 22. Создание схемы данных

3. Установите связь между таблицами **СписокПациентов** и **Журнал**. Для этого установите курсор на имя поля «**НомерКарточки**» – в этой таблице это поле – ключевое (таблица *Список Пациентов*) и, не отпуская кнопку мышки, перетащите ее на поле «**НомерКарточки**» – подчиненной таблицы *Журнал*. Отпустите мышку. Появится диалоговое окно **Связи**.

4. Для автоматической поддержки целостности базы данных установите флажок: *Обеспечение целостности данных*. Кроме этого значка в этом окне представлены другие:

- *Каскадное обновление связанных полей*. При включении данного режима изменения, сделанные в связанном поле первой таблицы, автоматически вносятся в поля связанной таблицы, содержащей те же данные.
- *Каскадное удаление связанных записей*. При включении данного режима удаление записей в первой таблице приводит к удалению соответствующих записей связанной таблицы.

Примечание. Это возможно сделать только в том случае, если типы и размер полей заданы одинаково.

Щелкните по кнопке <Создать>. Появится связь **один-ко-многим**:

5. Прделайте те же шаги для поля **Должность**.

Внимание! Изменить связь и параметры объединения можно, щелкнув правой кнопкой мыши по *линии связи*. Откроется окно **Изменение связей**. Выбрав кнопку **Объединение**, можно выбрать другой способ, что позволяет перейти к созданию связи между любыми двумя таблицами базы данных (рис. 23).

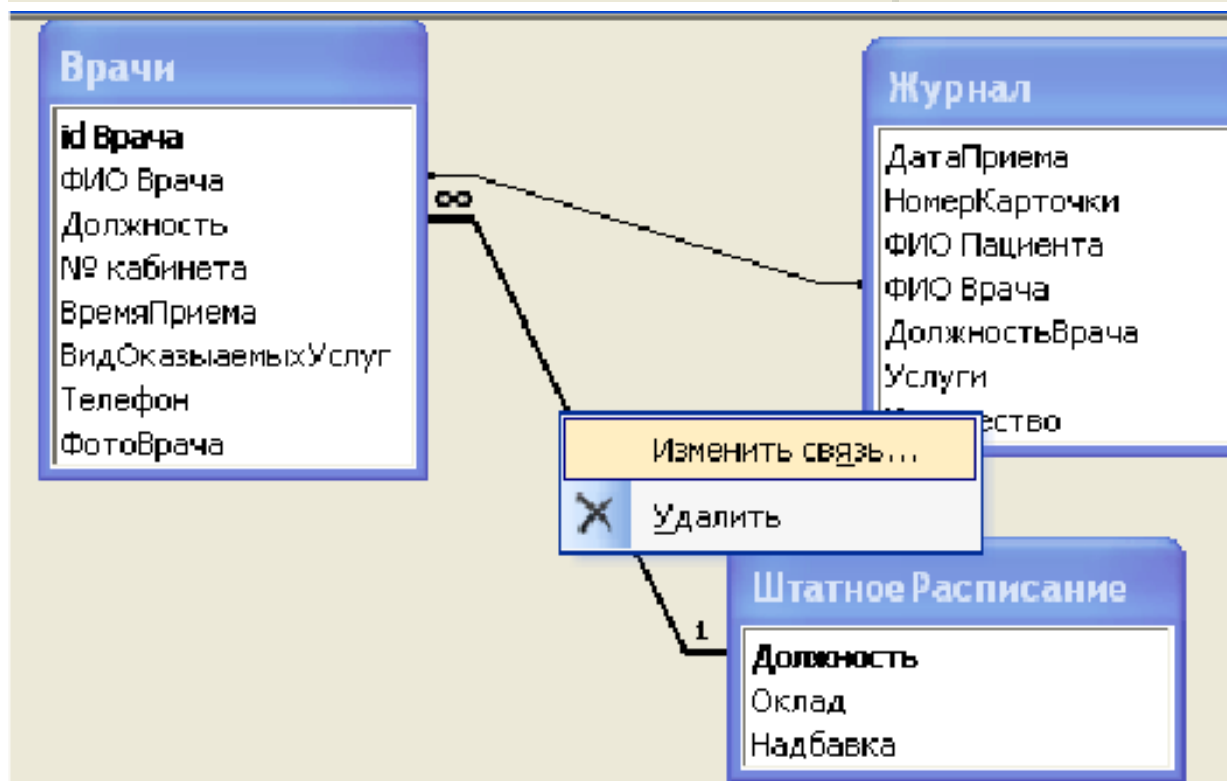
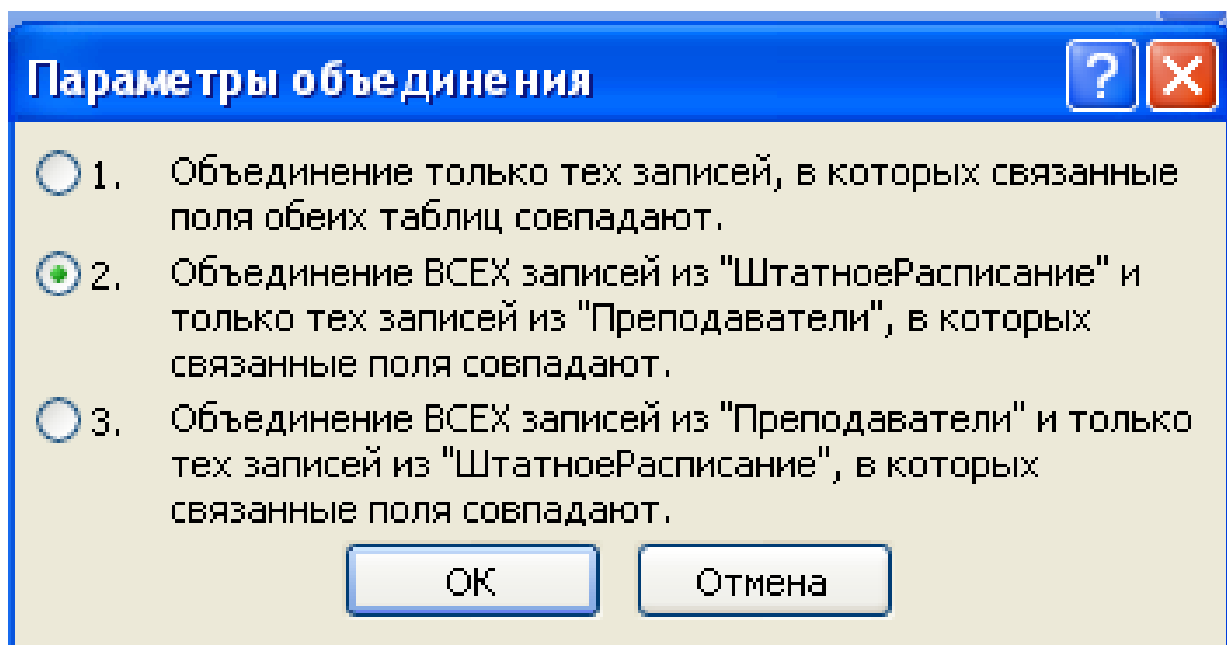


Рис. 23. Изменение связей (объединение)

ВЫПОЛНЕНИЕ САМОСТОЯТЕЛЬНОЙ РАБОТЫ № 1. (стр. 50).

ЗАДАНИЕ 7. АВТОМАТИЧЕСКИЙ ЗАПУСК ФОРМЫ ИЗ ФАЙЛА ПРИЛОЖЕНИЯ ACCESS

• Откройте форму «Главная» (или «Заставка») в режиме Конструктора. По команде Сервис Параметры запуска в диалоговом окне Параметры запуска в окне «Вывод формы/страницы» выберите форму, которая будет открываться при запуске приложения, в данном примере «Главная» (рис. 24).

• Можно задать заголовок приложения: Стоматология, который будет использован вместо заголовка MS Access.

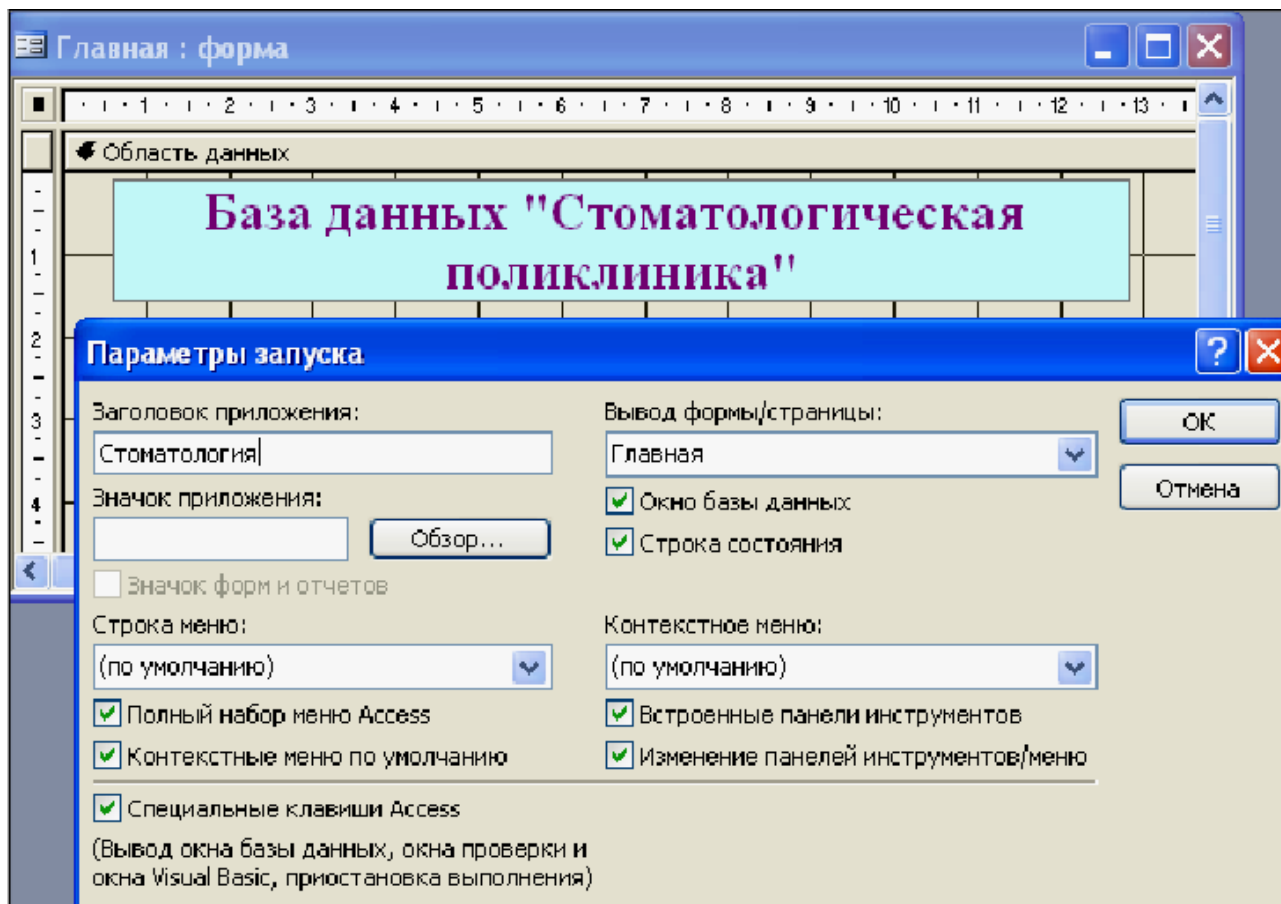


Рис. 24. Параметры запуска

ЗАДАНИЕ 8. СОЗДАНИЕ МНОГОТАБЛИЧНОЙ ФОРМЫ

1. Создание подчиненной формы.

Подчиненная форма создается только для связанных таблиц по ключевому полю.

Выберите объект **Формы**. Режим создания – **Мастер форм**. В качестве источника данных выберите из списка таблицу *ШтатноеРасписание*, нажмите <OK>.

В открывшемся в первый раз диалоговом окне **Создание форм** последовательно выбираются таблицы из списка и поля, необходимые для создания формы.

- В данном примере выбрать все поля таблицы *Штатное Расписание*: Должность, Оклад, Надбавка (рис. 25).

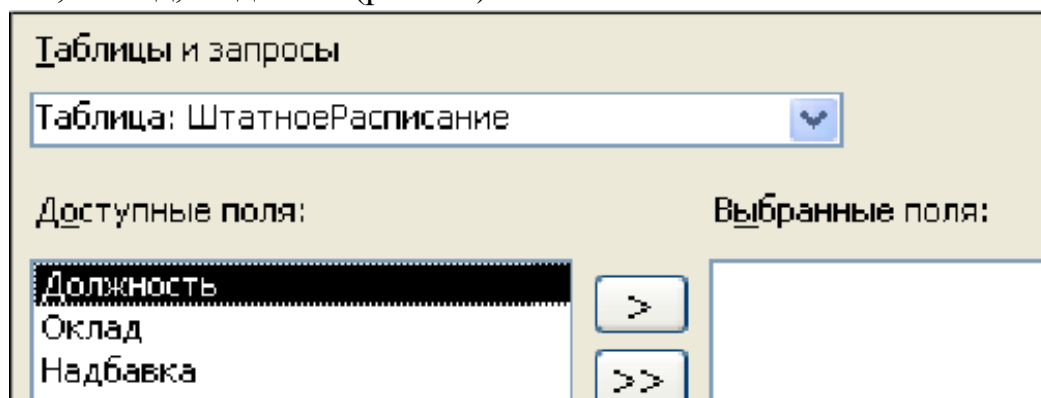


Рис.25. Выбор полей для формы посредством Мастера

- В этом же окне из выпадающего списка «Таблицы и запросы» выберите

таблицу *Врачи* и с помощью стрелки из **Доступных полей** выбрать необходимые поля таблицы (рис.26): «id_Врача», «ФИО_Врача», «Вид Оказываемых Услуг». Нажмите <Далее>.

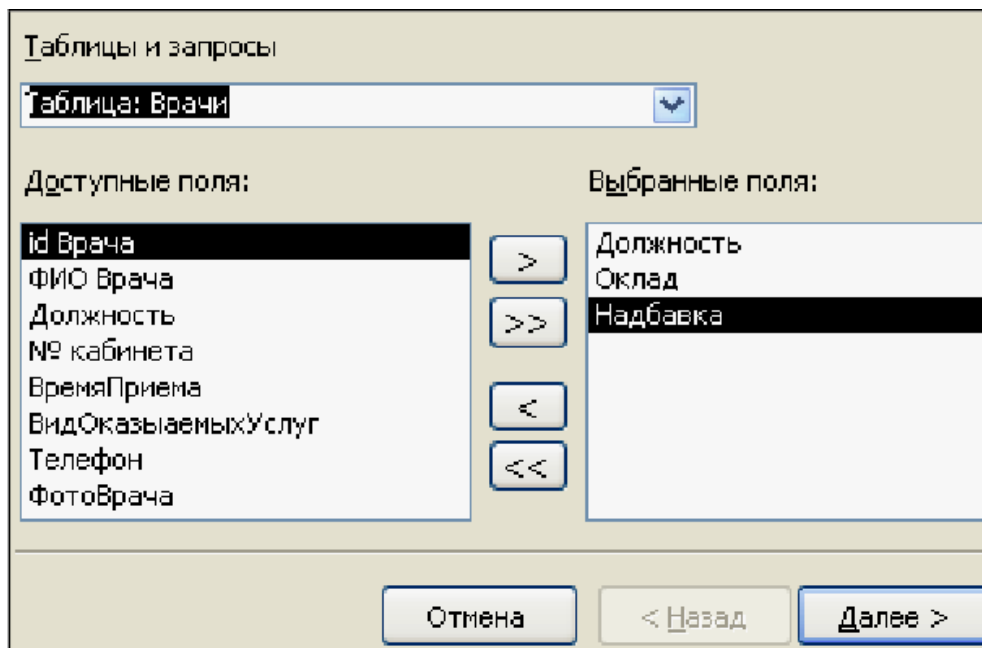


Рис.26. Выбор полей для подчиненный формы

- Таблица *ШтатноеРасписание* является главной по отношению к форме *Врачи*. В окне создания форм «Вид представления данных» выбираем для нашего примера таблицу *ШтатноеРасписание*.
- Выбираем переключатель **Подчиненные формы**. <Далее>.
- Выбрать внешний вид подчиненной формы – **Табличный**. <Далее>.
- В следующем окне можно изменить названия главной и подчиненной ей формы. Далее выбрать дизайн оформления. <Готово>.
- Готовая подчиненная форма имеет вид (см. рис. 27).

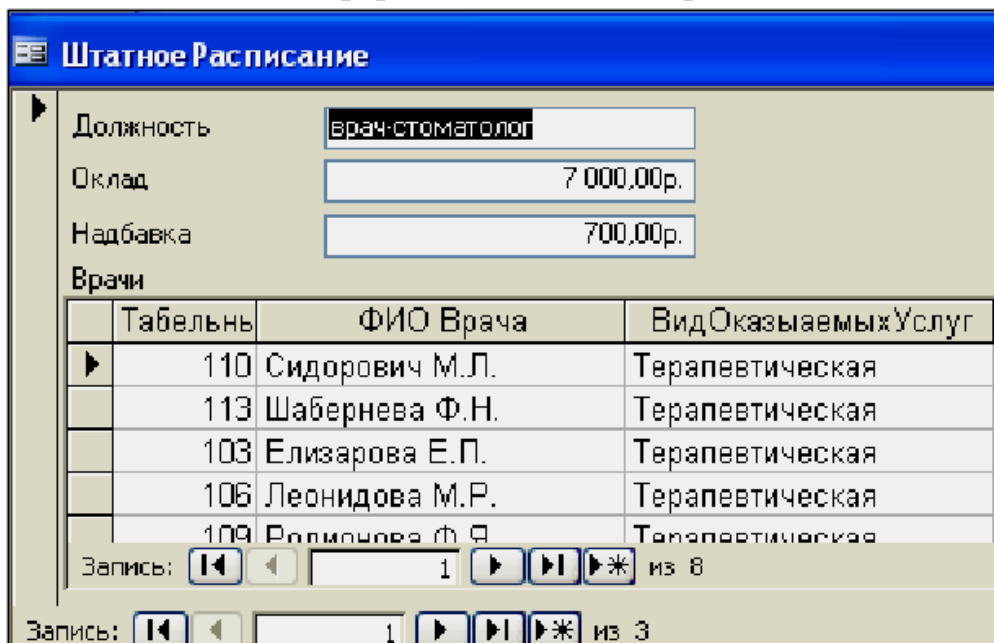


Рис. 27. Подчиненная форма

2. Создание связанной формы.

• Для создания связанной формы в качестве таблицы-источника выбрать таблицу *Список Пациентов*: все поля, в качестве связанной таблицу *Журнал*: выбрать поля *ВремяПриема*, *ФИО врача*, *ДолжностьВрача*, *Услуги*, *Количество*. Повторяющиеся поля не рекомендуется включать. В окне создание форм выберите переключатель **Связанные формы**. <Готово> (рис. 28).

• Измените дизайн форм, увеличив шрифт на *11pt*.

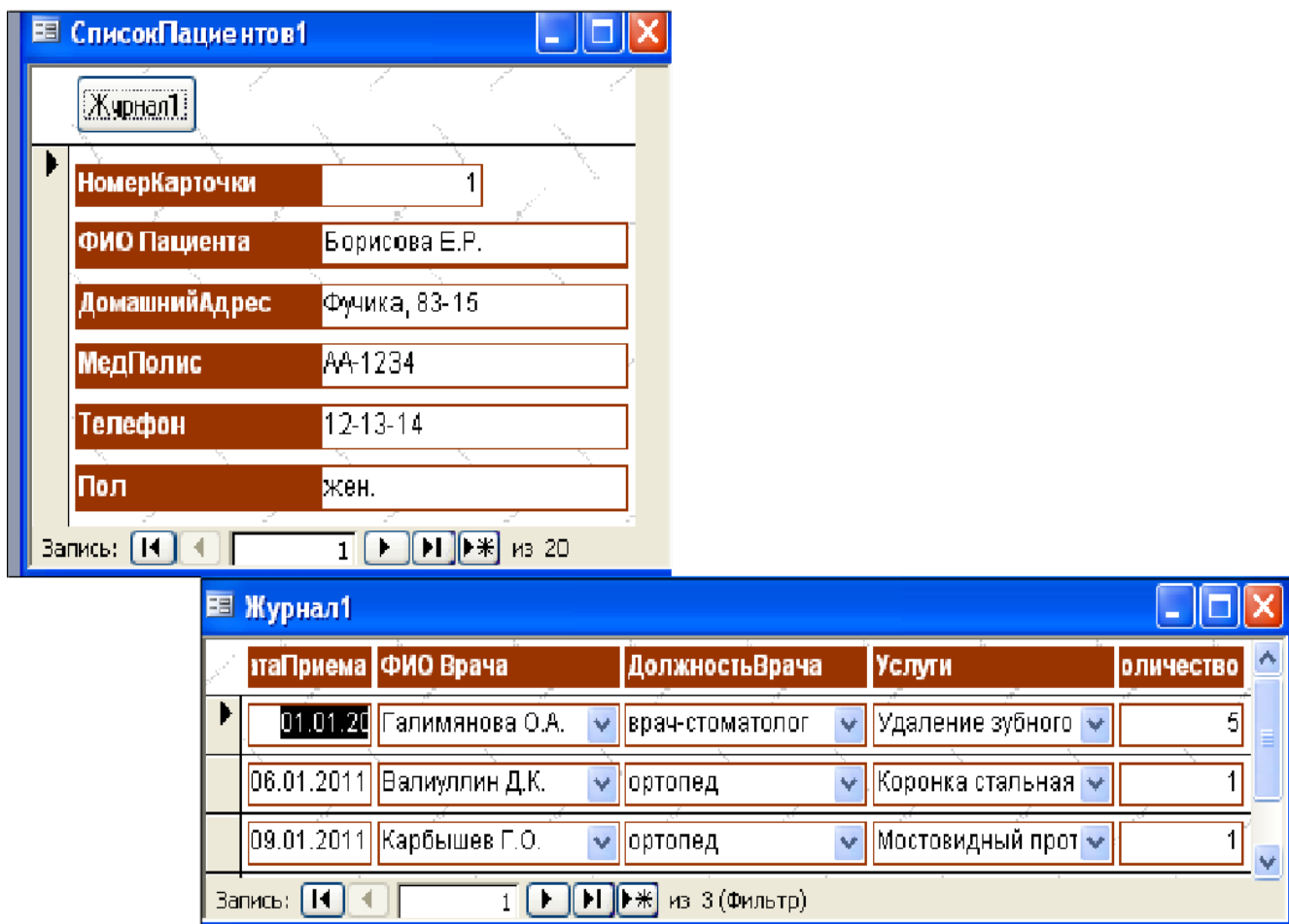

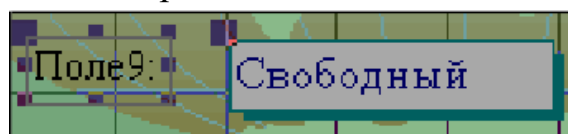


Рис.28. Готовая связанная форма

ЗАДАНИЕ 9. СОЗДАНИЕ ВЫЧИСЛЯЕМЫХ ПОЛЕЙ В ФОРМЕ

• Откройте подчиненную форму *Штатное Расписание* в режиме **Конструктора**.

- Активируйте кнопку **Панель элементов** .
- Выберите элемент **Поле** и перенесите его на свободное место формы:



- Переименуйте <Поле 9> в <Всего зарплата>.
- Щелкнув правой кнопкой мыши по окошку с надписью *Свободный*, вы-

берите **Свойства**. В открывшемся диалоговом окне *Поле: <Поле 9>* во вкладке **Данные** в строке **Данные** открыть **Построитель выражений** (рис. 29):

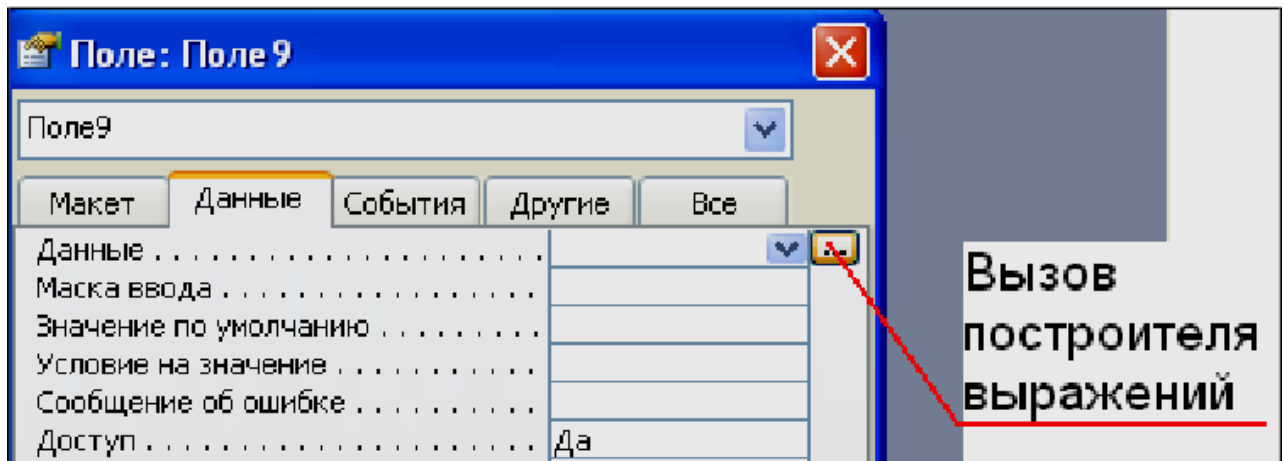


Рис. 29. Вызов построителя выражений

• В окне **Построитель выражений** (рис. 30) должно быть записано следующее выражение `=[Оклад]+[Надбавка]`. Следовательно, в новом окне поля будет подсчитана **ВСЕГО** зарплата с учетом оклада и надбавки. Для этого выберите таблицу **ШтатноеРасписание**. В следующем окне **Построителя** выбираем поля, которые участвуют в вычислении: *Оклад* и *Надбавка*. <ОК>.

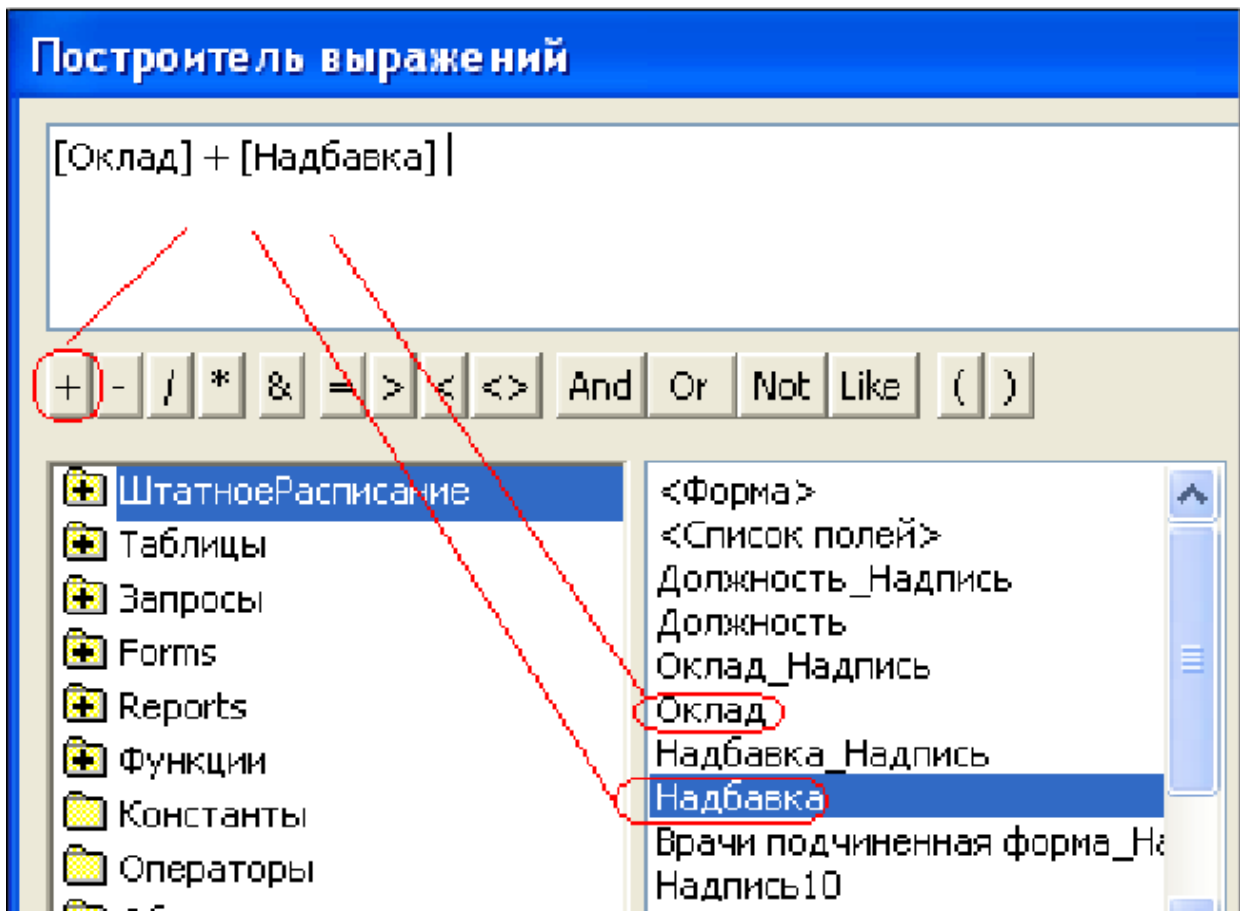


Рис. 30 Построитель выражений

- Измените шрифт полей и окон формы на 12, зайдя в свойства поля, вкладка **Макет**. Измените надпись: «Поле9» на «Зарплата всего».

ЗАДАНИЕ 10. СОЗДАНИЕ ДРУГИХ КНОПОК НА ФОРМЕ

- Откройте форму *Врачи* в режиме **Конструктора**. Выберите «Кнопку» на **Панели элементов**. Перенесите ее на форму.

- Для создания кнопки перехода по записям используем мастер кнопок. В открывшемся диалоговом окне **Создание кнопок** выберите «Категории»: *Переход по записям*, и «Действия»: *Следующая запись*. В качестве указателя выберите стрелку.

- Установите свойства формы: «Кнопки перехода» – выберите «Нет».

- Самостоятельно создайте кнопку: *Предыдущая запись*. В результате форма может иметь окончательный вид (рис. 31).

Рис. 31 Окончательная форма

Запросы. Создание таблиц – не единственная задача БД. Нам может понадобиться сравнить таблицу за прошлый год с таблицей за этот год, обновить данные или, например, подсчитать налог с продаж.

Запрос – это программа, которая задает вопрос, подходящий к табличным данным. Они предназначены для поиска в БД информации, отвечающей определенным критериям. В сущности, запрос представляет собой вопрос, сформулированный в терминах Базы данных.

Запрос состоит на основе одной или нескольких взаимосвязанных таблиц, позволяя комбинировать содержащую в них информацию. При этом могут использоваться как таблицы БД, так и сохраненные таблицы, полученные в результате выполнения запроса.

В Access можно создавать следующие типы запросов:

- запрос на выборку;
- запрос с параметрами;
- перекрестный запрос;
- запрос на изменение (запрос на удаление, обновление и добавление за-

писей на создание таблицы);

- запросы SQL (запросы на объединение, запросы к серверу, управляющие запросы, подчиненные запросы).

Запрос на выборку используется наиболее часто. При его выполнении данные, удовлетворяющие условиям отбора, выбираются из одной или из нескольких таблиц и выводятся в определенном порядке.

Примечание. Простые запросы на выборку практически не отличаются от фильтров. Более того, фильтры можно сохранять как запросы. Можно также использовать запрос на выборку, чтобы сгруппировать записи для вычисления сумм, средних значений, пересчета и других действий.

Запрос с параметрами – это запрос, при выполнении которого в диалоговом окне пользователю выдается приглашение ввести данные, на основе которых будет выполняться запрос.

Перекрестные запросы – это запросы, в которых происходит статистическая обработка данных, результаты которой выводятся в виде таблицы.

Окно запроса. Для создания простых запросов используется **Мастер запросов** или **Конструктор**. Для этого выбираем вкладку **Запросы** **Создание запроса в режиме конструктора (или мастера)**. Открывается поле *Запрос на выборку* (рис. 32).

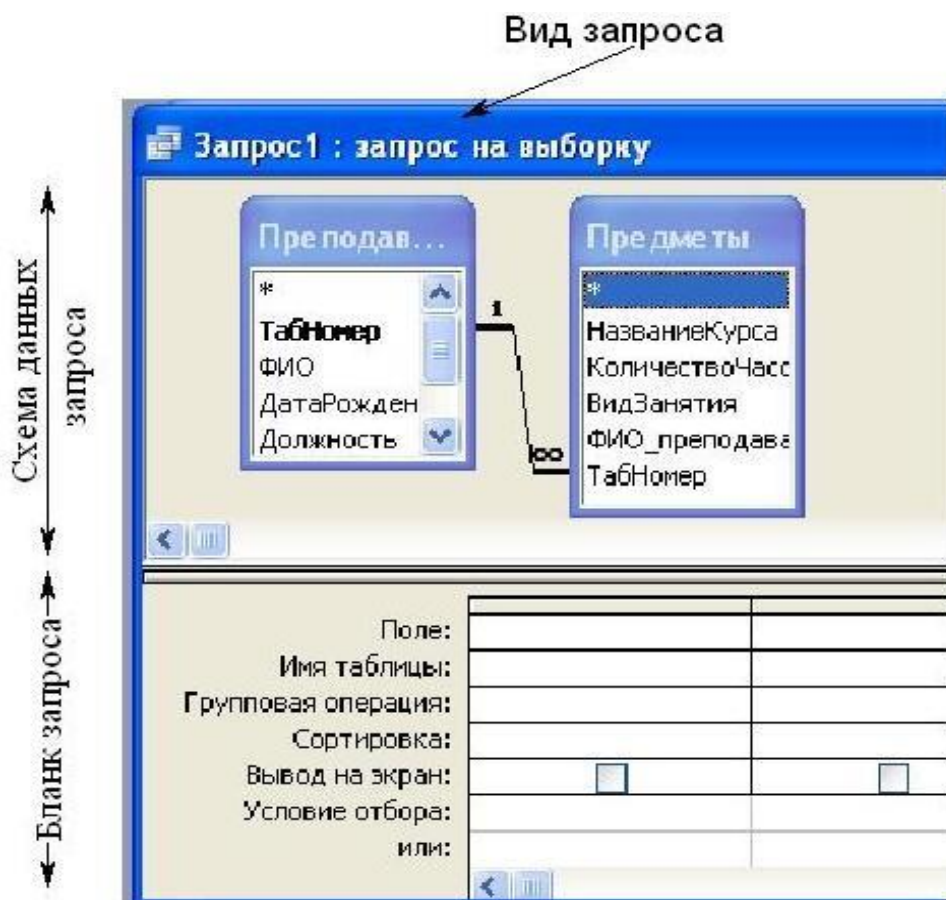


Рис. 32. Вид запроса на выборку

Первая строка бланка -запроса – **Поле** содержит имена полей, включенных в запрос.

Во второй строке – **Имя таблицы** указана таблица, к которой принадлежит каждое поле.

Третья строка – **Групповая операция** (*Total*) позволяет выполнять вычисления над значениями полей.

Четвертая строка – **Сортировка** указывает принцип сортировки записей. Флажок в пятой строке – **Вывод на экран** (*Show*) определяет, будет ли поле отображаться в результате запроса.

Шестая строка с именем **Условие отбора** (*Criteria*) позволяет задать критерий отбора записей из таблицы.

Седьмая строка – **Или** задает альтернативный критерий.

Добавление таблицы в схему данных осуществляется с помощью команды: **Запрос** → **Отобразить таблицу** или нажатием правой кнопки мыши на поле схемы данных запроса командой **Добавить таблицу**.

Для создания нового запроса надо в окне базы данных выбрать вкладку *Запросы* и щелкнуть по кнопке <Создать>. Откроется окно «Новый запрос».

В окне можно выбрать один из пяти пунктов: **Конструктор, Простой запрос, Перекрестный запрос, Повторяющиеся записи, Записи без подчиненных**.

ВЫПОЛНЕНИЕ САМОСТОЯТЕЛЬНОЙ РАБОТЫ № 2. (стр. 51).

ЗАДАНИЕ 11. ФОРМИРОВАНИЕ ЗАПРОСОВ НА ВЫБОРКУ

1. На основе таблицы *Врачи* создать простой запрос на выборку, в котором должны отображаться **Табельный номер, ФИО врачей** и их **должность**.


2. Использовать фильтр для отображения некоторых данных, например все **врачи-стоматологи**.

3. Создать запрос на выборку с параметром, в котором должны отображаться **ФИО врачей, время приема** и **номера кабинетов**.

4. Создать запрос с помощью построителя выражений.

Этапы формирования запросов.

1. Для создания **простого** запроса:


- в окне базы данных откройте вкладку *Запросы*;
- в открывшемся окне щелкните по кнопке <Создать>;
- из появившихся пунктов окна «Новый запрос» выберите **Простой запрос** и щелкните по кнопке <ОК>;
- в появившемся окне в строке *Таблицы/запросы* выберите таблицу *Врачи*;
- в окне «Доступные поля» переведите выделение на параметр *ID Врача*, щелкните по кнопке . Поле *ID Врача* перейдет в окно «Выбранные поля»;
- аналогично в окно **Выбранные поля** переведите поля: «ФИО врача», «Должность» (в таком порядке данные и будут выводиться);

- щелкните по кнопке. <Далее>;
- в строке параметра «Сохранение запроса» введите новое имя «Должности врачей». Нажмите <Готово>.

2. Для выборки данных: всех врачей, занимающих должность *Врач-стоматолог*, выполните следующие действия:

- в режиме запроса поставьте курсор в строку с записью «Врач-стоматолог»;
- на панели инструментов выполните следующую команду **Записи** → **Фильтр** → **Фильтр по выделенному**;
- в таблице отобразятся только данные врачей, занимающих должность *стоматолога*;




- для отмены фильтра нажмите кнопку .

3. Для создания запроса на выборку с параметром:


- создайте запрос на выборку для следующих полей таблицы «Врачи»: *ID Врача, ФИО врача, № кабинета, Время Приема*; таблицы «Смена_подстановка»: *НомерСмены*. Запрос создавайте аналогично тому, как это делалось в п.1; задайте имя запросу «*Время приема*»;



- щелкните по кнопке  на экране появится таблица с результатами запроса;

- перейдите в режиме конструктора запроса;
- в строке параметра *Условия отбора* для поля «*НомерСмены*» введите фразу в квадратных скобках: [*Введите номер смены*];



- выполните запрос .
- в появившемся окне введите «1» и щелкните по кнопке <ОК>. На экране появится таблица с данными всех врачей, работающих в первую смену;
- в режиме **Конструктор** уберите галочку в строке вывода на экран (чтобы в результирующей таблице не выводиться повторяющихся данных с номером смены в каждой строке);
- сохраните запрос и закройте окно.

4. Создайте запрос с помощью **Конструктора**.

Во вкладке запроса выполните команду: **Создать** → **Конструктор**.

В появившемся окне «Запрос1: Запрос на выборку» выберите таблицы *Врачи* и *НаименованиеУслуг*. Кнопка <Добавить>.

Двойным щелчком мыши выберите поля:

- из таблицы «**Врачи**»: *ID Врача, ФИО Врача*;
- из таблицы «**НаименованиеУслуг**»: *НаименованиеУслуги*.

В столбце: *ФИО Врача* в строке: **Условие отбора** ввести: *М**.

Щелкните по кнопке . Посмотрите результат запроса.

В результате мы получим информацию о врачах и услугах, фамилии которых начинается на букву **М**.

Закройте запрос и сохраните под названием «Врачи-выборка».

5. Создание запроса с помощью построителя выражения.

Создайте запрос: расчет зарплаты, добавив таблицы:

«Врачи»: поля - *ID Врача, ФИО Врача, Должность*;
«Штатное расписание»: поля – *Зарплата, Надбавка*.

В свободной столбце в строке **Поле:** нажмите правой кнопкой мыши, выберите Построить...

В появившемся диалоговом окне **Построитель выражения:**

- выберите из списка таблицу «Штатное расписание»;
- из открывшегося списка полей щелкните по полю [Оклад], затем знак «+» и поле [Надбавка];
- нажмите <ОК> (см. рис. 33).

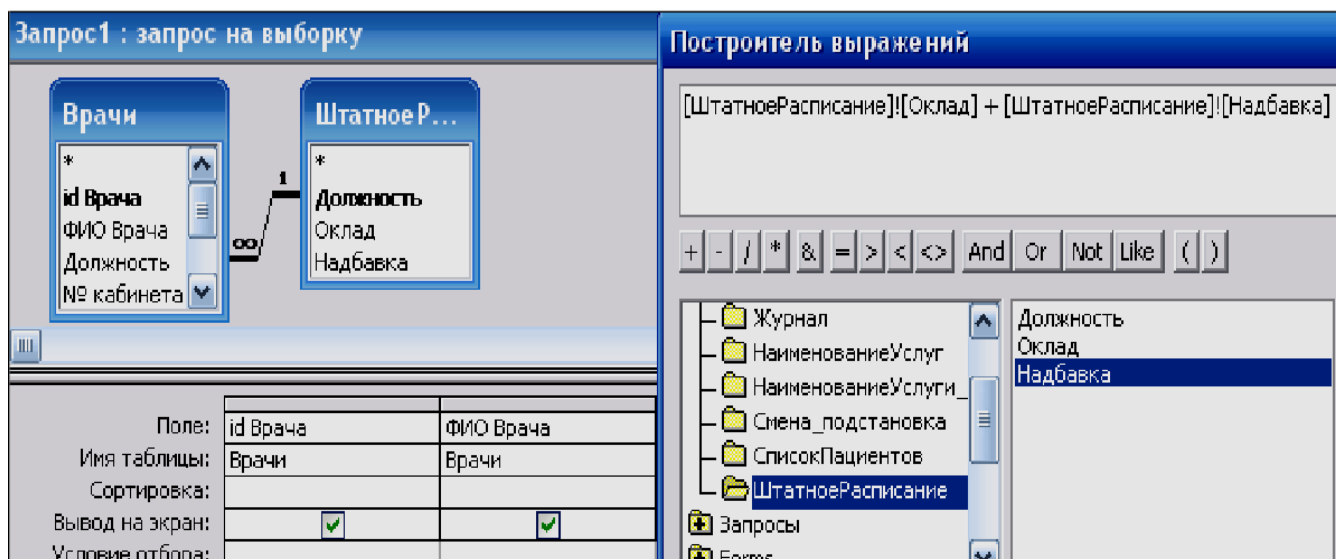


Рис. 33. Создание запроса с вычисляемым полем

– щелкните по кнопке

– сохраните запрос «Итого зарплата».

– в режиме Конструктора в строке **Поле:** выделите *Выражение1:* и переименуйте на «Итого:».

6. Создайте запрос: выборка данных за определенный период, добавив таблицу: «Журнал», поля: *ДатаПриема, ФИО Врача, Услуга, Количество, ФИО Пациента*.


В свободном столбце в строке **Поле:** нажмите правой кнопкой мыши и выберите Построить...

В появившемся диалоговом окне **Построитель выражения** выберите: **Функции** → **Встроенные функции** → **Дата/время**. Щелкните два раза кноп-

кой мыши <Month>. Выделите «number». Далее выберите таблицу «Журнал» из списка окошка **Построителя** для выборки поля *ДатаПриема*, щелкнув по раскрывающемуся списку: **Таблицы** → **Журнал**, **ДатаПриема**.

В поле построителя должно появиться следующее выражение:

Month([Журнал]![ДатаПриема])

Нажмите <ОК> (рис. 34). В этом же **столбце** в строке **Условие отбора** введите: [Введите номер месяца]. Нажмите кнопку .

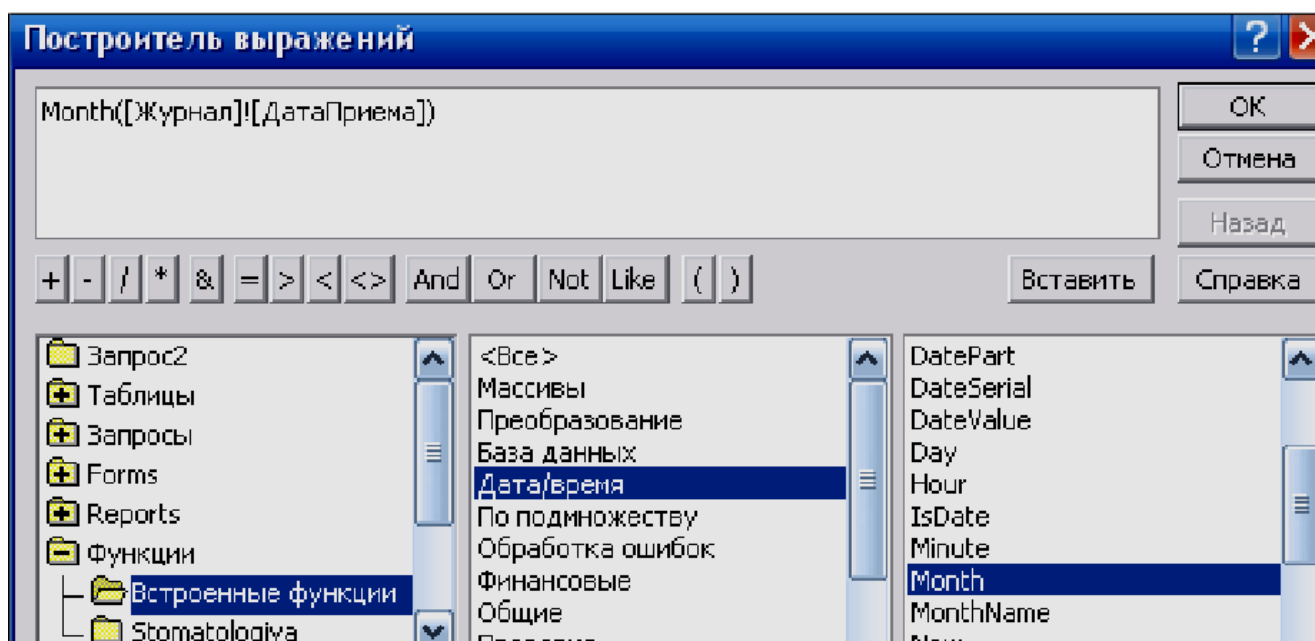


Рис. 34. Создание запроса


Для удаления повторяющихся записей в режиме **Конструктора** уберите галочку в столбце, где записано выражение.

САМОСТОЯТЕЛЬНО: создайте запрос на выборку фамилий пациентов, рожденных после 1990 года:

Для этого добавьте поле *ДатаРождения* в таблицу «*Пациенты*» в формате **Краткий формат даты**:

Поле *Year([Пациенты]![ДатаРождения])*
Условие отбора [Год рождения]

7. Используя таблицу *Журнал* подсчитайте *Общее количество* выполненных услуг каждого врача.

Для этого выберите поля *ФИО врача* и *Количество*. На панели инструментов нажмите  – **Групповые операции**. В столбце *Количество* в стро-

ке «Групповая операция» выбрать из выпадающего списка **Sum** (таблица 5). Просмотреть запрос и проверить правильность, используя таблицу *Журнал* и фильтр записей.

Таблица 5

Назначение агрегатных функций

Функция	Назначение
Sum	Вычисляет сумму значений, содержащихся в поле
Avg	Вычисляет среднее арифметическое для всех значений поля
Count	Определяет число значений поля, не считая пустых (Null) значений
Min	Находит наименьшее значение поля
Max	Находит наибольшее значение поля
StDev	Определяет среднеквадратичное отклонение от среднего значения
Var	Вычисляет дисперсию значений поля

Примечания.

- Функция *Scir*, используемая в выражении, преобразует результаты вычислений в денежный формат.
- Функция *Count* вычисляет число записей, содержащих значение в поле.

8. Создайте запрос на подсчет количества врачей по должностям (рис. 35):

- для этого в режиме **Конструктора** выберите таблицу *Врачи* (поле: *Должность* добавьте два раза). Выберите групповую операцию;
- во втором столбце в строке «Групповая операция» из выпадающего списка выберите функцию **Count**.

Запрос4 : запрос на выборку	
Должность	Count-Должност
врач-стоматолог	8
ортопед	3
хирург-терапевт	3

Рис. 35. Результат запроса

ВЫПОЛНЕНИЕ САМОСТОЯТЕЛЬНОЙ РАБОТЫ № 3. (стр. 51).

ЗАДАНИЕ 12. ФОРМИРОВАНИЕ ЗАПРОСОВ НА ОБНОВЛЕНИЕ И УДАЛЕНИЕ

1. Сформируем *Запрос на обновление*.

Создайте таблицы: *Товар* и *Отгрузка* (рис. 36).

Имя поля	Тип данных
Код товара	Числовой
Наименование	Текстовый
Цена	Денежный

Таблица <ТОВАР>

Имя поля	Тип данных
НомерНакладной	Числовой
Код товара	Числовой
Отгружено	Числовой
ЦенаЗаЕд	Денежный
Стоимость	Денежный

Таблица <ОТГРУЗКА>

Рис. 36. Вид таблиц *Товар* и *Отгрузка*.

Заполните поля таблиц. Поля *ЦенаЗаЕд* и *Стоимость* оставьте незаполненными! Они будут заполнены в результате выполнения **запроса на обновление**, который данные из таблицы «Товар» перенесет в таблицу «Отгрузка».

Запрос 1.

Создайте запрос на обновление. Для этого:

- добавьте обе таблицы в **запрос**;
- выберите поле *ЦенаЗаЕд* (табл. «Отгрузка»). Щелкните на панели инструментов кноп-



- в строке «Обновление» щелкните правой кнопкой «**Построить**»;
- в открывшемся диалоговом окне Построителя выражений выберите команду, используя таблицу «Товар»: **[Товар]![Цена]** (рис. 37).

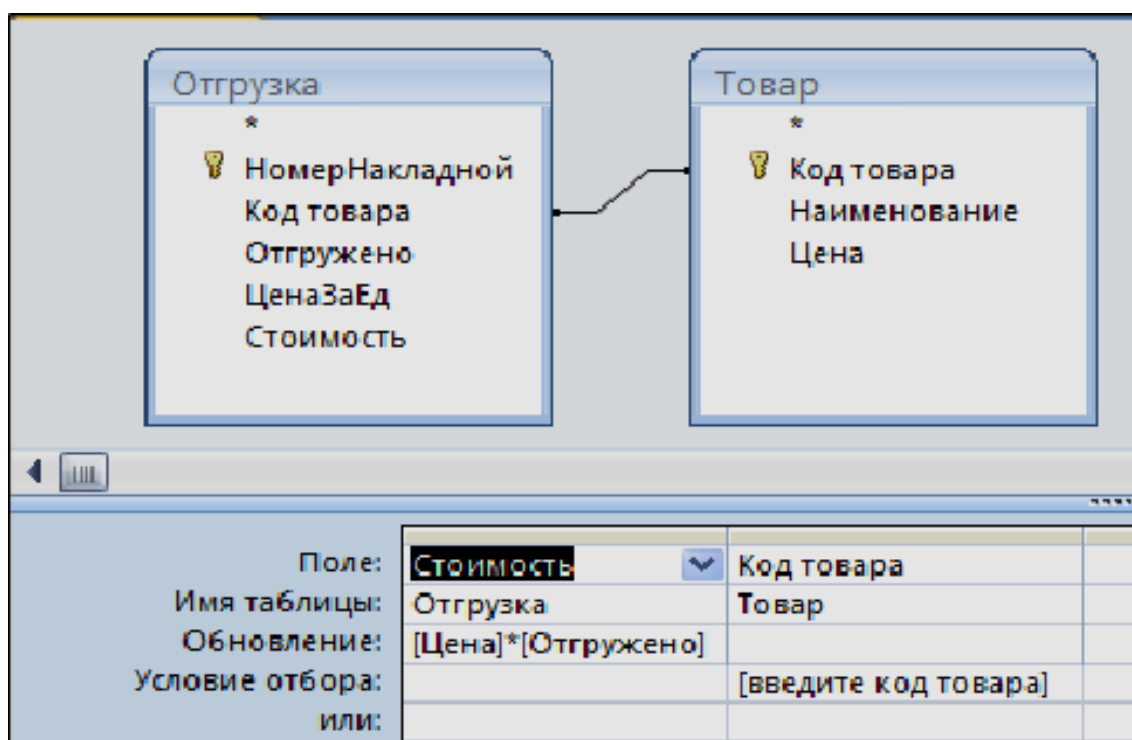


Рис. 37. Запрос 1 на обновление.

В результате запроса данные поля **Цена** (таблица «**Товар**») отобразятся в поле **ЦенаЗаЕд** (таблица «**Отгрузка**»).

Запрос 2.

Создайте запрос на обновление поля **Стоимость**, данные для которого являются результатом **Цена*Количество** товара (рис. 38):

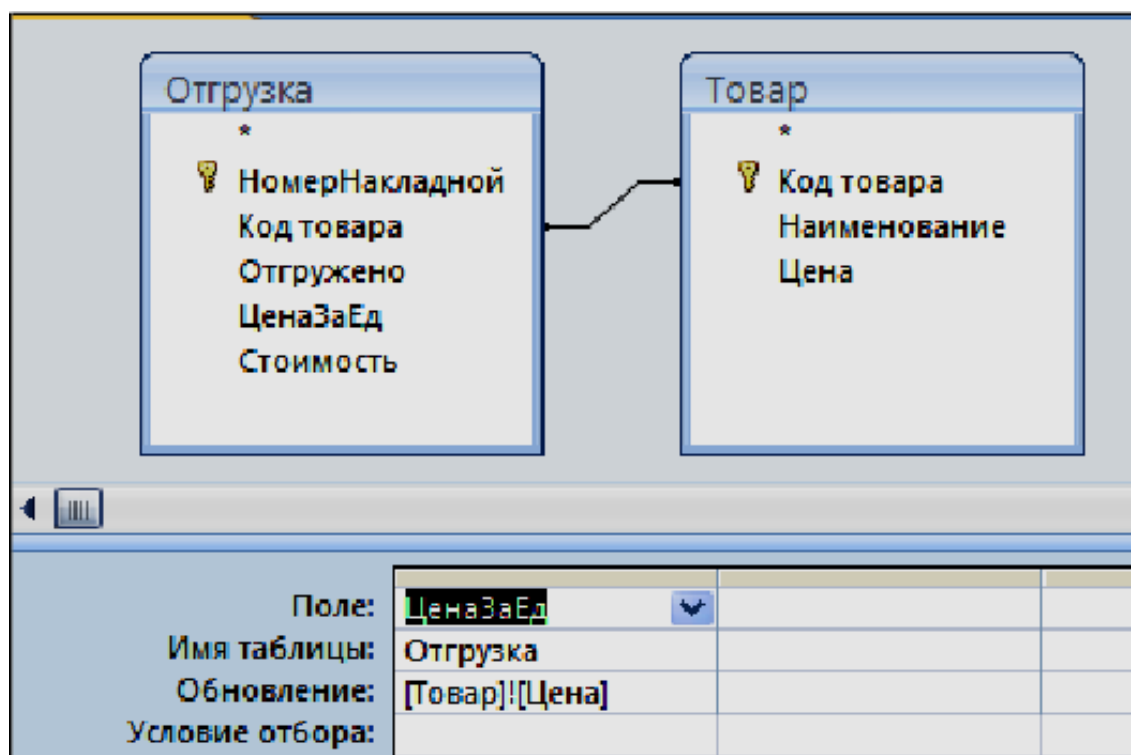



Рис. 38. Окно создания запроса на обновление

2. Сформулируем *Запрос на удаление*.

Создать запрос на удаление записей из таблицы *Отгрузка*: количество отгруженного товара больше 5. Для этого:

- добавьте таблицу *Отгружено* в запрос, выберите все поля, щелкнув по * («*» добавляет в запрос все поля таблицы);
- добавьте поле **Отгружено**;
- щелкните на панели инструментов по кнопке  или выполните команду «**Запрос Удаление**».

В результате запроса из таблицы *Отгружено* удалятся записи, соответствующие заданному условию.

САМОСТОЯТЕЛЬНО: создайте запросы на обновление записей в таблице «Журнал», для этого:

- В режиме **Конструктора** добавьте в таблицу «Журнал» поле «*СтоимостьУслуг*», тип данных – Денежный.
- Создайте запрос на обновление записей из таблицы «НаименованиеУслуг».
- В режиме **Конструктора** в таблице «Журнал» поменяйте название поля «*СтоимостьУслуг*», зайдя во вкладку **Общие** → **Подпись** → ввести «*СтоимостьОднойУслуги*».
- Создайте запрос на обновление (добавление) записей в поле *ФИО Пациента* и *ФИО Врача* в таблице «Журнал», используя таблицы «**СписокПациентов**» и «**Врачи**».

ЗАДАНИЕ 13. СОЗДАНИЕ ПЕРЕКРЕСТНОГО ЗАПРОСА

Перекрестные запросы служат для более компактного отображения информации. В перекрестном запросе отображаются результаты статистических расчетов (такие, как суммы, количество записей, средние значения), выполненных по данным из одного поля. Эти результаты группируются по двум наборам данных в формате перекрестной таблицы. Первый набор выводится в левом столбце и образует заголовки строк, а второй – выводится в верхней строке и образует заголовки столбцов.

Создадим перекрестный запрос, в котором в строках выводятся *фамилии врачей*, в столбцах – *дни приема*, в результирующих ячейках подсчитывается количество приемов в каждый из дней.

- В режиме **Конструктора** создайте новый запрос, добавьте таблицу «Журнал».

- Перетащите поля «*Дата приема*», «*ФИО Врача*», «*ФИО Пациента*» в первые три столбца бланка.

- Выберите команду меню **Запрос** → **Перекрестный**. Заголовок окна запроса *Запрос1: на выборку* изменится на *Запрос!перекрестный запрос*. Кроме того, в бланк запроса будут добавлены строки *Перекрестная таблица* и *Групповая операция*, в которую во всех столбцах автоматически вводится операция **Группировка**.

- Выберите в списке ячейки <*Перекрестная таблица*> столбца «*Дата приема*» значение **Заголовки столбцов**. Для столбца «*ФИО Врача*» – **Заголов-**

ки строк.

• Выберите в списке ячейки *Групповая операция* столбца «*ФИО Пациента*» значение *Count*. В ячейке *Условие отбора – Значение* (рис. 39).

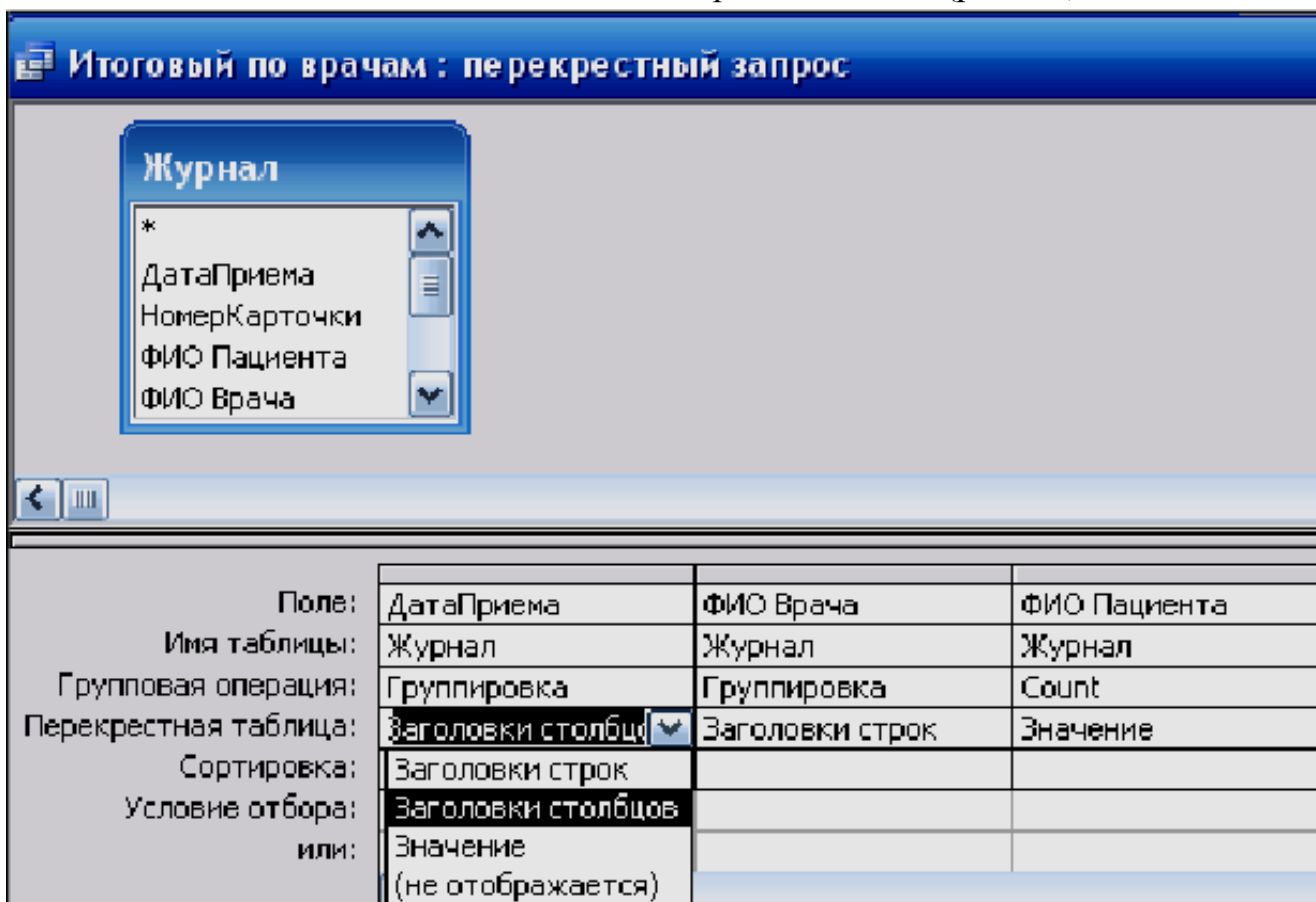


Рис. 39. Перекрестный запрос в режиме *Конструктора*

ЗАДАНИЕ 14. СОЗДАНИЕ СВОДНЫХ ТАБЛИЦ И ДИАГРАММ

Сводные таблицы и диаграммы предназначены для интерактивного анализа данных различной сложности и объема. Они позволяют произвести сравнительный анализ данных простыми и удобными средствами.

Сводные таблицы являются многомерными, и это позволяет представлять данные, сгруппированные по различным координатам.

1. Разработка сводной таблицы для таблицы базы данных.

Рассмотрим таблицу «**Журнал**» в режиме сводной таблицы. Чтобы открыть таблицу в этом режиме, выберите её в окне базы данных, нажмите кнопку «**Открыть**», а затем для перехода из режима таблицы в режим сводной таблицы выберите на панели **Вид** → **Сводная таблица**.

При первом открытии в ее рабочую область не включено не одного поля таблицы.

Пусть необходимо подсчитать количество выполненных услуг каждым

Журнал : таблица										
ДолжностьВрача ▾										
Все										
ФИО Врача ▾	ДатаПриема ▾									Общие итоги
	01.01.2011	02.01.2011	03.01.2011	04.01.2011	06.01.2011	09.01.2011	01.02.2011	09.02.2011	Общие итоги	
Количество ▾	Количество ▾	Количество ▾	Количество ▾	Количество ▾	Количество ▾	Количество ▾	Количество ▾	Количество ▾	Количество ▾	Нет итогов
Валиуллин Д.К.	1	2				1				
Галимянова О.А.	5							1		
Дюкина Я.Б.	4									
Елизарова Е.П.						2	2			
Жуков А.Е.			2							
Карбышев Г.О.						3	1			
Леонидова М.Р.				3			2			
Накипов Н.Д.		2						1		
Никоненко П.С.					2	1				
Родионова Ф.Я.			1							
Сидорович М.Л.					4		1			
					2					
					1					
					2					1
Халимов И.Ш.										
Харитоновна Н.Н.						4				
Шабернева Ф.Н.							1			2
Общие итоги										

Список полей сводной таблицы

Перетащите в таблицу

Журнал

- + ДатаПриема
- + ДатаПриема по неделям
- + ДатаПриема по месяцам
- + НомерКарточки
- + ФИО Пациента
- + ФИО Врача
- + ДолжностьВрача
- + Услуги
- + Количество
- + СтоимостиУслуги

Добавить в Строки

Рис. 41. Формирование итоговых значений

Журнал : таблица

ДолжностьВрача ▾
Все

ДатаПриема ▾

	01.01.2011	02.01.2011	03.01.2011	04.01.2011	06.01.2011	09.01.2011	01.02.2011	09.02.2011	Общие итоги
ФИО Врача ▾	Количество ▾	Количество ▾	Количество ▾	Количество ▾	Количество ▾	Количество ▾	Количество ▾	Количество ▾	Сумма "Количество"
Валиуллин Д.К.	1	2				1			4
Галимянова О.А.	1	2				1			6
Дюкина Я.Б.	5						1		6
Елизарова Е.П.	5						1		6
Жуков А.Е.	4								4
Карбышев Г.О.	4								4
Леонидова М.Р.			2						2
Накипов Н.Д.			2						2
Никоненко П.С.			4						4
Родионова Ф.Я.				3					3
Сидорович М.Л.				3					3
						2	1		3
						2	1		3
			1						1
			1						1
				4			1		5
				2					2

Список полей сводной таблицы

Перетащите в таблицу

- Журнал
- Итоги
 - Сумма "Количество"
- ДатаПриема
 - ДатаПриема по неделям
 - ДатаПриема по месяцам
- НомерКарточки
- ФИО Пациента
- ФИО Врача
- ДолжностьВрача
- Услуги
- Количество
- СтоимОднойУслуги

Добавить в Данные

Рис. 42. Сводная таблица с вычисляемыми полями

САМОСТОЯТЕЛЬНО: выполните фильтрацию данных и вывод результатов по должностям, используя поле *ДолжностьВрача*, выбрав **ВСЕ** или отдельные должности.

Также можно получить сводную диаграмму по таблице, выбрав на панели инструментов **Вид** → **Сводная диаграмма**.

ЗАДАНИЕ 15. СОЗДАНИЕ ОТЧЕТОВ

Отчеты используются для отображения информации, содержащейся в таблицах, в отформатированном виде, который легко читается как на экране компьютера, так и на бумаге.

Отчет можно отобразить в **трех режимах**:

- *конструктора*, позволяющем изменить внешний вид и макет отчета;
- *просмотра образца*, где можно просмотреть все элементы готового отчета, но и в сокращенном виде;
- *предварительного просмотра*.

Отчет можно создать в режиме **конструктора** и в режиме **мастера**.

Для создания отчета перейдите во вкладку **Отчеты** → **Создание с помощью мастера**.

• Выберите таблицу **«Штатное расписание»**, поля *Должность*, *Оклад*, *Надбавка*.

• Из таблицы **«Врачи»** выберите поля: *ID*, *ФИО Врача*, *ВидОказываемыхУслуг*. Нажмите кнопку **<Далее>**.

• В следующем диалоговом окне в строке **«Вид представления данных»** будет выбрана первая таблица, в нашем примере **«Штатное расписание»**.

• Нажмите кнопку **<Далее>**.

• Добавьте с помощью стрелок в уровень представления данных поле *Должность*.

• В следующих окнах выберите дизайн отчета.

• Нажмите **<Готово>**.

Пример готового отчета представлен на рис. 43.

Штатное Расписание

Должность	Оклад	Надбавка	id	ФИО Врача	Оказываемые Услуги
врач-стоматолог					
	7 000,00р.	2 500,00р.			
			104	Жуков А.Е.	Терапевтическая
			102	Дюкина Я.Б.	Терапевтическая
			112	Харитоновна Н.Н.	Терапевтическая
			109	Родионова Ф.Я.	Терапевтическая
			106	Леонидова М.Р.	Терапевтическая
			103	Елизарова Е.П.	Терапевтическая
			113	Шабернева Ф.Н.	Терапевтическая
			110	Сидорович М.Л.	Терапевтическая
ортопед					
	14 000,00р.	3 000,00р.			
			105	Карбышев Г.О.	Ортопедическая
			100	Валиуллин Д.К.	Ортопедическая
			108	Никоненко П.С.	Ортопедическая

Рис. 43. Отчет

САМОСТОЯТЕЛЬНО. Создайте отчеты по выполненным запросам.

ЗАДАНИЕ 16. СОЗДАНИЕ МАКРОСОВ

Макросы – это небольшие программы на языке макрокоманд СУБД Access, состоящие из последовательности определенных команд (одной или нескольких макрокоманд). Макросы являются простейшими средствами автоматизации действий над объектами Access. Необходимо отметить, что макросы легко создавать и для этого не нужно знать синтаксис языка программирования.

Для создания макроса откройте заглавную форму в режиме конструктора и добавьте кнопку. Пусть с помощью вновь созданной кнопки можно открыть отчеты для просмотра по деятельности стоматологии.

Щелкните правой кнопкой мыши по полю кнопки и выберите **Обработка событий** → **Макрос**. Откроется окно конструктора макросов.

В этом окне надо выбрать макрокоманду из выпадающего списка, щелкнув на кнопке открытия списка макрокоманд. В списке насчитывается около 50 различных макрокоманд для выполнения действий.

Для нашего примера выберите: *Открыть отчет* (рис. 44). Затем выберите имя отчета и режим: *Просмотр*. Сохраните макрос.

Все созданные макросы будут отображаться в вкладке **Макросы**.

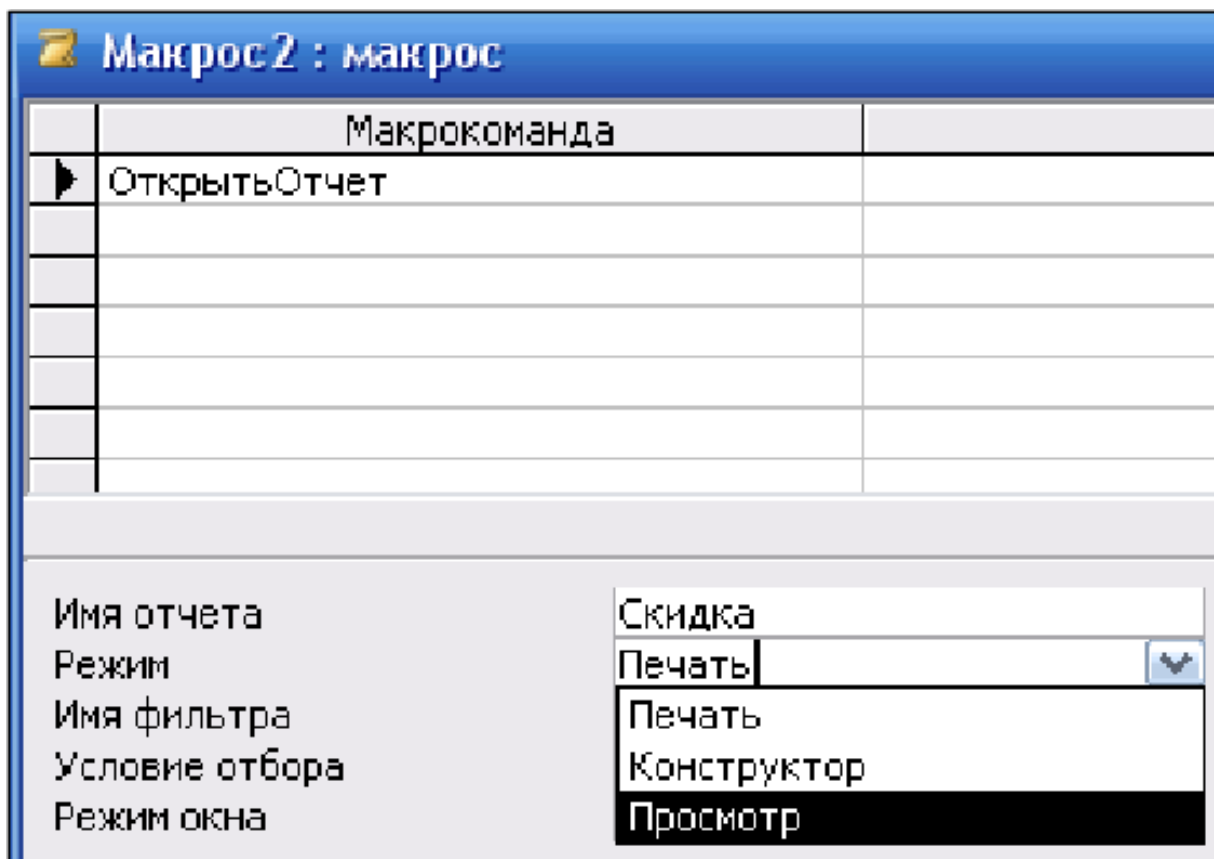


Рис. 44. Отчет

САМОСТОЯТЕЛЬНО: Добавьте остальные кнопки, дополнив их соответствующими командами-макросами.

Выполнение самостоятельной работы № 4. (стр. 51)

Вопросы для самостоятельной работы

1. Что называется информационной системой?
2. Раскройте понятие базы данных.
3. Что такое система управления базами данных (СУБД)?
4. Приведите примеры современных СУБД.
5. Как Вы понимаете термин «реляционная база данных»?
6. Назовите области применения информационных систем.
7. Что такое модель представления данных?
8. Что такое словарь данных?
9. Как Вы понимаете термин «вычислительная система»?
10. Назовите функции информационной системы.
11. Перечислите основные функции СУБД.
12. Какие классы (модели) организации баз данных Вы знаете?
13. Объясните, что такое архитектура базы данных.
14. Что такое концептуальное проектирование базы данных?
15. Объясните назначение и область применения баз данных.

16. Назовите основные компоненты СУБД.
17. С какой целью осуществляется контроль доступа к данным?
18. Возможно ли восстановление данных в случае утраты их в СУБД?
19. В чем заключается принцип телеобработки применимо к базам данных?
20. Приведите примеры файл-серверных решений, с которыми Вам удалось столкнуться.
21. В чем суть архитектуры «файл-сервер»?
22. В чем суть архитектуры «клиент-сервер»?
23. Укажите преимущества технологии «клиент-сервер».
24. Укажите преимущества технологии «файл-сервер».
25. Каким образом осуществляется многопользовательская обработка данных?
26. Какие подходы предусмотрены к распределению обязанностей между сервером и клиентом?
27. Что такое клиент и что такое сервер?
28. Что такое централизованное хранение данных?
29. Приведите классификацию СУБД.
30. Как разрабатываются пользовательские приложения?
31. В чем различия между персональной и многопользовательской СУБД?
32. Раскройте понятие языка описания данных.
33. Раскройте понятие языка манипулирования данными.
34. Поясните необходимость буферизации данных.
35. Что называется транзакцией?
36. Назовите основные свойства транзакции.
37. Приведите пример транзакций.
38. С какой целью осуществляется обеспечение целостности данных?
39. Назовите функциональные возможности СУБД Microsoft Access.
40. Назовите основные достоинства СУБД Microsoft Access.
41. Раскройте содержание технологии OLE и DDE.
42. Назовите типы объектов в СУБД Microsoft Access.
43. Что представляет собой таблица, форма, запрос и отчет в СУБД Microsoft Access?
44. Приведите примеры запросов к базе данных, содержащей информацию о студентах вуза.
45. Укажите типы данных, которыми оперирует СУБД Microsoft Access.
46. Как осуществляется контроль достоверности ввода данных в СУБД Microsoft Access?
47. Что такое маска ввода и как ей пользоваться. Поясните на примерах.
48. Какие типы связей между таблицами используются в СУБД Microsoft Access?
49. Как осуществляется связывание таблиц? Поясните на примерах.
50. Что называется первичным ключом и для чего он нужен?
51. Что такое язык структурированных запросов SQL и в чем его назначение?
52. Назовите основные команды языка SQL.
53. Приведите примеры, демонстрирующие работу языка SQL.

САМОСТОЯТЕЛЬНАЯ РАБОТА № 1

1. Разработать модель «Сущность-связь» по индивидуальному заданию и предварительно определить структуру таблиц базы данных. Согласовать с преподавателем.

2. Разработать структуру (схему данных) базовых таблиц (не менее трех) базы данных (смотри таблицу заданий к работе), удовлетворяющих требованиям целостности, непротиворечивости и избыточности. В таблицах в соответствии с типом данных, размещенных в каждом поле, определите наиболее подходящий тип для каждого поля.

3. Создать структуры базовых таблиц и наполнить их содержимым, состоящим более чем из 15 записей. При создании структуры таблиц целесообразно задавать ключевые (уникальные) поля. Это поможет в дальнейшем для организации связей между таблицами.

Варианты заданий

1	База данных «Студенческая библиотека». Ориентировочные таблицы: «Название», «Автор», «Издательство», «количество экземпляров»
2	База данных «Страховая фирма». Ориентировочные таблицы: «Виды страховок», «Клиенты\объекты», «Страховая деятельность»
3	База данных «Агентство недвижимости». Ориентировочные таблицы: «Объекты недвижимости», «Продажи», «Покупки»
4	База данных ГИБДД (Государственная инспекция безопасности дорожного движения)
5	База данных «Деканат вуза». Ориентировочные таблицы: «Список студентов», «Список предметов», «Сессия»
6	База данных отдела кадров производственного предприятия. Ориентировочные таблицы: «Сотрудники», «Штатное расписание», «Отделы», «Цеха»
7	База данных фирмы покупки и продажи автомобилей. Ориентировочные таблицы: «Продажи», «Покупки», «Автомобили»
8	База данных «Гостиница» Ориентировочные таблицы: «Номера», «Счета», «Клиенты»
9	База данных «Расчет квартплаты ТСЖ» Ориентировочные таблицы: «Список жильцов», «Оплаты», «Тарифы»
10	База данных «Железнодорожные кассы» Ориентировочные таблицы: «Продажи», «Посадочные места»,
11	База данных «Авиапассажирские перевозки» Ориентировочные таблицы: «Рейсы», «Самолеты», «Продажи»

12	База данных музея. Ориентировочные таблицы: «Экспонаты», «Авторы», «Экспозиции»
13	База данных «Спортивные комплексы района» Ориентировочные таблицы: «Нормативы», «Спортсмены», «Соревнования»
14	База данных «Экзаменационная сессия». Ориентировочные таблицы: «Предметы», «Оценки», «Студенты»
15	База данных «Турагентство». Ориентировочные таблицы: «Туры», «Продажи»
16	База данных «Аптека». Ориентировочные таблицы: «Товары», «Поставщики», «Продажи»
17	База данных «Сборка и реализация компьютеров». Ориентировочные таблицы: «Продукция», «Клиенты», «Заказы»
18	База данных «Продуктовые магазины района». Ориентировочные таблицы: «Продажи», «Отделы», «Товары».
19	База данных больницы (одного отделения). Ориентировочные таблицы: «Больные», «Диагнозы», «Врачи»
20	База данных «Видеотека». Ориентировочные таблицы: «Артисты», «Фильмы», «Продажи»

САМОСТОЯТЕЛЬНАЯ РАБОТА № 2

Для созданной базы данных по своему варианту:

1. Создать необходимые однотабличные формы.
2. Создать подчиненную или связанную форму для связанных таблиц.
3. Оформить главную кнопочную форму.

При создании форм использовать дополнительные кнопки, вычисляемые поля.

САМОСТОЯТЕЛЬНАЯ РАБОТА № 3

Для созданной Вами базы данных составьте список запросов и согласуйте их с преподавателем.

САМОСТОЯТЕЛЬНАЯ РАБОТА № 4

1. Создайте одну сводную таблицу в базе данных своего варианта.
2. Создайте все необходимые макросы: для открытия представления (запросов), просмотра, печати отчетов и другие, необходимые для удобной работы функции.

ЗАДАНИЕ НА РАЗРАБОТКУ БАЗЫ ДАННЫХ

Требуется разработать реляционную базу данных (БД) (основанную на реляционной модели данных) на заданную тематику в системе управления базами данных (СУБД) Microsoft Access.

Разрабатываемая реляционная база данных должна включать в себя таблицы с данными о выбранной предметной области, запросы на выборку данных из БД, формы и отчеты, а также запросы на языке SQL. Список возможных тем приведен в конце данного раздела.

Требования к разрабатываемой базе данных

1. БД должна включать в себя как минимум четыре таблицы с данными.
2. Общее количество атрибутов во всех таблицах БД должно быть не менее 25.
3. При создании атрибутов БД использовать различные (подходящие по смыслу) типы данных (счетчик, числовой, денежный, текстовый (различной размерности), логический, MEMO, OLE, и др.).
4. Необходимо обеспечить контроль достоверности ввода данных. Примерно 30% атрибутов должны иметь маску ввода.
5. Информация о предметной области должна быть сгруппирована и занесена в таблицы так, чтобы проектируемая БД удовлетворяла требованиям первых трех нормальных форм.
6. Общее количество записей (строк) во всех таблицах БД должно быть не менее 80.
7. Таблицы БД должны быть определенным образом связаны между собой с использованием ключевых полей таблиц БД (первичных ключей) и внешних ключей таблиц.
8. Связи между таблицами БД должны быть реализованы по технологии один к одному (1:1) и один ко многим (1:M).
9. Запросы к БД должны включать:
 - запросы на выборку данных (по различным критериям / условиям, в том числе запросы с параметрами);
 - запросы на добавление данных в таблицы (с параметрами);
 - запросы на удаление данных из таблицы (с параметрами);
 - запросы на обновление данных в таблицах (с параметрами);
 - запросы на создание новых таблиц с выборками данных (с параметрами);
 - прочие запросы.
- Общее количество запросов должно быть не менее 30 (включая запросы с параметрами).
10. На каждую таблицу БД должна быть сделана форма (или, например, может быть одна форма на 2 таблицы и т.п.). При проектировании форм использовать различные элементы формы (текстовые поля, надписи, кнопки, переключатели, графические элементы, вкладки и др. элементы), а также различные стили оформления этих элементов (шрифт, цвет, толщина линий и др.).

11. Проектируемая БД также должна включать в себя несколько отчетов (по выбору). Общее количество отчетов должно быть не менее 4. При оформлении отчетов необходимо использовать различные элементы и стили оформления.

12. БД должна иметь запросы на языке SQL (не менее 20). Типы запросов должны быть, как в п. 9.

Результаты проделанной работы должны быть оформлены и сданы преподавателю в виде отчета (40–50 страниц формата А4).

Содержание отчета

1. Титульный лист.

2. Содержание.

3. Введение. Во введении необходимо кратко отразить теоретические основы БД и СУБД. В конце раздела указывается и конкретизируется предметная область (тема), по которой планируется разработка базы данных. Дается краткое содержание отчета.

4. Основная часть. Основная часть должна включать в себя следующие отдельные разделы (в указанном порядке):

Структура таблиц базы данных. В разделе необходимо представить структуру всех таблиц БД с указанием типа данных каждого атрибута (представить каждую таблицу в режиме конструктора). Привести несколько примеров масок ввода. Далее необходимо указать первичные и внешние ключи таблиц и предложить возможную реализацию связей между таблицами. Привести окончательный вариант схемы данных (скриншот окна «схема данных» СУБД Access). В конце раздела приводятся примеры (фрагменты) заполненных данными таблиц, а также один пример, демонстрирующий работоспособность реализованных в БД связей (скриншот полностью развернутого списка от начальной таблицы (главной) до таблицы нижнего уровня).

Запросы к базе данных. В начале раздела перечисляются все существующие запросы. Далее приводятся все запросы в режиме конструктора (скриншоты) и результаты их выполнения (если запрос требует ввода параметра, то его также необходимо указать).

Формы и отчеты. В данном разделе приводятся скриншоты всех форм и отчетов базы данных. Несколько форм и отчетов приводятся в режиме конструктора (1–2 формы, 1–2 отчета). Выборочно можно прокомментировать функциональность некоторых форм и отчетов.

SQL-запросы. В разделе перечисляются и поясняются все запросы на языке структурированных запросов SQL.

5. Заключение (1–2 стр.). В заключении дается краткое описание разработанной базы данных. Указываются технологии, использованные при проектировании БД (например, можно указать: применен контроль достоверности данных (маски ввода), реализованы связи типа один ко многим, разработаны кнопочные формы и отчеты для наглядного представления информации пользователю, реализованы различные запросы к данным для извлечения информации

по определенным критериям, предложен рад запросов на языке структурированных запросов SQL и т.п.). В конце раздела отмечаются функциональные возможности и достоинства спроектированной БД.

б. Список литературы.

Примечание. Все рисунки и таблицы отчета обязательно должны быть пронумерованы (рис. 1. «название рисунка», табл. 1 «название таблицы»). Номер страницы проставляется внизу листа, начиная со страницы «введение».

ВАРИАНТЫ ЗАДАНИЙ НА СОЗДАНИЕ ТАБЛИЦ

1. Отдел кадров (ФИО, адрес, телефон, дата рождения, должность, ИНН, образование).
2. Библиотека (название книги, ФИО автора, количество страниц, цена, количество экземпляров, издательство, год издания).
3. Кадровые агентства (фирма, телефон фирмы, должность, количество мест, оклад, требования, срок контракта).
4. Ведомость по зарплате (фамилия, имя, отчество, табельный номер, должность, разряд, сумма к выдаче).
5. Бытовая техника (товар, изготовитель, дата выпуска, цена, гарантия, поставщик, кредит).
6. Отдых (мероприятие, место проведения, дата проведения, время начала, стоимость билета, организатор, номер телефона).
7. Телеком (ФИО абонента, адрес, номер телефона, тариф за 1 мин., время разговоров, льготы, сумма к оплате).
8. Турагенство (страна, дата вылета, количество дней, количество путевок, стоимость, доп. услуги, фирма-организатор).
9. Деканат (ФИО студента, группа, дата рождения, дата поступления, прописка, специальность, национальность).
10. Спортивный магазин (инвентарь (одежда), материал, фирма, цвет, производитель, размер, количество).
11. Абитуриент (фамилия, имя, отчество, средний балл аттестата, предмет сдачи, оценка, факультет).
12. Аптечный склад (название, лекарственная форма, фирма-изготовитель, дата выпуска, срок годности, количество, вид применения).
13. Видеосалон (название фильма, продолжительность просмотра, жанр, киностудия, цена, год создания фильма, главный герой (имя актера)).
14. Аудио салон (название песни, исполнитель, авторы, год выпуска, продолжительность, цена, рейтинг).
15. Мотоциклы (марка, год выпуска, пробег, цвет, производитель, цена, максимальная скорость).
16. Автомобили (марка, год выпуска, объем двигателя, производитель, цвет, цена, прежний владелец).
17. Экзаменационная ведомость (ФИО, факультет, специальность, предмет, оценка, дата сдачи, преподаватель).

18. Бытовая химия (товар, фирма-производитель, форма выпуска, назначение, цена, срок годности, дата выпуска).
19. Справочное бюро (ФИО, адрес, дата рождения, семейное положение, пол, номер телефона, примечания).
20. Магазины (название, адрес, деятельность, скидки, хозяин, площадь помещения, год постройки).
21. Континент (страна, население, язык, валюта, дата образования государства, импорт, экспорт).
22. Спортсмены (ФИО, вид спорта, клуб, год рождения, цена контракта, рейтинг, срок контракта).
23. Новобранец (ФИО, образование, специальность, год рождения, род войск, место службы, срок службы).
24. Футбол (фамилия, имя, отчество, клуб, пас, гол, очки).
25. Музыка (ФИО исполнителя, песня, дата записи, продолжительность, ФИО автора, ФИО композитора, рейтинг).
26. Телевизоры (марка, производитель, год выпуска, цвет, диагональ, цена, гарантия).
27. Фирмы (название фирмы, вид деятельности, дата основания, кол-во сотрудников, телефон, адрес, ИНН).
28. Школа (ФИО ученика, класс, уклон, адрес, телефон, год рождения, хобби).
29. Сотрудники (Фамилия, Имя, Отчество, год рождения, год поступления на работу, образование, количество детей).
30. Зарплата (ФИО, отдел, оклад, УК, КТУ, ПН, табличный номер).
31. Продукт (наименование, дата выработки, срок годности, изготовитель, цена, количество, калорийность).
32. АВИА (ФИО пассажира, вес багажа, количество мест багажа, номер рейса, пункт назначения, дата вылета, место).
33. Ж/Д (ФИО пассажира, номер поезда, тип вагона, номер вагона, место, дата отправления, дата прибытия).
34. Паспортные данные (Фамилия, Имя, Отчество, дата рождения, место рождения, № паспорта, кем выдан).
35. Склад (наименование материала, шифр, материала, цена, № склада, кол-во, дата изготовления).
36. Анкета (ФИО, пол, год рождения, стаж работы, место работы, образование, профессия).
37. График учебного процесса (специальность, номер семестра, число учебных недель, количество экзаменов, количество зачетов, КР (дисциплина), дата сессии).
38. Кулинария (название блюда, вид блюда, размер порции, сложность, страна, время приготовления, срок годности).
39. Депутаты (Фамилия, Имя, Отчество, дата рождения, стаж работы, фракция, избирательный округ).
40. Соревнования (клуб, вид спорта, место, результат, фамилия, разряд, дата).
41. Ателье (ФИО заказчика, вид изделия, вид материала, размер, цена, дата заказа, срок изготовления).

42. Жилой фонд (улица, номер дома, номер квартиры, площадь, цена, этаж, дата постройки).
43. Недвижимость (название, адрес, цена, дата постройки, ФИО владельца, площадь, характеристика).
44. Почта (индекс, адрес, дата, вид перевода, вес, ФИО получателя).
45. Телеграф (телефон абонента, куда звонил, продолжительность разговора, стоимость 1 мин, время, дата, стоимость к оплате).
46. Расписание (факультет, специальность, группа, день недели, время, аудитория, дисциплина).
47. Календарь (число, месяц, год, день недели, событие, долгота дня, восход солнца).
48. Россия (область, площадь, население, основное производство, дата образования, губернатор, столица).
49. Вузы (наименование, страна (область), количество студентов, количество специальностей, ФИО ректора, год образования, рейтинг).
50. Рыбалка (место рыбалки (водоем), название рыбы, вес улова, название насадки, вид снасти, дата, ФИО рыбака).
51. Производство (название предприятия, адрес, наименование изделия, материал, производительность за день, средняя зарплата, ФИО директора).
52. Квартплата (месяц, владелец квартиры, площадь квартиры, стоимость жилья, газ, электричество, льготы).
53. Городской транспорт (город, вид транспорта, номер маршрута, время в пути, стоимость проезда, протяженность маршрута, дата основания).
54. Банки (наименование, дата образования, адрес, вклад, срок, первоначальный, валюта, ставка).
55. Больница (Фамилия врача, Имя, отчество, номер кабинета, номер телефона, специальность, режим работы).
56. Физкультура (ФИО студента, номер группы, пол, возраст, упражнения, единицы измерения, результат).
57. Прогноз погоды (дата, страна, город, осадки, температура, ветер, давление).
58. Личный состав взвода (ФИО, возраст, звание, должность, срок контракта, номер табельного оружия, дата призыва).
59. Зоопарк (питомец, номер клетки, где обитает, чем питается, возраст, продолжительность жизни, дрессировка).
60. Сборная России по хоккею (ФИО, клуб, страна, позиция, номер игрока, рост, вес).
61. Одежда (наименование, цена, цвет, сезон, количество, название магазина, адрес магазина).
62. Аукцион (наименование, номер лота, старт. цена, автор (производитель), год выпуска, материал, прежний владелец).
63. Ясли (ФИО ребенка, количество лет, рост, вес, адрес, телефон, оплата за месяц).
64. Галактика (название планеты, расстояние от Земли, размер, обитатели, связь с Землей, дата открытия, принадлежность к системе).

65. Рекорды (Фамилия, Имя, Отчество, тип рекорда, страна, результат, дата установления).
66. Рынок (наименование, номер лотка, единицы измерения, цена за единицу, дата изготовления, изготовитель, количество).
67. Кинотеатр (название, адрес, картина, время начала, дата проката, цена за билет, количество билетов).

ТЕМЫ РЕФЕРАТОВ

1. Стандартизация средств СУБД. Стандартизированный язык запросов по шаблону QBE.
2. Язык структурированных запросов SQL.
3. Новые информационные технологии хранения и обработки документов.
4. Организация индексирования в документно-ориентированных БД. Полнотекстовый поиск в документно-ориентированных БД.
5. Архитектура «клиент-сервер» (централизованная модель, распределенная модель, мультипроцессорная модель. Двухуровневая архитектура клиент-сервер.
6. Трехуровневая архитектура «клиент-сервер».
7. Модели организации сетевых баз данных.
8. Способы поддержания целостности распределенных БД.
9. Примеры организации сетевых баз данных.
10. Технологии корпоративных информационных систем.
11. Автоматизированные системы управления документооборотом.
12. Организация корпоративных информационных систем.
13. Понятие корпоративной информационной системы (КИС). Структура КИС. Основные требования к КИС.
14. Методологии, используемые при создании КИС.
15. Методология IDEFO.
16. Методологии DFD и IDEF3.
17. Методология IDEF1X.
18. Организация защиты информации в корпоративных информационных системах.
19. Экспертные системы.
20. Классификация знаний. База знаний. Структура, этапы проектирования и принципы функционирования.
21. Применение экспертных систем в системах организационного управления.
22. Характеристики объектно-ориентированных БД. Сравнение реляционных и объектно-ориентированных БД.
23. Стратегия и перспективы развития БД в 21 веке.
24. Федеративный доступ к базам данных.
25. Анализ качества баз данных.
26. Защита информации в базах данных.
27. Архитектура и функционирование адресных баз данных.
28. Технологии объектных баз данных.

29. Объектно-ориентированные базы данных: основные концепции, организация и управление.
30. Системная архитектура и структура ORACLE.
31. Стратегические направления в системах баз данных.
32. Стратегии резервного копирования и восстановления БД SQLBase.
33. Современные системы управления базами данных.
34. Проектирование баз данных.
35. Нормализация в базах данных. Нормальные формы.

Библиографический список

Основной

1. Малыхина, М.П. Базы данных: основы, проектирование, использование, /М.П. Малыхина – 2-е изд. перераб. и доп. – СПб.: БХВ-Петербург, 2006. – 529 с.: ил.
2. Невиницын, В.Ю. Разработка реляционных баз данных в СУБД Microsoft Access: учеб. пособие / В.Ю. Невиницын, М.А. Чешинский; под ред. А.Н. Лабутина; Иван. гос. хим.-технол. ун-т. – Иваново, 2015. – 92 с.
3. Кузин, А. В. Базы данных : учеб. пособие для вузов по направлению подготовки дипломированных специалистов «Информатика и вычисл. Техника»/ А. В. Кузин – 4-е изд., стер. –М. : Академия, 2010. – 316 с. : ил. - (Высшее профессиональное образование). - Библиогр. : с. 313. –ISBN 978-5-7695-7368-2.

Дополнительный

1. Кузнецов, С.Д. Основы баз данных. Курс лекций : учеб. пособие для вузов по спец. в области информационных технологий / С.Д. Кузнецов–М. : Интернет-Университет Информ. Технологий, 2005. –484 с.: ил. –(Серия "Основы информационных технологий"). - ISBN 5-9556-0028-0.
2. Баженова, И. Ю. Основы проектирования приложений баз данных : учеб. пособие/ И.Ю. Баженова –М. : Интернет-Университет Информ. Технологий; БИНОМ Лаборатория знаний, 2006. –234 с.: ил. –(Основы информационных технологий). –Библиогр.: с. 323. –ISBN 5-94774-539-9 (БИНОМ. ЛЗ) . –ISBN 5-9556-0068-X.

Методические указания для обучающихся по освоению дисциплины

1. Системы управления базами данных: методические указания к лабораторным работам [Электронный ресурс] / сост. Е.С. Шандаров— Томск: ТУСУР, 2012. — 41 с. — Режим доступа: <https://edu.tusur.ru/publications/2140>
2. Базы данных: Методические указания по проведению лабораторных, практических и самостоятельной работ студентов для направления 090301 – Информатика и вычислительная техника [Электронный ресурс] / сост. Д.П. Вагнер –Томск: ТУСУР, 2016. – 62с. – Режим доступа: <https://edu.tusur.ru/publications/6165>
3. Базы данных: методические указания для выполнения лабораторных и самостоятельных работ для студентов направления: 080500.62 – «Бизнес-информатика» [Электронный ресурс] / сост. П.В. Сенченко –Томск: ТУСУР, 2015. –31с. – Режим доступа: <https://edu.tusur.ru/publications/5174>

4. Управление данными: методические указания по проведению лабораторных, практических и самостоятельной работ для студентов направлений 230400 – Информационные системы и технологии [Электронный ресурс] / сост. Д.П. Вагнер–Томск: ТУСУР, 2014. –55с. –Режим доступа:

<https://edu.tusur.ru/publications/3952>

Перечень ресурсов сети «Интернет», необходимых для освоения дисциплины

1. Электронный каталог ИГХТУ – <http://www.isuct.ru>.

2. Электронная библиотека Ивановского государственного химико-технологического университета с полнотекстовыми документами – <http://www.isuct.ru/e-lib>.

3. Виртуальная образовательная среда Ивановского государственного химико-технологического университета – http://edu.isuct.ru/course/index.php?_CategoryId=13.

4. ЭБС «Лань» – <http://e.lanbook.com/books>.

5. Национальная электронная библиотека – <http://нэб.рф>.

6. Федеральный портал «Российское образование» – <http://www.edu.ru>.

7. Информационная система «Единое окно доступа к образовательным ресурсам» – <http://window.edu.ru>.

Учебное издание

**Методические указания для самостоятельной работы студентов
по курсу «Системы управления базами данных»**

Составитель **Князева** Елена Яковлевна

Редактор О.А. Соловьёва

Подписано в печать 17.04.2019. Формат 60×84 1/16. Бумага писчая.
Усл. печ. л. 3,7. Тираж 50 экз. Заказ

ФГБОУ ВО «Ивановский государственный
химико-технологический университет»
Отпечатано на полиграфическом оборудовании
редакционно-издательского центра ФГБОУ ВО «ИГХТУ»
153000, г. Иваново, пр. Шереметевский, 7