

Документ подписан простой электронной подписью
Информация о владельце:
ФИО: Баламирзоев Назим Лимдинович
Должность: Ректор
Дата подписания: 11.09.2025 13:36:02
Уникальный программный ключ:
5cf0d6f89e80f49a334f6a4ba58e91f3326b9926

Министерство науки и высшего образования Российской Федерации
ФГБОУ ВО «Дагестанский государственный технический университет»

Учебно-методические указания
к выполнению лабораторных работ №1-№8
по дисциплине «Технологии разработки и защиты баз данных» для студентов
по направлению бакалавриата 01.03.02 «Прикладная математика и
информатика» профиль подготовки «Системное программирование и
компьютерные технологии».

УДК 681.3.06(072)

Учебно-методические указания к выполнению лабораторных работ №1-№8 по дисциплине «Технологии разработки и защиты баз данных» для студентов по направлению бакалавриата 01.03.02 «Прикладная математика и информатика» профиль подготовки «Системное программирование и компьютерные технологии». –Махачкала: ИПЦ ДГТУ, 2021.-36с.

Учебно-методические указания подготовлены в соответствии с Государственным образовательным стандартом высшего профессионального образования.

Учебно-методические указания содержат описания лабораторных работ по темам:

1. Основы проектирования структуры БД.
2. Создание однотабличной базы данных. Заполнение базы данных.
3. Создание новых таблиц. Создание схемы данных.
4. Ввод и просмотр данных посредством формы. Создание многотабличной формы
5. Формирование запросов на выборку.
6. Создание сводных таблиц и диаграмм.
- 7-8. Создание и настройка отчетов.

Каждая лабораторная работа содержит теоретический материал и практическую часть.

Составители: ст. преподаватель кафедры «Прикладной математики и информатики» Алиосманова О.А.

Рецензент:

Зав. кафедрой «ПОВТ и АС» ДГТУ, к.э.н., доцент Т.Г.Айгунов

Ведущий научный сотрудник лаборатории комплексного освоения возобновляемых источников энергии Института проблем геотермии и возобновляемой энергетики – филиала ОИВТ РАН, д.т.н. Д.Н. Кобзаренко

Печатается согласно постановлению
Ученого совета Дагестанского государственного технического университета
от _____ 2021 г.

Оглавление

Лабораторная работа № 1. Основы проектирования структуры БД.	4
Лабораторная работа № 2. Создание однотабличной базы данных. Заполнение базы данных.	5
Лабораторная работа №3. Создание новых таблиц. Создание схемы данных.	11
Лабораторная работа № 4. Ввод и просмотр данных посредством формы. Создание многотабличной формы.....	14
Лабораторная работа № 5. Формирование запросов на выборку.	18
Лабораторная работа № 6. Создание сводных таблиц и диаграмм.	21
Лабораторная работа № 7-8. Создание и настройка отчетов. Создание макросов.....	24
Задания на лабораторную работу.....	27
Набор данных к варианту 1	27
Набор данных к варианту 2	28
Набор данных к варианту 3	29
Набор данных к варианту 4	30
Набор данных к варианту 5	31
Набор данных к варианту 6	32
Набор данных к варианту 7	33
Набор данных к варианту 8	34
Набор данных к варианту 9	35
Набор данных к варианту 10	36
Список литературы	

Лабораторная работа № 1. Основы проектирования структуры БД.

Цель работы: спроектировать БД для выбранной предметной области согласно примеру, представленному в методическом указании. Провести нормализацию (до 3 нормальной формы).

Проектирование схемы БД должно решать задачи минимизации дублирования данных и упрощения процедур их обработки и обновления. При неправильно спроектированной схеме БД могут возникнуть аномалии модификации данных. Они обусловлены отсутствием средств явного представления типов множественных связей между объектами ПрО и неразвитостью средств описания ограничений целостности на уровне модели данных. Для решения подобных проблем проводится нормализация отношений. Механизм нормализации реляционных отношений разработал Э.Ф. Кодд (E.F. Codd). Этот механизм позволяет по формальным признакам любое отношение преобразовать к третьей нормальной форме.

Построение ER-модели

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*.

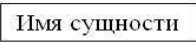
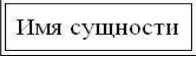
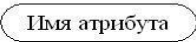
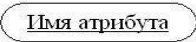
Атрибут представляет собой свойство сущности. Так, атрибутами сущности ВРАЧИ являются: *табельный номер, ФИО, должность* и т.д.

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.

Таблица 1. 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. (см.рис. 1).



Рис.1. Пример ER-диаграммы

Следовательно, для построения ER-модели необходимо сначала выявить сущности, характерные для заданной предметной области, и определить для каждой сущности набор атрибутов, имеющих значение с учетом поставленной задачи. Затем необходимо определить все имеющиеся связи между сущностями и характеристики этих связей. После этого строится ER-диаграмма, т. е. собственно графическое изображение модели с использованием одной из существующих нотаций. Для простых случаев ER-диаграмму можно рисовать вручную, для более сложных можно воспользоваться специализированным программным обеспечением, известным как CASE-средства.

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

Лабораторная работа № 2. Создание однотабличной базы данных. Заполнение базы данных.

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.

1.1. Для создания 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*

1.2. Для создания таблицы базы данных в окне базы данных выберите вкладку *Создание*, а затем щелкните по кнопке <Конструктор таблиц> (рис. 2).

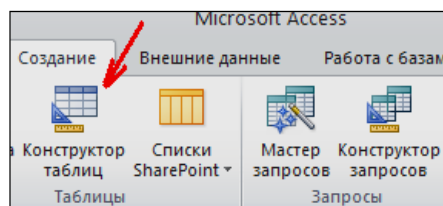


Рис. 2. Создание таблиц в режиме Конструктора

1.3. Для определения 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 (см.рис. 3).

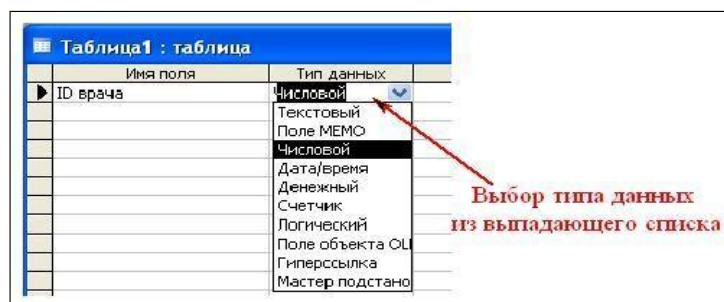


Рис. 3. Выбор типа данных

- для определения всех остальных полей таблицы базы данных *Врачи* выполните действия, аналогичные указанным выше в соответствии с табл. 2.

Таблица 2. 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)

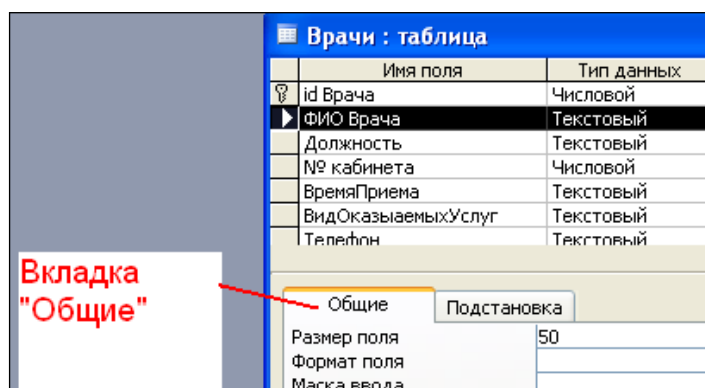
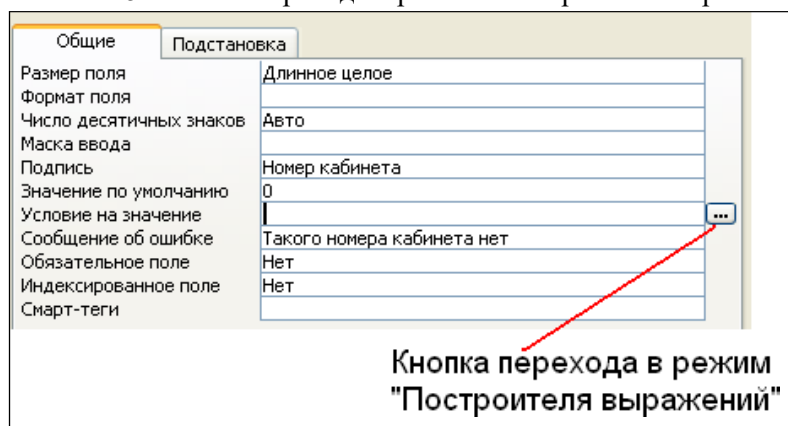


Рис. 4. Использование для поля дополнительных параметров с помощью вкладки «Общие»

1.4. 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. 4) и введите условие: $\geq 100 \text{ And } \leq 130 \text{ Or } \geq 200 \text{ And } \leq 230$ (см. рис. 5). В строке «Сообщение об ошибке» введите: «Такого номера кабинета нет».

Рис. 5. Кнопка перехода в режим «Построителя выражений»



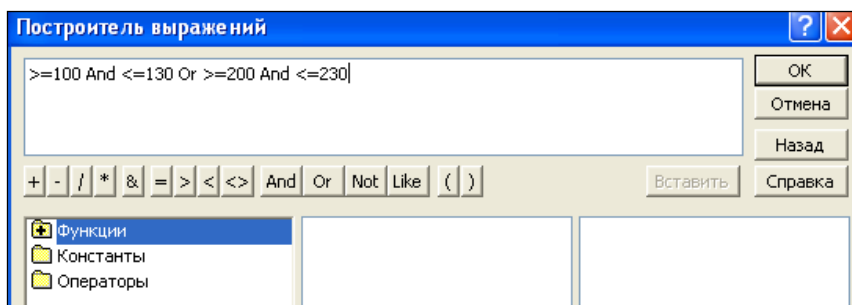


Рис. 6. Окно «Построителя выражений»

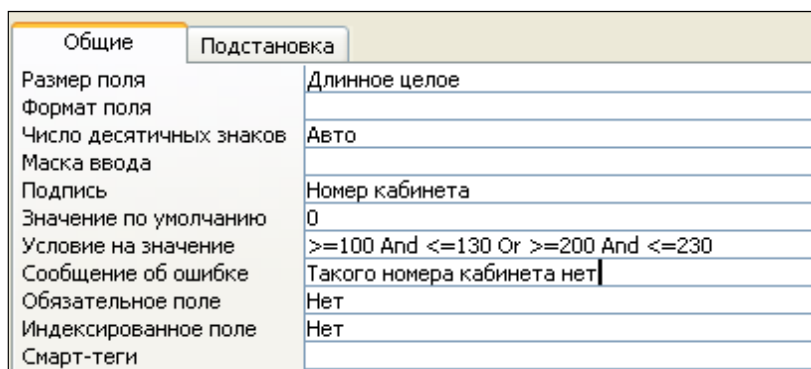


Рис. 7. Вкладка «Общие» с заданными условиями для поля «№ кабинета»

1.5. 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. 8):

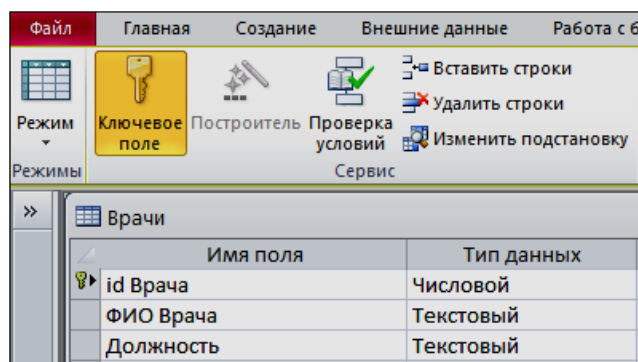


Рис. 8. Назначение поля ключевым

Закройте таблицу. MS Access предложит сохранить таблицу (по умолчанию «Таблица 2»). Задайте имя таблицы «Врачи».

Заполнение базы данных.

- 1) Создать таблицы-подстановки.
- 2) Заполнить таблицы данными.
- 3) Создать подстановочное поле.
- 4) Ввести ограничения на данные в поле ID врача; эти данные не должны повторяться.

2.1. Создание таблиц-подстановок.

Для упрощения ввода данных в таблицу «Врачи» создайте таблицы-подстановки для полей: «ВремяПриема» и «ВидОказываемыхУслуг». 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.

2.1.1. Таблица «Смена-подстановка» (используется для поля «ВремяПриема»). Поля таблицы (см.табл.3):

Таблица 3.Описание свойств полей таблицы «Смена_подстановка»

Смена_подстановка : таблица	
Имя поля	Тип данных
НомерСмены	Текстовый
ВремяПриема	Текстовый

Примечание 1. В результате щелчка по кнопке <ОК> MS Access предложит вам задать ключевое поле. Для создаваемой таблицы щелкните по кнопке <Нет>.

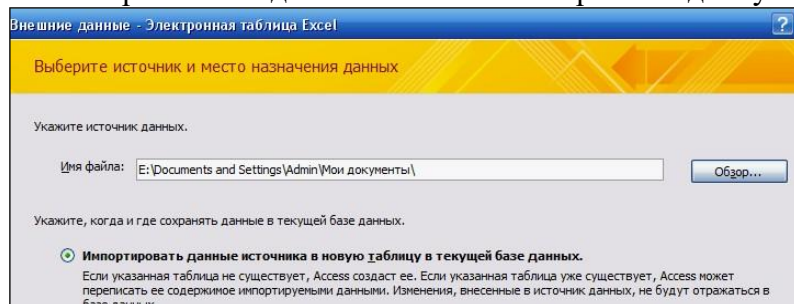
2.1.2. Таблицу «НаименованиеУслуги_подстановка» (используется для поля «Вид оказываемых услуг») создайте в **Excel**, чтобы импортировать её в нашу базу данных. Поля таблицы (см.табл.4):

Таблица 4. 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»,
- далее подтвердите описание и название полей,
- далее выберите «Не создавать ключ», Готово.

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

2.1.3. Таблица «Штатное расписание» (данные поля «НаименованиеДолжности» будут использоваться для подстановки в таблицу «Врачи» (см.табл. 5).

Таблица 5. Описание свойств полей таблицы «Штатное расписание»

ШтатноеРасписание : таблица	
Имя поля	Тип данных
Должность	Текстовый
Оклад	Денежный
Надбавка	Денежный

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

2.2. 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 (см.рис. 9).

Рис. 9. Таблица «Смена_подстановка» с данными

Смена_подстановка	
НомерСме	ВремяПрие
1	08.00-12.00
2	13.00-19.00

- Заполните таблицу «ШтатноеРасписание» данными (см.рис. 10).

Штатное Расписание : таблица			
	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. 10. Таблица «ШтатноеРасписание» с данными

2.3. Создание подстановочного поля.

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**, или
- использование языка запросов SQL (с помощью построителя запросов).

1) 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» (см.рис. 11):

Имя поля	Тип данных
id Врача	Числовой
ФИО Врача	Текстовый
Должность	Текстовый
№ кабинета	Числовой
ВремяПриема	Текстовый
ВидОказываемыхУслуг	Текстовый
Телефон	Текстовый
ФотоВрача	Поле объекта

Общие	Подстановка
Тип элемента управления	Поле со списком
Тип источника строк	Таблица или запрос
Источник строк	ШтатноеРасписание
Присоединенный столбец	Рейтинг
Число столбцов	Скидка
Заглавия столбцов	Смена_подстановка
Ширина столбцов	СписокПациентов
Число строк списка	СтоимостьУслугВрачей
Ширина списка	СтоимостьУслугВрачей_1
Ограничиться списком	СуммаВсего_1

ШтатноеРасписание
Рейтинг
Скидка
Смена_подстановка
СписокПациентов
СтоимостьУслугВрачей
СтоимостьУслугВрачей_1
СуммаВсего_1

Рис. 11. Подстановка в режиме **Конструктора**

2) Для создания подстановки для 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. 12).

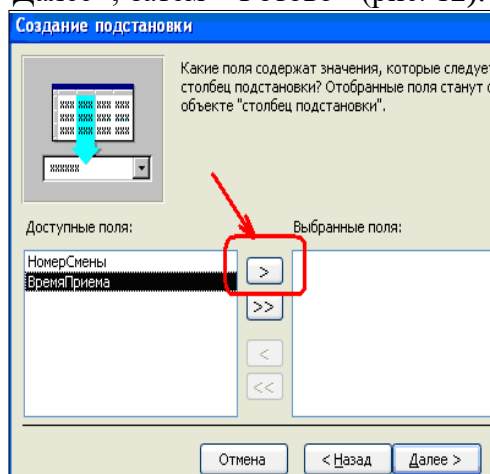


Рис.12. Использование Мастера подстановки

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

- 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 будет иметь вид:

SELECT Смена_подстановка.ВремяПриема FROM Смена_подстановка;

САМОСТОЯТЕЛЬНО. Для поля «ВидОказываемыхУслуг» создайте подстановочное поле.

2.4. 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 2. Индекс - это средство MS Access, ускоряющее поиск и сортировку данных в таблице.

2.5. Заполните таблицу «Врачи» данными, используя подстановку (см.пример: рис. 13):

Использование режима "Подстановка"						
Врачи : таблица						
	Табельный ном	ФИО Врача	Должность	Номер кабин	ВремяПриема	ВидОказываемыхУслуг
▶	100	Валиуллин Д.К.	ортопед	101	08.00-12.00	Ортопедическая
	101	Галимянова О.А.	врач-стоматолог	102	08.00-12.00	Хирургическая
	102	Дюкина Я.Б.	ортопед	203	08.00-12.00	Терапевтическая
	103	Елизарова Е.П.	хирург-терапевт	115	08.00-12.00	Терапевтическая
	104	Жуков А.Е.	врач-стоматолог	120	08.00-12.00	Терапевтическая
	105	Карбышев Г.О.	ортопед	214	08.00-12.00	Ортопедическая
	106	Леонидова М.Р.	врач-стоматолог	224	13.00-19.00	Терапевтическая
	107	Накипов Н.Д.	хирург-терапевт	315	08.00-12.00	Хирургическая
	108	Никоненко П.С.	ортопед	124	13.00-19.00	Ортопедическая

Рис. 13. Пример заполнения таблицы «Врачи» с использованием режима **Подстановки**

Лабораторная работа №3. Создание новых таблиц. Создание схемы данных.

Цель работы: Создание базы данных (таблицы и связи между ними) средствами СУБД Microsoft Access

3.1. Создайте таблицу «**СписокПациентов**», которая содержит следующие поля:

СписокПациентов : таблица		
	Имя поля	Тип данных
	НомерКарточки	Числовой
	ФИО Пациента	Текстовый
	ДомашнийАдрес	Текстовый
	МедПолис	Текстовый
	Телефон	Текстовый
	Пол	Текстовый

Рис. 14. Таблица «СписокПациентов»

- Для поля «МедПолис» создайте маску ввода по образцу: «АБ-1234».
- Для поля «Телефон» создайте маску ввода как в задании 3.
- Для поля «Пол» во вкладке **Общие** в строке «Значение по умолчанию» введите: муж.

Это значение будет указано по умолчанию. В случае необходимости его можно заменить.

- Заполните таблицу данными (не менее 15 записей).

3.2. Создайте таблицу «НаименованиеУслуг», которая содержит следующие поля:

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

Рис. 15. Таблица «НаименованиеУслуг».

Заполните таблицу данными по примеру:

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

3.3. Создайте таблицу «**Журнал**», которая содержит следующие поля:

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

Рис. 16. Таблица «Журнал».

1) Для поля «НомерКарточки» используйте подстановку:

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

2) Поля «*ФИО пациента*» и «*Должность врача*» не заполняйте. Для их заполнения будет использован **запрос на обновление**.

3) Для поля «*ФИО врача*» используйте подстановочное поле из таблицы «*Врачи*».

4) Для поля «*Услуги*» используйте подстановку из таблиц услуг (таблица «*НаименованиеУслуг*»).

Создание схемы данных

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.

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 (см.рис.17). Закройте окно.

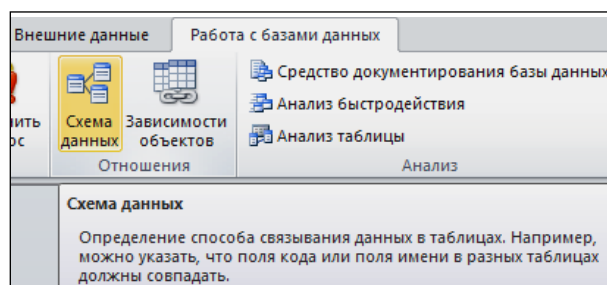


Рис. 17. Окно **Схема данных**

Примечание 3: Жирным текстом в схеме будут выделены поля, имеющие ключевое значение.

3. Установите связь между таблицами *СписокПациентов* и *Журнал*. Для этого установите курсор на имя поля «**НомерКарточки**» - в этой таблице это поле - ключевое (таблица *СписокПациентов*) 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.

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

Примечание 4. Это возможно сделать только в том случае, если типы и размер полей заданы одинаково.

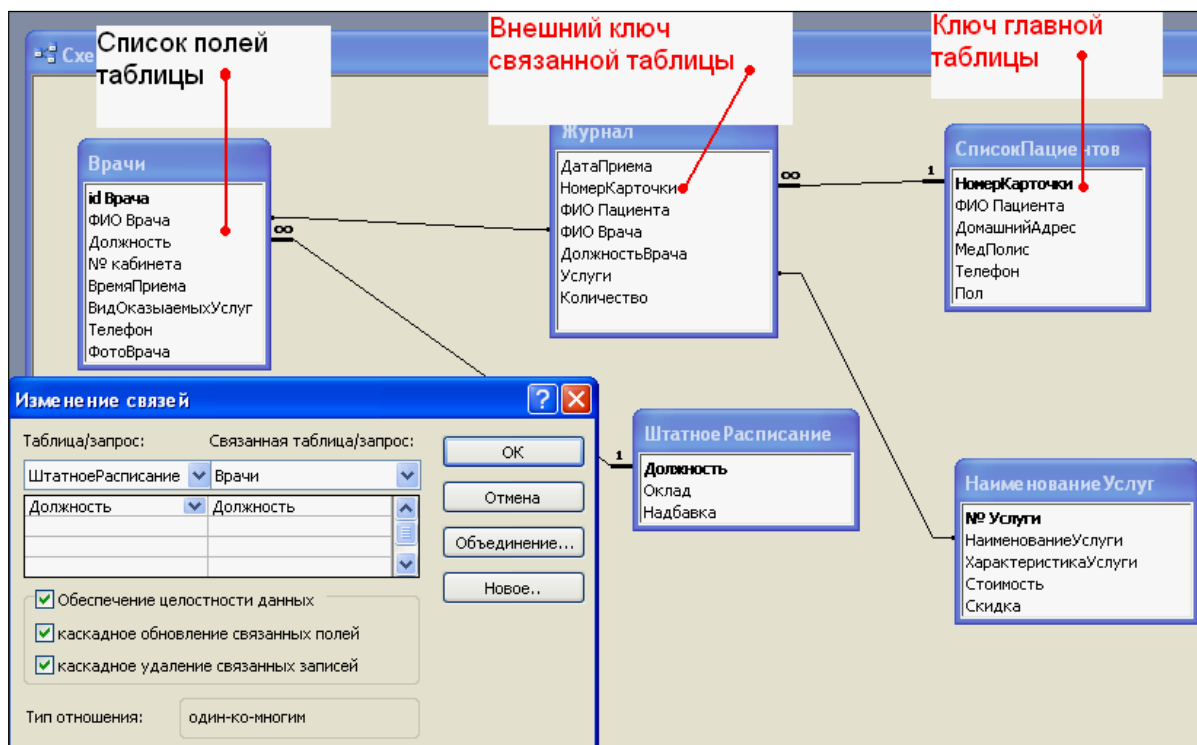


Рис. 18. Создание схемы данных

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

Внимание! Изменить связь и параметры объединения можно, щелкнув правой кнопкой мыши по линии связи, как показано на рис. 19. Откроется окно **Изменение связей**. Выбрав кнопку **Объединение...**, можно выбрать другой способ, что позволяет перейти к созданию связи между любыми двумя таблицами базы данных

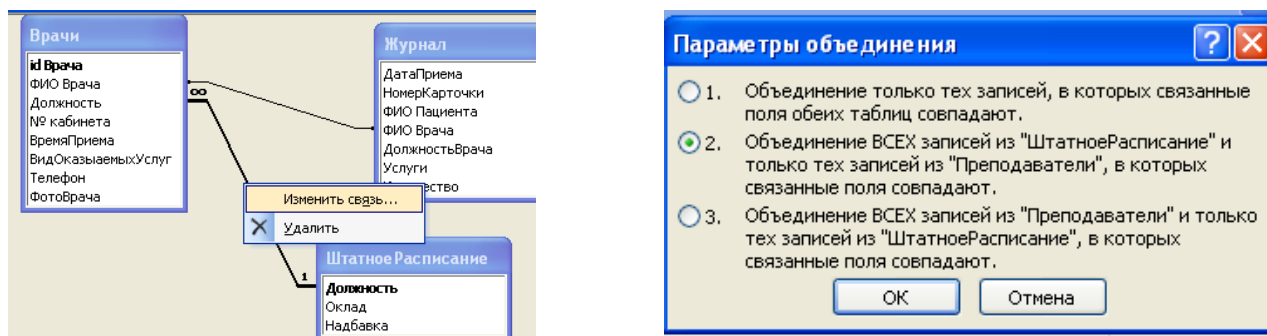


Рис. 19. Изменение или удаление связи

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.

Лабораторная работа № 4. Ввод и просмотр данных посредством формы. Создание многотабличной формы

Цель работы: Создание, ввод и просмотр данных посредством многотабличной формы.

1) Создать формы для ввода данных с последующей их модификацией.

2) Создать главную форму «Заставка».

Формы являются основным средством организации интерфейса пользователя в приложениях Access.

4.1. 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**.

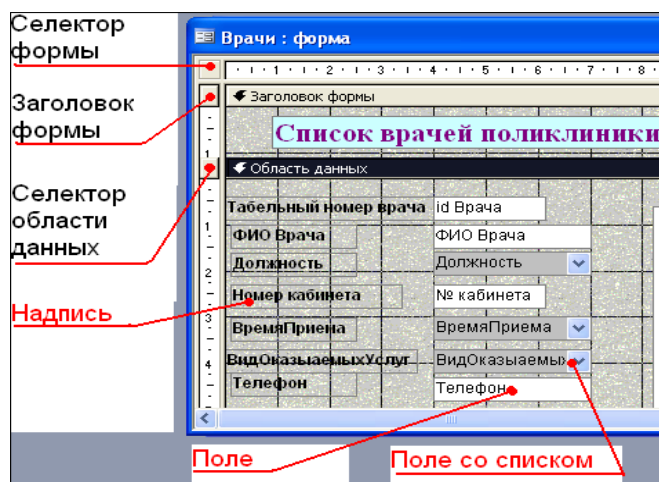


Рис. 20. Форма в режиме **Конструктора**

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.

4.2. Для создания главной кнопочной формы создайте 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 форм**. Появится пустая форма. Задайте мышкой размеры формы. → **Конструктор**

Откроется вкладка **Конструктор** - панель элементов, которая позволяет создавать элементы управления на форме и осуществлять необходимые действия при конструировании (рис. 21):

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

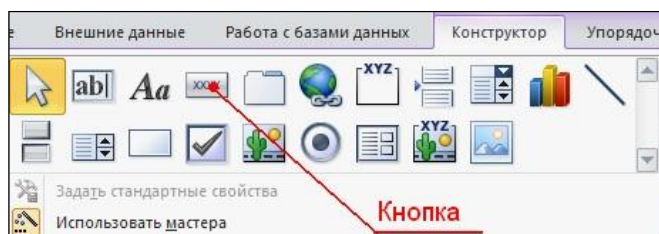
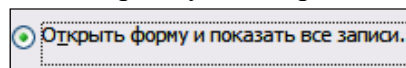


Рис. 21 Элементы управления

- Создайте заголовок формы. Для этого выберите кнопку щелкнув по ней, расширьте область заголовка формы и введите в поле надпись База данных «Врачи». Измените размер и цвет шрифта.
- Создайте заголовок формы. Для этого выберите кнопку 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*.
- Задайте имя кнопки *Список* и нажмите <Готово>.



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. 22):

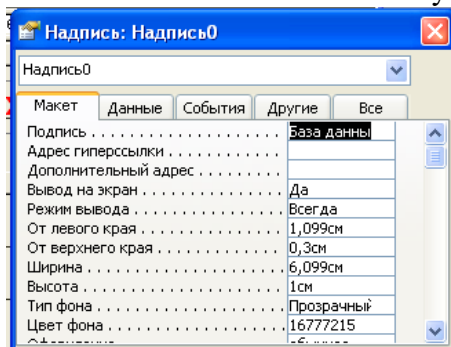


Рис.22. Окно свойств полей данных

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

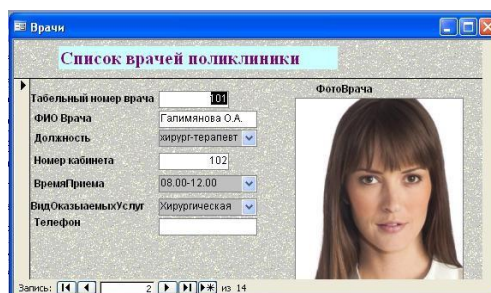
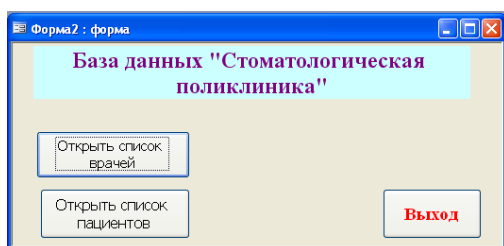


Рис. 23. Главная форма и форма «Врачи»

Создание многотабличной формы.

4. 3. 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 <ОК>.

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 *ШтатноеРасписание*: Должность, Оклад, Надбавка (рис. 24);

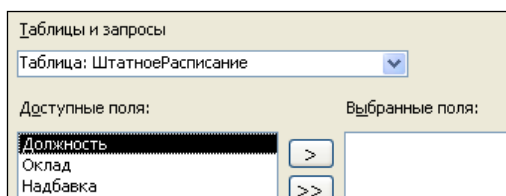


Рис.24. Выбор полей для формы посредством **Мастера**

- 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. 25): «id_Врача», «ФИО_Врача», «ВидОказываемыхУслуг» .
- Нажмите <Далее>.

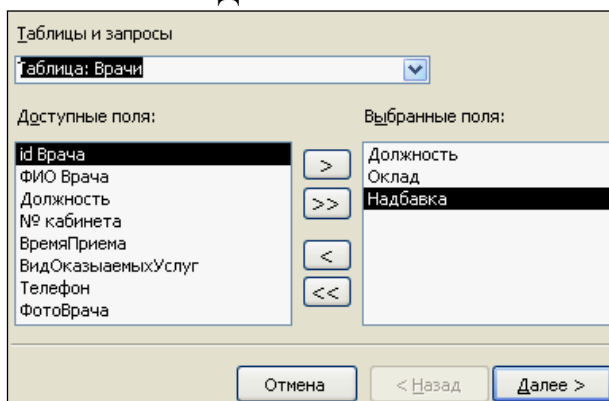


Рис.25. Выбор полей для подчиненной формы

- 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. 26).

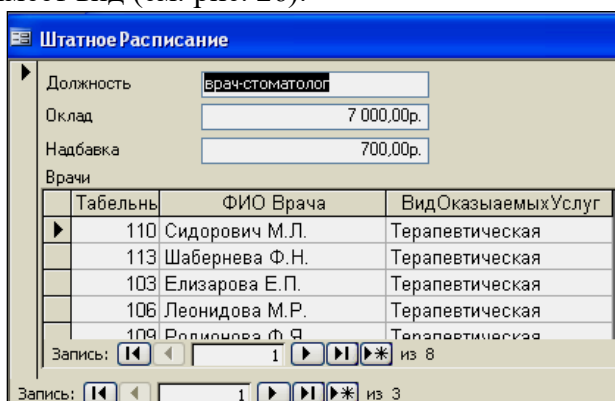


Рис. 26. Подчиненная форма

4. 4. Создание связанной формы.

- Для создания связанной формы в качестве таблицы-источника выбрать таблицу *СписокПациентов*: 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. 27).
- Измените дизайн форм, увеличив шрифт на **11pt**.

СписокПациентов1

Журнал1

НомерКарточки: 1

ФИО Пациента: Борисова Е.Р.

ДомашнийАдрес: Фучика, 83-15

МедПолис: АА-1234

Телефон: 12-13-14

Пол: жен.

Запись: 1 из 20

Журнал1

ДатаПриема	ФИО Врача	ДолжностьВрача	Услуги	Количество
01.01.20	Галимянова О.А.	врач-стоматолог	Удаление зубного	5
06.01.2011	Валиуллин Д.К.	ортопед	Коронка стальная	1
09.01.2011	Карбышев Г.О.	ортопед	Мостовидный прот	1

Запись: 1 из 3 (Фильтр)

Рис.27. Готовая связанная форма

Лабораторная работа № 5. Формирование запросов на выборку.

Цель работы: Научиться создавать запросы на выборку данных с помощью мастера и конструктора запросов.

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.
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» (порядок важен -в таком порядке данные и будут выводиться);
 - щелкните по кнопке. <Далее>;
 - в строке параметра «Сохранение запроса» введите новое имя «*Должности врачей*». <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
 - 3.1. Для создания запроса на выборку с параметром:

- создайте запрос на выборку для следующих полей таблицы «Врачи»: *ID Врача, ФИО врача, № кабинета, ВремяПриема*; 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.

3.2. 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*».

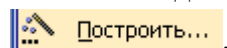
4. Создание запроса с помощью построителя выражения.

4.1. 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



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. 28).

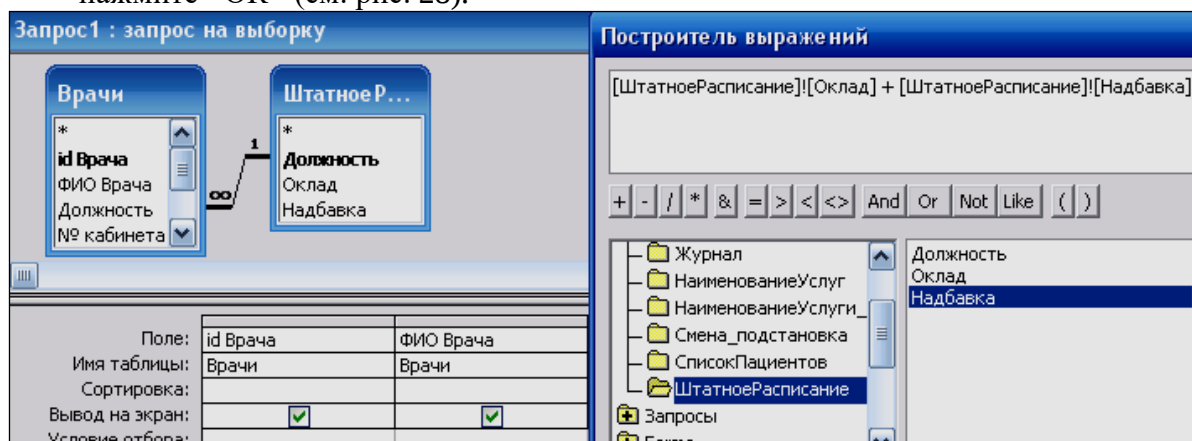
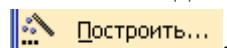


Рис. 28. Создание запроса с вычисляемым полем

- щелкните по кнопке ,
- сохраните запрос «Итого зарплата».
- в режиме Конструктора в строке **Поле:** выделите *Выражение1:* и переименуйте на «Итого:».

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

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



В появившемся диалоговом окне **Построитель выражения** выберите:

Функции → **Встроенные функции** → **Дата/время**. Щелкните два раза кнопкой 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> (см. рис. 29).

В этом же **столбце** в строке **Условие отбора** вести: [Введите номер месяца].

Нажмите кнопку .

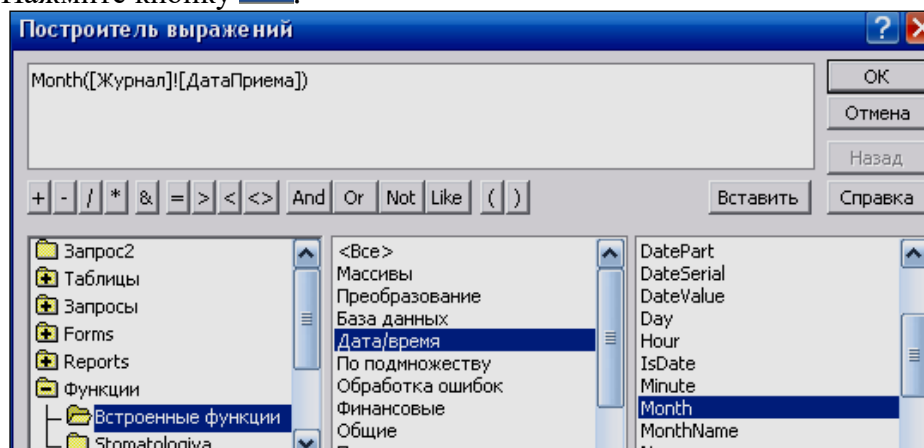


Рис. 29. Создание запроса

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

4.3. 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	<i>Year([Пациенты]![ДатаРождения])</i>
Условие отбора	<[Год рождения]

4.4. 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 строке «Групповая операция» выбрать из выпадающего списка **Sum**. Просмотреть запрос и проверить правильность, используя таблицу *Журнал* и фильтр записей.

Таблица: 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

Примечание:

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

4.5. Создайте запрос на подсчет количества врачей по должностям.

- 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**.

- В результате запроса должны получиться следующие данные:

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

Лабораторная работа № 6. Создание сводных таблиц и диаграмм.

Цель работы: ознакомление с возможностями работы табличного процессора со списком данных, приобретение навыков создания списков (базы данных) пользователя, свободных 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 (рис. 30):

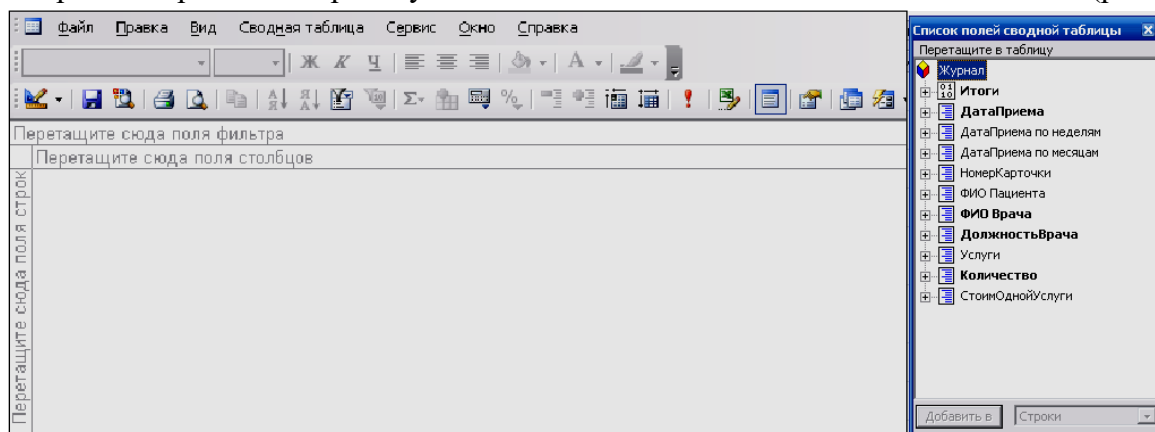


Рис. 30. Пустой макет сводной таблицы **Журнал** и список полей источника записей

Пусть необходимо подсчитать количество выполненных услуг каждым врачом за каждый день (за определенный период). Если список полей этой таблицы не отображается, нажмите на панели инструментов пиктограмму **Список полей**.

- Уникальные значения фамилии врачей составят заголовки строк. Перетащите в эту область поле **ФИО Врачей**.
- В область полей столбцов перетащите поле **ДатаПриема**.
- Пусть в нашем примере фильтрация составит по должностям. Перетащите в область фильтра – поле **ДолжностьВрача** (см. рис. 31).

ДолжностьВрача	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	Количество	Нет итогов
Валиуллин Д.К.	Количество	1	2				1		1	
Галимянова О.А.	Количество	5								
Дюкина Я.Б.	Количество	4								
Елизарова Е.П.	Количество					2	2			
Жуков А.Е.	Количество		2							
Карбышев Г.О.	Количество		2			3	1	2		
Леонидова М.Р.	Количество			3				2		
Накипов Н.Д.	Количество		2				1			
Никоненко П.С.	Количество				2	1				
Родионова Ф.Я.	Количество			1						
Сидорович М.Л.	Количество				4		1			
Халимов И.Ш.	Количество				2					1
Харитоновна Н.Н.	Количество					4				
Шабернева Ф.Н.	Количество						1			2
Общие итоги	Количество									

Рис. 31. Сводная таблица с фамилиями врачей и датами приема

- Для подсчета итоговых значений щелкните в области данных правой кнопкой мыши и выберите **Автовычисления** → **Сумма** (см. рис. 32).

ФИО Врача	Количество	Количество	Количество	Количество	Количество
Валиуллин Д.К.	1	2			
Галимянова О.А.	5				
Дюкина Я.Б.					
Елизарова Е.П.					
Жуков А.Е.					
Карбышев Г.О.					
Леонидова М.Р.					
Накипов Н.Д.					

Рис. 32. Формирование итоговых значений.

- В результате получим таблицу с итоговыми значениями (количество выполненных услуг) и суммарной величиной этих значений. (см. рис. 33).

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.	5						1		6
Дюкина Я.Б.	4								4
Елизарова Е.П.					2	2			4
Жуков А.Е.		2				2			4
Карбышев Г.О.		4					1		6
Леонидова М.Р.				3		3			5
Накипов Н.Д.		2		3					3
Никоненко П.С.		2			2	1			3
Родионова Ф.Я.				1					1
Сидорович М.Л.					4		1		8

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
- СтоимостьУслуги

Добавить в Данные

Рис. 33. 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**. (см.рис. 34).

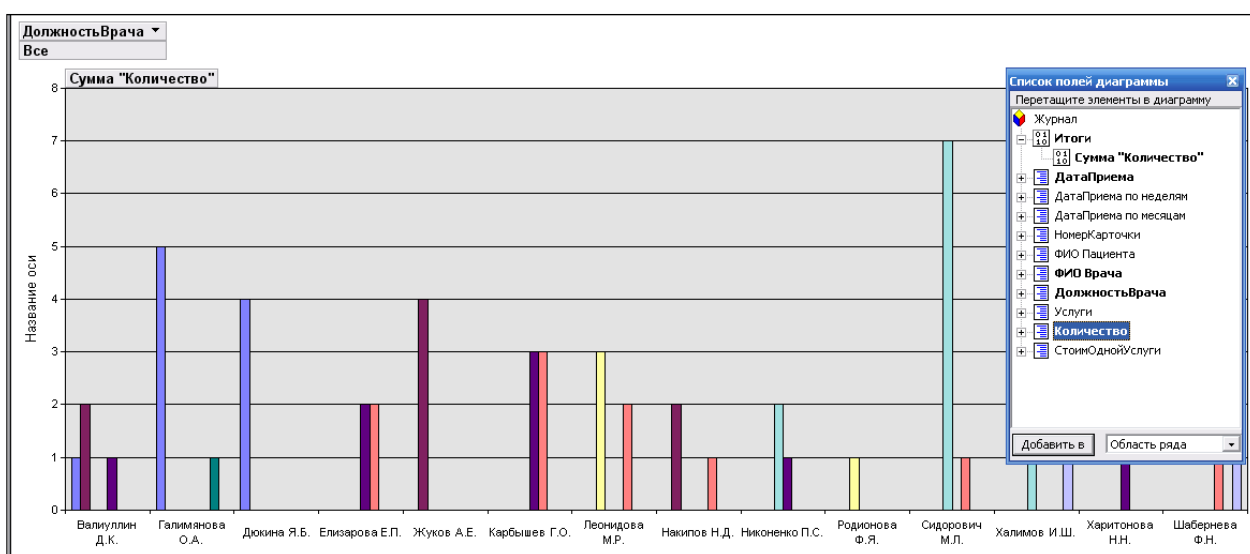


Рис. 34. Таблица *Журнал* в режиме сводной диаграммы.

Лабораторная работа № 7-8 Создание и настройка отчетов.

Цель работы: изучить среду проектирования баз данных в MS Access; создать отчеты для печати; ознакомиться со всеми типами отчетов; подготовить отчет по результатам лабораторной работы.

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**:

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

Пример готового отчета представлен на рисунке 35.

ШтатноеРасписание					
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	Харитонова 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. 35. Отчет

САМОСТОЯТЕЛЬНОЕ ЗАДАНИЕ. Создайте отчеты по выполненным запросам.

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.

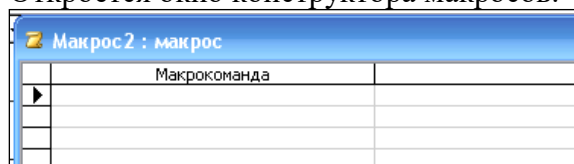


Рис. 36.Окно конструктора макросов.

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*. Затем выберите имя отчета и режим: *Просмотр*.

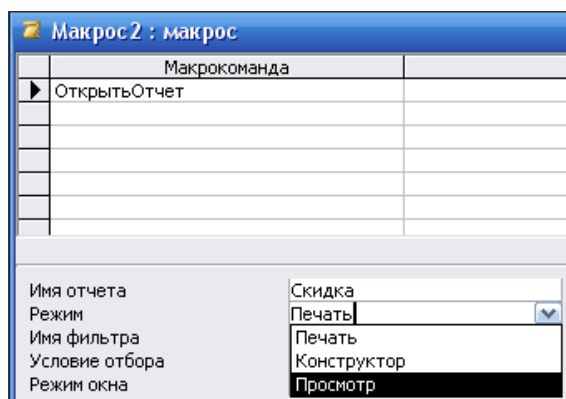


Рис. 37. Макрос.

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

САМОСТОЯТЕЛЬНОЕ ЗАДАНИЕ. Добавьте остальные кнопки, дополнив их соответствующими командами-макросами.

Задания на лабораторную работу

Выполните разработку программного комплекса в соответствии с требованиями, подтверждающими квалификацию разработчика прикладного программного обеспечения. Программный комплекс не должен терять работоспособности при некорректных действиях пользователя. В интерфейсе должны использоваться только термины, понятные пользователю. Появление англоязычных сообщений СУБД недопустимо. Язык диалога должен категорически исключить компьютерный сленг.

Вариант 1. Разработать прикладное программное обеспечение деятельности депо по ремонту пассажирских вагонов. Депо выполняет несколько видов ремонта. Депо ремонт – после пробега вагоном 450 тыс. км или два года эксплуатации (что наступит раньше). ТО-2 – подготовка вагона к зимним или летним условиям эксплуатации. ТО-3 – текущее обслуживание – после пробега 150 тыс. км или один год эксплуатации. Текущий ремонт – круглосуточно, при котором ремонтируются вагоны всех дорог России. Основные причины поступления вагона в текущий ремонт: неисправность колесной пары, неисправность буксового узла и т. д. Каждый вагон имеет уникальный номер. Тип вагона также имеет значение при ремонте: купейный, СВ, плацкартный, почтовый, багажный. Каждый вагон приписан к дирекции по обслуживанию пассажиров (ДОП-1, ДОП-2, ДОП-3 и т. д.). Текущий ремонт выполняют ремонтные бригады в четыре смены. Для выполнения остальных ремонтов привлекается, как правило, одна бригада. За высокое качество ремонта члены бригады получают премию.

Таблица 9.1

Набор данных к варианту 1

№	Поле	Тип	Размер	Описание
1	RegNumber	Числовой	10	Регистрационный номер вагона
2	RegName	Текстовый	60	Приписка вагона к дороге
3	RegChief	Текстовый	20	Приписка вагона к дирекции
4	Type	Текстовый	20	Тип вагона (купейный, СВ, и т. д.)
5	TypeYear	Числовой	4	Год выпуска вагона
6	TypeRepair	Текстовый	39	Тип ремонта
7	Picture	Поле OLE	Авто	Фотография вагона
8	Money	Денежный	15	Стоимость ремонта
9	Bonus	Логический	1	Качество ремонта (отличное/по нормам)
10	BonusPercent	Числовой	2	Премия в процентах (общая)
11	DateStart	Дата/Время	Авто	Начало ремонта
12	DateStop	Дата/Время	Авто	Окончание ремонта
13	Reason	Текстовый	40	Причина поступления в ремонт
14	External	Логический	1	Внешняя/Местная железная дорога
15	BankExternal	Текстовый	60	Банк внешней железной дороги
16	InnExternal	Числовой	10	ИНН внешней железной дороги
17	AddressExternal	Текстовый	80	Юридический адрес внешней ж/дороги
18	FIOchief	Текстовый	40	ФИО бригадира
19	Base	Текстовый	15	Образование бригадира (ВУЗ)
20	FIOworker	Текстовый	40	ФИО работника
21	BaseWorker	Числовой	15	Образование работника (ВУЗ)
22	YearWorker	Числовой	2	Стаж работы
23	SpecialWorker	Текстовый	30	Основная специальность работника
24	BonusWorker	Денежный	15	Премия в рублях работнику
25	Comment	Поле Мемо	Авто	Примечания (за что премия)
26	NumberBankKart	Текстовый	60	Номер карты для перечисления З/П

Вариант 2. Разработать прикладное программное обеспечение деятельности ремонтно-эксплуатационного локомотивного депо. Депо выполняет несколько видов ремонта: текущий ремонт (ТР), средний ремонт (СР), техническое обслуживание (ТО) и внеплановый ремонт. При внеплановом ремонте локомотив снимается с рейса и заменяется резервным, поэтому сроки внепланового ремонта должны быть минимальными, а сам ремонт проводится порой в четыре смены. Каждый локомотив имеет уникальный номер и приписан к определенному локомотивному депо. Технология ремонта зависит от типа локомотива (пассажирский или грузовой). Для выполнения первых трех видов ремонта привлекается, как правило, одна бригада. За высокое качество выполненных работ члены бригады получают дополнительное вознаграждение (квартальная премия, месячная премия, 13 и 14 зарплата). За переработку (сверхурочные) также выплачиваются дополнительные суммы.

Таблица 9.2

Набор данных к варианту 2

№	Поле	Тип	Размер	Описание
1	RegNumber	Числовой	10	Регистрационный номер локомотива
2	RegName	Текстовый	60	Приписка локомотива к депо
3	Kind	Текстовый	20	Марка локомотива (ВЛ-80с, ВЛ-80р и т. д.)
4	Type	Текстовый	20	Тип локомотива (грузовой, пассажирский)
5	TypeYear	Числовой	4	Год выпуска локомотива
6	TypeRepair	Текстовый	39	Тип ремонта
7	Picture	Поле OLE	Авто	Фотография локомотива
8	Money	Денежный	15	Стоимость ремонта
9	Bonus	Логический	1	Качество ремонта (отличное/по нормам)
10	BonusPercent	Числовой	2	Премия в процентах (общая)
11	DateStart	Дата/Время	Авто	Начало ремонта
12	DateStop	Дата/Время	Авто	Окончание ремонта
13	Reason	Текстовый	40	Причина поступления в ремонт
14	External	Логический	1	Внешнее/Местное депо
15	BankExternal	Текстовый	60	Банк внешнего депо
16	InnExternal	Числовой	10	ИНН внешнего депо
17	AddressExternal	Текстовый	80	Юридический адрес внешнего депо
18	FIOchief	Текстовый	40	ФИО бригадира
19	Base	Текстовый	15	Образование бригадира (ВУЗ)
20	FIOworker	Текстовый	40	ФИО работника
21	BaseWorker	Числовой	15	Образование работника (ВУЗ)
22	YearWorker	Числовой	2	Стаж работы
23	SpecialWorker	Текстовый	30	Основная специальность работника
24	BonusWorker	Денежный	15	Премия в рублях работнику
25	Comment	Поле Мемо	Авто	Примечания (за что премия)
26	NumberBankKart	Текстовый	60	Тип премии

Вариант 3. Разработать прикладное программное обеспечение деятельности судоходной компании «Балтика». Эта крупная компания занимается перевозками грузов между континентами. В ее собственности несколько десятков судов различного класса и грузоподъемности. К услугам этой компании обращаются тысячи клиентов из различных стран мира.

На судне может находиться несколько партий грузов для различных грузополучателей из различных стран и городов. Одна партия груза может состоять из нескольких разновидностей грузов. У одной партии груза может быть только один отправитель и только один получатель. Судно следует по маршруту. Маршрут разрабатывается главным менеджером компании и проходит через несколько портов. В очередном порту назначения производится лишь частичная погрузка и выгрузка грузов, и судно следует дальше.

Таблица 9.3

Набор данных к варианту 3

№	Поле	Тип	Размер	Описание
1	RegNumber	Числовой	10	Регистрационный номер судна
2	Name	Текстовый	60	Название судна
3	Skipper	Текстовый	60	ФИО капитана судна
4	Type	Текстовый	15	Тип судна (танкер, сухогруз)
5	Capacity	Числовой	10	Грузоподъемность судна
6	Year	Числовой	4	Год постройки судна
7	Picture	Поле OLE	Авто	Фотография судна
8	Dockyard	Текстовый	15	Порт приписки
9	CustomValue	Числовой	10	Таможенный номер партии груза
10	DepartureDate	Дата	Авто	Дата убытия груза
11	ArriveDate	Дата	Авто	Дата прибытия груза
12	Origin	Текстовый	20	Пункт отправления
13	Destination	Текстовый	20	Пункт назначения
14	CustomClearance	Логический	1	Необходимость таможенной декларации
15	Number	Числовой	4	Номер груза в партии
16	Shipment	Текстовый	30	Название груза
17	DeclareValue	Числовой	8	Заявленная величина груза
18	Unit	Текстовый	10	Единица измерения груза
19	InsureValue	Числовой	8	Застрахованная величина груза
20	Sender	Текстовый	30	Отправитель груза
21	INNsender	Числовой	10	ИНН отправителя груза
22	BankSender	Текстовый	60	Банк отправителя груза
23	AddressSender	Текстовый	80	Юридический адрес отправителя груза
24	Consignee	Текстовый	30	Получатель груза
25	INNconsignee	Числовой	10	ИНН получателя груза
26	BankConsignee	Текстовый	60	Банк получателя груза
27	AddressConsign	Текстовый	80	Юридический адрес получателя груза
28	Comment	Поле Мемо	Авто	Примечания

Вариант 4. Разработать прикладное программное обеспечение деятельности учреждения юстиции. По существующему законодательству на это учреждение возложена обязанность регистрации прав юридических и физических лиц на недвижимое имущество (здания, квартиры, земельные участки). В этом задании вам необходимо разработать лишь часть программного комплекса, обеспечивающего регистрацию прав граждан на квартиры. Имейте в виду! В здании несколько квартир. В одной квартире – несколько собственников, причем в базе данных должна храниться история перехода квартиры от одних собственников к другим. Кадастровый номер здания однозначно определяет его среди других зданий города. Смело используйте его в качестве первичного ключа таблицы зданий.

Таблица 9.4

Набор данных к варианту 4

№	Поле	Тип	Размер	Описание
1	Kadastr	Текстовый	20	Кадастровый номер здания
2	Address	Текстовый	60	Адрес здания
3	District	Текстовый	15	Район города
4	Land	Числовой	10	Площадь земельного участка
5	Year	Числовой	4	Год постройки здания
6	Material	Текстовый	15	Материал стен здания
7	Base	Текстовый	15	Материал фундамента
8	Comment	Поле Мемо	Авто	Примечания
9	Wear	Числовой	2	Износ в процентах
10	Flow	Числовой	2	Число этажей в здании
11	Line	Числовой	5	Расстояние от центра города
12	Square	Числовой	10	Площадь нежилых помещений
13	Picture	Поле OLE	Авто	Фото здания
14	Flats	Числовой	3	Количество квартир в здании
15	Elevator	Логический	1	Наличие лифта
16	Flat	Числовой	4	Номер квартиры
17	Storey	Числовой	2	Номер этажа
18	Rooms	Числовой	1	Количество комнат
19	SquareFlat	Числовой	Авто	Общая площадь квартиры
20	Dwell	Числовой	Авто	Жилая площадь квартиры
21	Branch	Числовой	Авто	Вспомогательная площадь квартиры
22	Balcony	Числовой	Авто	Площадь балкона
23	Height	Числовой	Авто	Высота квартиры
24	Record	Числовой	2	Номер записи о праве собственности
25	Document	Текстовый	60	Документ на право собственности
26	DateDoc	Дата	Авто	Дата документа о собственности
27	FioHost	Текстовый	60	Ф.И.О. собственника
28	Passport	Поле Мемо	Авто	Данные его паспорта
29	Part	Числовой	Авто	Принадлежащая ему доля, %
30	Born	Числовой	4	Год рождения собственника

Вариант 5. Разработать прикладное программное обеспечение деятельности малого научно-внедренческого предприятия «Квадро». Это предприятие занимается прокладкой компьютерных сетей и разработкой программных комплексов для организаций нашего города.

Численность работников в «Квадро» – примерно 80 человек. Одновременно находится в разработке до 30 проектов. Один разработчик может участвовать в нескольких проектах одновременно, но зарплата его от этого не зависит. Одна организация может заказать в «Квадро» несколько разработок. Стоимость каждого проекта оговаривается отдельно. При досрочном выполнении работы заказчик перечисляет научно-внедренческому предприятию определенный, заранее оговоренный процент премии.

Таблица 9.5

Набор данных к варианту 5

№	Поле	Тип	Размер	Описание
1	EmployeeID	Числовой	3	Идентификатор работника
2	EmployeeName	Текстовый	60	ФИО работника
3	Address	Текстовый	60	Домашний адрес
4	District	Текстовый	15	Район города
5	Experience	Числовой	2	Опыт работы по специальности
6	Year	Числовой	4	Год рождения
7	Language	Текстовый	15	Базовый язык программирования
8	Base	Текстовый	15	Образование (вуз)
9	Comment	Поле Мемо	Авто	Примечания
10	Salary	Денежный	15	Зарплата
11	Bonus	Денежный	15	Премия
12	GrossSalary	Денежный	15	Полная зарплата
13	Exempt	Денежный	15	Льготы
14	Picture	Поле OLE	Авто	Фото работника
15	ProjectID	Числовой	3	Идентификатор проекта
16	ProjectName	Текстовый	40	Название проекта
17	ProjectStart	Дата	Авто	Дата начала проекта
18	ProjectStop	Дата	Авто	Дата окончания проекта
19	Chief	Текстовый	60	Руководитель проекта
20	Customer	Текстовый	60	Заказчик проекта
21	Cost	Числовой	Авто	Стоимость разработки
22	Phone	Текстовый	10	Телефон заказчика
23	Bank	Текстовый	60	Банк заказчика
24	Account	Текстовый	20	Номер счета в банке
25	INN	Текстовый	10	ИНН заказчика
26	AddressCust	Текстовый	60	Адрес заказчика
27	FioWorker	Текстовый	60	Ответственный от заказчика
28	PhoneWorker	Текстовый	10	Телефон ответственного
29	BonusAll	Числовой	Авто	Премия, %, при досрочном выполнении
30	EmployeeStart	Дата	Авто	Начало участия работника в проекте
31	EmployeeStop	Дата	Авто	Конец участия работника в проекте

Вариант 6. Разработать прикладное программное обеспечение деятельности ООО «Киновидеопрокат». Это предприятие фактически контролирует демонстрацию кинофильмов в кинотеатрах города. Отдел маркетинга, изучив ситуацию на рынке кинофильмов, принимает решение о покупке тех или иных кинолент. Отдел закупок претворяет эти решения в жизнь, причем лента может быть куплена как у производителя, так и у посредника.

Отдел аренды киновидеопроката сдает закупленные фильмы кинотеатрам города в аренду. Так как всегда закупается только одна копия фильма, он не может демонстрироваться одновременно в нескольких кинотеатрах. У одного поставщика может быть куплено несколько фильмов. Также несколько лент может быть в аренде у одного кинотеатра одновременно.

Таблица 9.6

Набор данных к варианту 6

№	Поле	Тип	Размер	Описание
1	Provider	Текстовый	40	Поставщик кинофильма
2	INN	Текстовый	10	ИНН поставщика кинофильма
3	Address	Текстовый	60	Юридический адрес поставщика
4	Bank	Текстовый	60	Банк поставщика кинофильма
5	Account	Текстовый	20	Номер счета в банке
6	Sign	Логический	1	Признак посредника
7	Film	Текстовый	20	Название кинофильма
8	Script	Текстовый	60	Автор сценария
9	Comment	Поле Мемо	Авто	Краткое содержание фильма
10	Producer	Текстовый	60	Режиссер-постановщик
11	Company	Текстовый	40	Компания-производитель
12	Year	Числовой	4	Год выхода на экран
13	Expense	Денежный	15	Затраты на производство
14	Cost	Денежный	15	Стоимость приобретения
15	Translate	Логический	1	Наличие дублирования
16	Cinema	Текстовый	20	Название кинотеатра
17	INNcinema	Текстовый	10	ИНН кинотеатра
18	AddressCinema	Текстовый	60	Адрес кинотеатра
19	Chief	Текстовый	60	Директор кинотеатра
20	Owner	Текстовый	60	Владелец кинотеатра
21	BankCinema	Текстовый	60	Банк кинотеатра
22	Phone	Текстовый	10	Телефон кинотеатра
23	District	Текстовый	15	Район города
24	AccountCinema	Текстовый	20	Номер счета кинотеатра в банке
25	Capacity	Числовой	4	Число посадочных мест
26	DateStart	Дата	Авто	Дата начала демонстрации фильма
27	DateStop	Дата	Авто	Окончание демонстрации
28	PhoneWorker	Текстовый	10	Телефон ответственного
29	Worker	Текстовый	60	Ответственный от кинотеатра
30	Summa	Денежный	15	Сумма оплаты за аренду ленты
31	Tax	Денежный	15	Пени за несвоевременный возврат

Вариант 7. Разработать прикладное программное обеспечение деятельности предприятия LADA-сервис. Эта крупная компания занимается продажей автомобилей марки ВАЗ в нашем городе. Она имеет несколько филиалов в разных районах. Автомобиль может быть продан как со склада компании, так и на заказ с завода-изготовителя по предоплате. Покупатель может заказать модель, цвет, тюнинг и оговорить срок поставки заказанного автомобиля. Одновременно с новыми авто на площадках компании имеется большой выбор подержанных автомобилей, как отечественных, так и иностранных. Покупателем может быть как физическое лицо, так и организация. В первом случае – расчет наличными, во втором – через банк. Расчет производится в рублях.

Таблица 9.7

Набор данных к варианту 7

№	Поле	Тип	Размер	Описание
1	IDfilial	Числовой	1	Регистрационный номер филиала
2	Filial	Текстовый	20	Название филиала предприятия
3	InnFilial	Текстовый	10	ИНН филиала предприятия
4	Chief	Текстовый	60	Руководитель филиала
5	Capacity	Числовой	3	Число стояночных мест на площадке
6	Address	Текстовый	60	Адрес филиала предприятия
7	Phone	Текстовый	10	Номер телефона филиала
8	Brand	Текстовый	15	Марка автомобиля
9	Model	Текстовый	15	Модель автомобиля
10	BodyID	Текстовый	20	Номер кузова
11	EngineID	Текстовый	20	Номер двигателя
12	BodyModel	Текстовый	20	Модель кузова
13	Picture	Поле OLE	Авто	Фотография автомобиля
14	Volume	Числовой	5	Объем двигателя
15	Power	Числовой	3	Мощность двигателя, л.с.
16	Helm	Логический	1	Руль (правый/левый)
17	Drive	Логический	1	Привод на все колеса
18	DateStart	Дата	Авто	Дата появления в продаже
19	Cost	Денежный	15	Стоимость автомобиля
20	New	Логический	1	Новый/подержанный
21	Year	Числовой	4	Год выпуска автомобиля
22	Distance	Числовой	6	Пробег автомобиля, км
23	Type	Текстовый	15	Тип кузова автомобиля
24	Client	Текстовый	60	Покупатель автомобиля
25	Sign	Логический	1	Признак покупателя (юр/физ. лицо)
26	Bank	Текстовый	60	Банк покупателя
27	Account	Текстовый	20	Номер счета в банке
28	Comment	Поле Мемо	Авто	Примечания
29	Customer	Текстовый	60	Заказчик
30	Price	Денежный	15	Стоимость заказанного автомобиля
31	StartDate	Дата	Авто	Дата заказа

Вариант 8. Разработать прикладное программное обеспечение торго- во-посреднической фирмы «Столица». Бизнес этого предприятия предель- но прост: «покупай дешевле – продавай дороже», или состыкуй продавца и покупателя и получи «комиссионные». Основной упор фирма делает на закупки продуктов питания в других регионах страны и за рубежом – там, где они производятся и стоят дешевле, чем в нашем регионе.

Часть продукции может быть закуплена и у местных продавцов. В этом случае фирма получает прибыль за счет того, что крупные партии товара стоят дешевле, чем мелкие.

Имейте в виду, что товар не может быть продан дешевле, чем онкуплен.

Таблица 9.8

Набор данных к варианту 8

№	Поле	Тип	Размер	Описание
1	Seller	Текстовый	60	Фирма – продавец товара
2	InnSeller	Текстовый	10	ИНН продавца
3	Country	Текстовый	15	Страна продавца
4	Chief	Числовой	60	Руководитель фирмы
5	Address	Числовой	60	Юридический адрес фирмы
6	Phone	Текстовый	10	Телефон руководителя
7	Manager	Текстовый	60	Главный менеджер фирмы
8	PhonePlus	Текстовый	10	Телефон отдела продаж
9	Bank	Текстовый	60	Банк продавца
10	Account	Текстовый	20	Номер счета в банке
11	GoodsID	Числовой	10	Штрих-код товара
12	Goods	Текстовый	30	Название товара
13	Picture	Поле OLE	Авто	Фото товара
14	Category	Текстовый	15	Категория товара (кофе, печенье)
15	DateStart	Дата	Авто	Дата изготовления товара
16	Period	Числовой	4	Срок хранения товара, дн.
17	Manufacturer	Текстовый	60	Изготовитель товара
18	Unit	Текстовый	10	Единица измерения
19	CostUnit	Денежный	15	Цена за единицу
20	Count	Числовой	Авто	Количество товара
21	Client	Текстовый	60	Покупатель товара
22	InnClient	Текстовый	10	ИНН покупателя
23	Director	Текстовый	60	Руководитель фирмы-покупателя
24	PhoneDir	Текстовый	10	Телефон директора
25	AddressClient	Текстовый	60	Юридический адрес фирмы
26	BankClient	Текстовый	60	Банк покупателя
27	AccountClient	Текстовый	20	Номер счета в банке
28	Volume	Числовой	Авто	Количество купленного товара
29	CostUnitVol	Денежный	15	Цена за единицу
30	DateVolume	Дата	Авто	Дата покупки товара
31	Comment	Поле Мемо	Авто	Примечания

Вариант 9. Разработать прикладное программное обеспечение деятельности отдела гарантийного ремонта товаров фирмы «Народная торговая компания». Это предприятие – лидер продаж кондиционеров, телевизоров и другой бытовой техники в городе. Хорошо известно, что техника часто выходит из строя, причем уже в период гарантийного срока, а в этом случае продавец товара должен бесплатно отремонтировать его. Ежедневно в отдел гарантийного ремонта обращаются несколько десятков человек, купивших технику в этой компании. Вы, скорее всего, также побывали в отделе гарантийного ремонта, что очень поможет Вам при разработке программного обеспечения.

Таблица 9.9

Набор данных к варианту 9

№	Поле	Тип	Размер	Описание
1	IDfilial	Числовой	1	Регистрационный номер филиала
2	Filial	Текстовый	20	Название филиала предприятия
3	InnFilial	Текстовый	10	ИНН филиала предприятия
4	Chief	Текстовый	60	Руководитель филиала
5	Capacity	Числовой	3	Количество работающих на ремонте
6	Address	Текстовый	60	Адрес филиала предприятия
7	Phone	Текстовый	10	Номер телефона филиала
8	GoodsID	Текстовый	15	Штрих-код товара
9	Goods	Текстовый	40	Название товара или прибора
10	Category	Текстовый	20	Категория (утюг, миксер)
11	Country	Текстовый	20	Страна-производитель
12	Company	Текстовый	40	Изготовитель
13	Picture	Поле OLE	Авто	Фотография товара или прибора
14	INNcompany	Текстовый	10	ИНН изготовителя
15	AddressComp	Текстовый	60	Адрес изготовителя
16	DateStart	Дата	Авто	Дата изготовления товара
17	Period	Числовой	4	Гарантийный период
18	DateBuy	Дата	Авто	Дата покупки
19	Cost	Денежный	15	Стоимость товара
20	Fax	Текстовый	12	Номер факса компании
21	PhoneCompany	Текстовый	12	Телефон компании
22	Email	Текстовый	20	Адрес электронной почты компании
23	Web	Текстовый	20	Адрес WEB-страницы
24	CostRepair	Денежный	15	Стоимость ремонта
25	CustomerID	Числовой	5	Идентификатор покупателя
26	Customer	Текстовый	60	Покупатель
27	AddressCust	Текстовый	60	Адрес покупателя
28	Comment	Поле Memo	Авто	Примечания (что было сделано)
29	Sign	Логический	1	Признак покупателя (юр/физ. лицо)
30	Guarantee	Числовой	5	Оставшийся гарантийный срок
31	StartDate	Дата	Авто	Дата приемки в ремонт
32	StopDate	Дата	Авто	Дата получения

Вариант 10. Разработать прикладное программное обеспечение деятельности отдела учета личного состава батальона железнодорожных войск. Это фактически отдел кадров воинской части. Батальон расквартирован на отдельной территории. В батальоне несколько рот, в каждой роте несколько взводов, каждый взвод состоит из трех отделений. В мирное время батальон занимается изучением техники и поддержанием ее в рабочем состоянии. Часть технических ресурсов «законсервирована». Поддержание такой техники в отличном состоянии также входит в обязанности личного состава батальона. В настоящее время существует три вида службы: срочная, сверхсрочная и по контракту. Каждый офицер части имеет удостоверение личности, которое заменяет паспорт, а военнослужащий срочной службы – военный билет.

Таблица 9.10

Набор данных к варианту 10

№	Поле	Тип	Размер	Описание
1	Number	Текстовый	15	Номер воинской части
2	Battalion	Текстовый	30	Название батальона
3	Commander	Текстовый	50	ФИО командира батальона
4	Rank	Текстовый	20	Воинское звание командира
5	ViceCommander	Текстовый	50	ФИО зам. командира батальона
6	Rank2	Текстовый	20	Воинское звание зам. командира
7	Photo	Поле OLE	Авто	Фото командира батальона
8	Commander2	Текстовый	50	ФИО командира роты
9	Rank3	Текстовый	20	Звание командира роты
10	Passport	Текстовый	20	Удостоверение личности
11	CompanyID	Числовой	1	Номер роты
12	CompanyName	Текстовый	20	Название роты
13	Photo2	Поле OLE	Авто	Фото командира роты
14	PlatoonID	Числовой	1	Номер взвода
15	Commander3	Текстовый	50	ФИО командира взвода
16	Rank4	Текстовый	20	Звание командира взвода
17	Photo3	Поле OLE	Авто	Фото командира взвода
18	PlatoonName	Текстовый	20	Название взвода
19	DepartmentID	Числовой	1	Номер отделения
20	Soldier	Текстовый	50	ФИО военнослужащего
21	Post	Текстовый	20	Должность (командир/солдат)
22	Rank5	Текстовый	20	Звание военнослужащего
23	Start	Дата	Авто	Начало службы
24	Stop	Дата	Авто	Конец службы
25	Kind	Логический	1	Вид службы (срочная или контрактная)
26	Address	Текстовый	60	Адрес проживания
27	Phone	Текстовый	15	Домашний телефон
28	Comment	Поле Memo	Авто	Примечание
29	Birth	Числовой	4	Год рождения

Список литературы

1. Свиридова, М. Ю. Система управления базами данных Access / М.Ю. Свиридова. - М.: Академия, 2016.
2. Персианов, Вячеслав Венедиктович; Технология Проектирования Информационной Базы Для Педагогических Вузов Страны. / Персианов Вячеслав Венедиктович;. - Москва: Огни, 2016
3. Кузнецов, С. Д. Основы баз данных / С.Д. Кузнецов. - М.: Бином. Лаборатория знаний, Интернет-университет информационных технологий, 2017
4. Латыпова, Р. Р. Базы данных. Курс лекций / Р.Р. Латыпова. - Москва: Высшая школа, 2016.
5. Фуфаев, Э. В. Базы данных / Э.В. Фуфаев, Д.Э. Фуфаев. - М.: Академия, 2016.