

Министерство сельского хозяйства Российской Федерации
Департамент научно-технологической политики и образования
*Федеральное государственное бюджетное образовательное
учреждение высшего образования*
«Красноярский государственный аграрный университет»

Титовская Н.В., Титовский С.Н.

**ПРОЕКТИРОВАНИЕ БАЗ ДАННЫХ В СУБД MICROSOFT
OFFICE ACCESS 2007**

Методические указания к лабораторным работам
Электронное издание

ФГОС ВО

Направление подготовки 09.03.03 «Прикладная
информатика», 01.03.02 «Прикладная математика и
информатика»

Красноярск 2018

ББК 32.973я73
Т45

Рецензент

С.А. Бронов, д-р техн. наук, профессор кафедры «Информационные технологии и математическое обеспечение информационных систем»

Титовская, Н.В., Титовский С.Н.

Проектирование баз данных в СУБД Microsoft Office Access 2007: метод. указания / Н.В.Титовская, С.Н.Титовский, Краснояр. гос. аграр. ун-т. – Красноярск, 2018. – 83 с.

Методические указания предназначены для проведения практических занятий со студентами, изучающих программу управления базами данных ACCESS. Подробно рассматриваются вопросы по созданию базы данных: проектирование информационно-логической модели предметной области (определение предметной области, выделение информационных объектов и определение ключевых реквизитов), создание таблиц, определение связей между информационными объектами, проектирование логической структуры базы данных, ввод данных, создание форм, способы создания и подключения макросов, все виды запросов, отчеты.

Подробно разбирается создание кнопочной формы и интерфейса базы данных.

Методические указания помогут организовать дифференцированную и самостоятельную работу студентов по созданию базы данных.

Предназначены для студентов обучающихся по направлению подготовки 09.03.03 «Прикладная информатика», 01.03.02 «Прикладная математика и информатика»

Печатается по решению редакционно-издательского совета
Красноярского государственного аграрного университета

© Титовская Н.В., Титовский С.Н., 2018
© ФГБОУ ВО «Красноярский государственный
аграрный университет», 2018

ВВЕДЕНИЕ	6
1. Разработка базы данных «Ученики»	8
1.1 Постановка задачи	8
1.2 Проектирование информационно-логической модели предметной области	8
1.2.1 Определение предметной области	8
1.2.2 Выделение информационных объектов и определение ключевых реквизитов	8
1.2.3 Определение связей между информационными объектами ...	9
1.3 Проектирование логической структуры базы данных.....	9
1.3.1 Разработка структуры реляционных таблиц: определение имен, типов и размеров полей в таблицах	9
1.3.2 Установление связей между таблицами.....	10
1.4 Создание интерфейса базы данных средствами СУБД MS Access 2007	10
1.4.1 Создание таблиц.....	12
1.4.2 Проектирование форм.....	14
1.4.3 Ввод данных	24
1.4.4 Запросы	25
1.4.5 Отчеты	29
1.4.6 Кнопочные формы	34
2. Проектирование базы данных «Прокат фильмов».....	37
2.1 Проектирование логической структуры БД.....	37
2.2 Создание БД	38
2.3 Создание таблиц.....	38
2.4 Установка связей между таблицами	40
2.5 Проектирование форм.....	42
2.6 Макросы.....	55
2.6.1 Процедура создания макроса.....	56
2.6.2 Подключение макросов.....	57
2.7 Запросы	60
2.7.1 Создание запроса на выборку	61
2.7.2 Параметрический запрос	63
2.7.3 Перекрёстный запрос	64
2.8 Отчеты	65
2.9 Главная кнопочная форма.....	66
3. Глоссарий	70

4. РЕКОМЕНДУЕМАЯ ЛИТЕРАТУРА.....	75
Приложение 1.....	76
Задания для доработки базы данных «Прокат фильмов».....	76
Задание 1.....	76
Приложение 2.....	76
Разноуровневые задания для самостоятельной проектной работы .	76
Задача № 1	76
Задача № 2	77
Задача № 3	78
Задача № 4	78
Задача № 5	79
Задача № 6	80
Задача № 7	80
Задача № 8	81
Список вопросов по базам данных	82

ВВЕДЕНИЕ

В настоящее время жизнь человека настолько сильно насыщена различного рода информацией, что для управления ею требуется создание огромного количества баз и банков данных различного назначения. Сейчас практически любая задача связана с манипуляцией информацией и данными. По этой причине в последние годы появилось множество различных компьютерных систем – систем управления базами данных, предназначенных для этих целей.

В предлагаемых методических указаниях рассматриваются вопросы теории проектирования баз данных и особенности разработки пользовательских приложений на основе СУБД Microsoft Access. В указаниях подробно рассматриваются вопросы по созданию базы данных: проектирование информационно-логической модели предметной области (определение предметной области, выделение информационных объектов и определение ключевых реквизитов), создание таблиц, определение связей между информационными объектами, проектирование логической структуры базы данных, ввод данных, создание форм, способы создания и подключения макросов, все виды запросов, отчеты.

Подробно разбирается создание кнопочной формы и интерфейса базы данных.

Методические указания помогут организовать дифференцированную и самостоятельную работу студентов по созданию базы данных.

В «Методические материалы по выполнению практических работ» включены:

- проектирование базы данных школы «Ученики» (состоящей из одной основной и одной вспомогательной таблицы);
- проектирование базы данных «Прокат фильмов» (состоящей из трех таблиц);
- разноуровневые задания для самостоятельной проектной работы;
- глоссарий, в который включены определения, требующиеся при изучении дисциплины Базы данных;

➤ вопросы для проработки теоретических аспектов по базам данных.

В настоящих методических указаниях особое внимание уделено вопросам, позволяющим сделать работу пользователя с базой данных удобной и лёгкой. Кроме того, студентам предлагается самостоятельно усовершенствовать созданную базу данных.

Предлагаемый материал не разбит по занятиям, что предполагает работу в индивидуальном темпе и доработку во время, отведенного для самостоятельной работы.

В указаниях представлены базы данных разной степени сложности, что предполагает изучение материала «от простого к более сложному».

Изучающие материал могут проявить собственную фантазию и создать новые формы, спроектировать новые запросы, добавить информационные объекты к рассмотренной предметной области, используя приемы, разобранные в методических указаниях. По мнению авторов, такой подход будет способствовать закреплению материала, более осознанному пониманию возможностей технологии и поможет при постановке собственной задачи в конкретной предметной области.

Особо надо отметить, что работа снабжена большим количеством экранных форм, которые облегчают изучение материала и построение баз данных. Студенты найдут в указаниях подробные описания проблем, с которыми сталкивается пользователь, работающий в СУБД Access 2007. В каждом параграфе изложена формулировка задачи, методика ее решения и ожидаемый результат. Все эти элементы снабжены иллюстрациями, которые помогают быстро понять суть. Методическое указание хорошо структурировано и снабжено подробным оглавлением, оно поможет получить достаточные для самостоятельной работы навыки и умения при работе с базами данных.

1. РАЗРАБОТКА БАЗЫ ДАННЫХ «УЧЕНИКИ»

1.1 Постановка задачи

Разработать базу данных «Ученики»:

1. Разработать карточку учета учеников класса, в которой должны храниться следующие сведения:

- фамилия;
- имя;
- отчество;
- дата рождения;
- класс;
- контактный телефон;
- место жительства;
- сведения о родителях.

Ученики учатся в 5а, 5б, 6а, 6б, 7а, 7б, 8а, 8б, 9а, 9б, 10а, 11а классах.

2. Создать запрос на выборку всех учеников.

3. Создать запрос на выборку из списка учеников по определенным признакам.

4. Создать отчет по всему списочному составу классов.

5. Создать отчет о распределении учеников по классам.

1.2 Проектирование информационно-логической модели предметной области

1.2.1 Определение предметной области

Предметной областью является учет учеников школы.

Входная информация: сведения об учениках.

Выходная информация: список учеников и выборки по признакам.

1.2.2 Выделение информационных объектов и определение ключевых реквизитов

Имеется два информационных объекта и соответствующие таблицы: «УЧЕНИКИ» и «КЛАССЫ».

«УЧЕНИКИ» – основной объект, **«КЛАССЫ»** – вспомогательный (служит для заполнения соответствующего реквизита в основной таблице).

В объекте **«УЧЕНИКИ»** нет реквизита, который бы однозначно определял (идентифицировал) учеников в списке, а любые описательные реквизиты могут совпадать. Поэтому необходимо добавить еще один реквизит, который обозначим как ключевой и присвоим ему имя **«Код Ученика»**. В объекте **«КЛАССЫ»** ключевым реквизитом может служить название класса.

1.2.3 Определение связей между информационными объектами

Поставленные условия для создания БД характеризуют данную задачу как простую, потому что не существует конкретной функциональной взаимосвязи между выделенными информационными объектами, так как ученик не может учиться в разных классах одновременно (тип связи один-ко-многим).

1.3 Проектирование логической структуры базы данных

1.3.1 Разработка структуры реляционных таблиц: определение имен, типов и размеров полей в таблицах

При описании полей таблицы необходимо определить тип данных в поле и размер (количество символов) поля.

Определяем тип и размер всех входящих в таблицу полей. В результате получаем графическую интерпретацию выделенных информационных объектов с их параметрами.

Описание полей таблицы **«УЧЕНИКИ»**:

Ключ	Имя поля	Тип данных	Размер
⇒	Код ученика	Текстовый	3
	Фамилия	Текстовый	30
	Имя	Текстовый	20

	Отчество	Текстовый	20
	День Рождения	Дата/Время	Краткий формат даты
	Класс	Текстовый	4
	Место Жительства	Текстовый	50
	Сведения о	Текстовый	50

Описание полей таблицы «**КЛАССЫ**»:

Ключ	Имя поля	Тип данных	Размер
⇒	Класс	Текстовый	4

1.3.2 Установление связей между таблицами

Так как установлено, что конкретная взаимосвязь между таблицами отсутствует, таблицу «**КЛАССЫ**» можно использовать только для формирования «поля со списком».

1.4 Создание интерфейса базы данных средствами СУБД MS Access 2007

1. Запускаем программную оболочку MS Access 2007.

2. «Файл» - «Создать» - «Новая база данных». В раскрывшемся диалоговом окне определяем имя БД, например, «Ученики» и ее местоположение, затем нажимаем кнопку «Создать» (рис. 1).

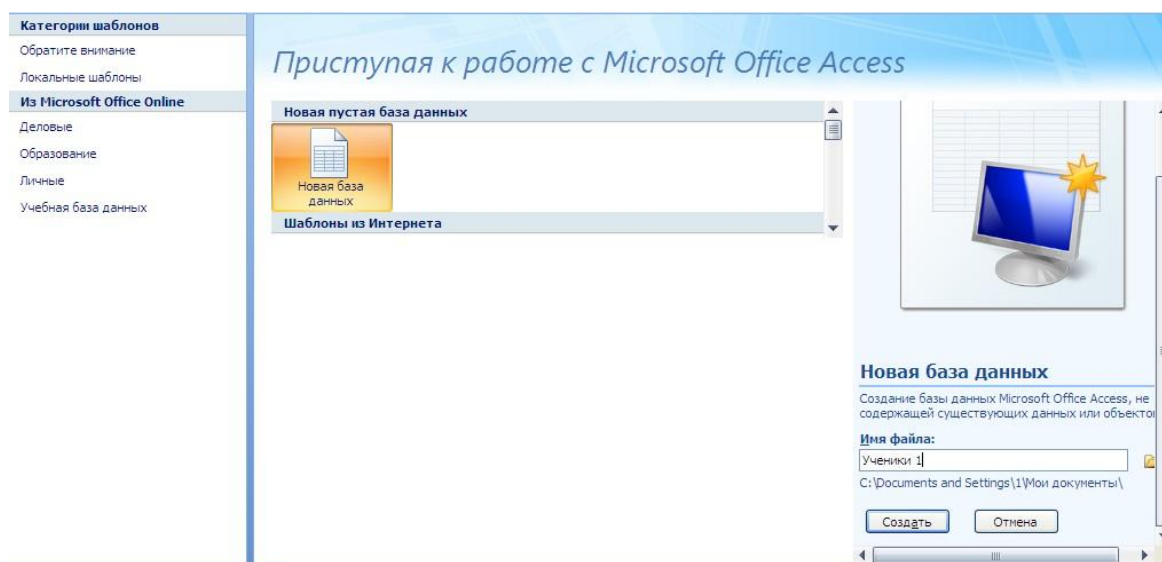


Рис. 1

3. Открывается основное рабочее окно для создания БД (Рис. 2):

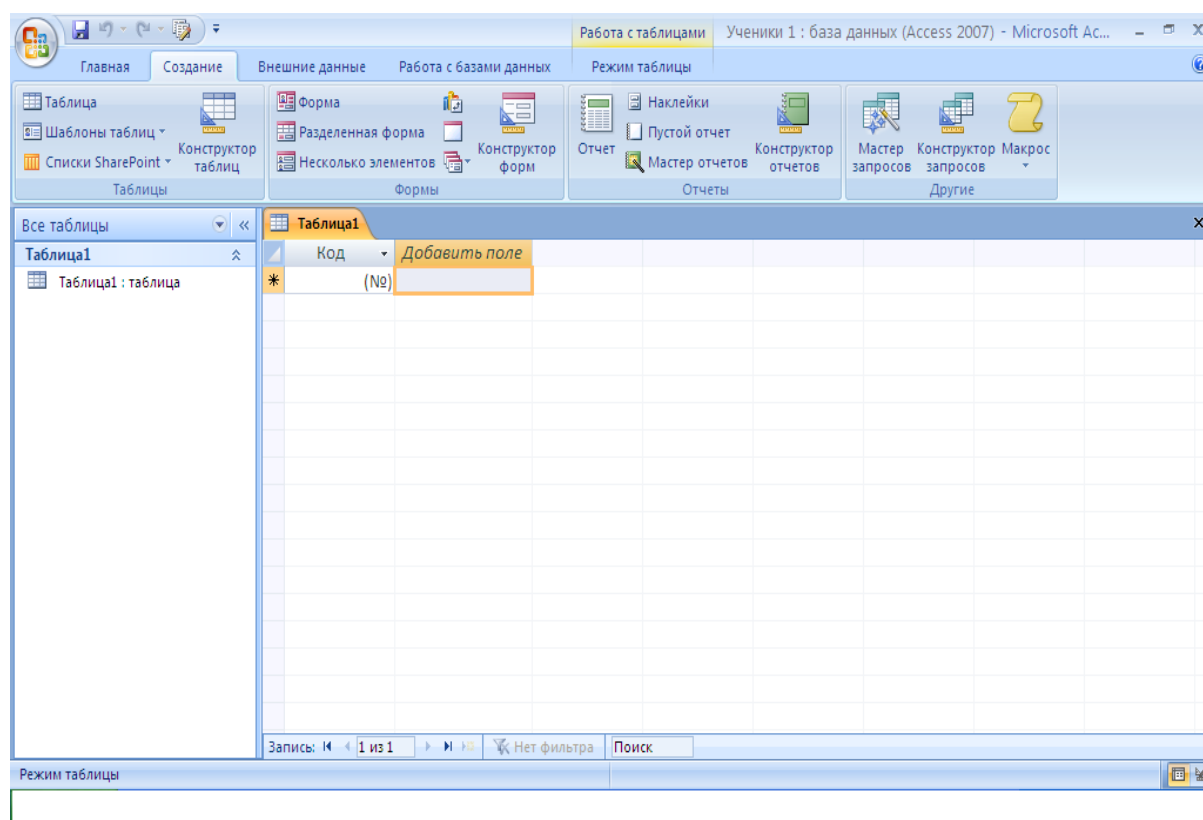


Рис. 2

4. На закладке «Создать» выбирается элемент БД, который необходимо создать. В меню находятся кнопки с пиктограммами режимов создания и редактирования («Конструктор») элементов БД. Те режимы, которые часто используются, отражены сверху рабочей поверхности окна.

1.4.1 Создание таблиц

1. Проектируем таблицы. Выбираем закладку «Таблицы» и выбираем режим «Создание таблицы в режиме конструктора», который позволяет описать все поля таблицы и задать их свойства.

2. Для каждого поля задаются имя, тип данных в соответствующих столбцах таблицы «Конструктора» (тип данных выбирается из предлагаемого списка). В нижней части окна располагаются «Свойства поля», где указывается его размер или стандартный формат и некоторые другие установки и свойства (Рис. 3).

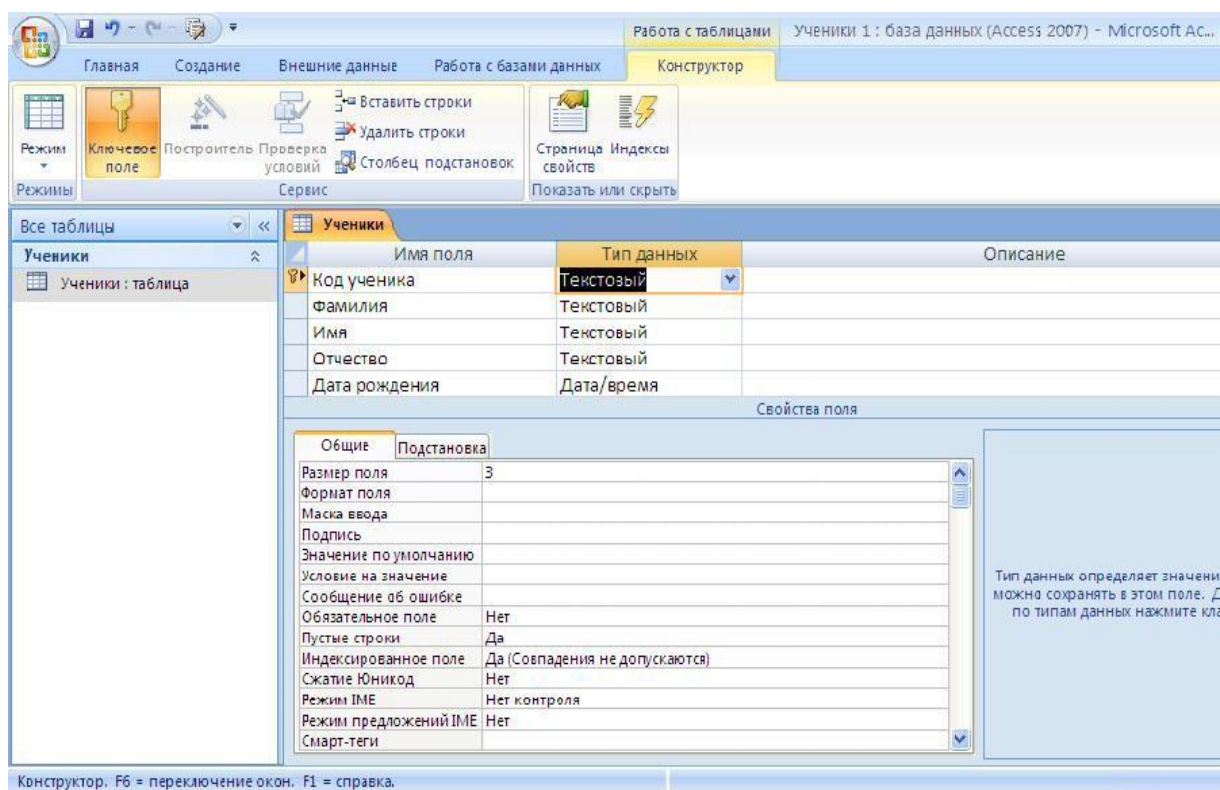


Рис. 3

3. После описания всех полей надо пометить ключевое поле, выделив поле «Код ученика» и нажав кнопку «Ключевое поле» .

Можно это сделать и посредством вызова контекстного меню через правую клавишу мыши.

4. Теперь таблицу нужно сохранить по команде меню «Файл» - «Сохранить» или кнопкой (Заккрыть) . Вместо предлагаемого имени «Таблица1» вводится имя таблицы (Рис. 4):

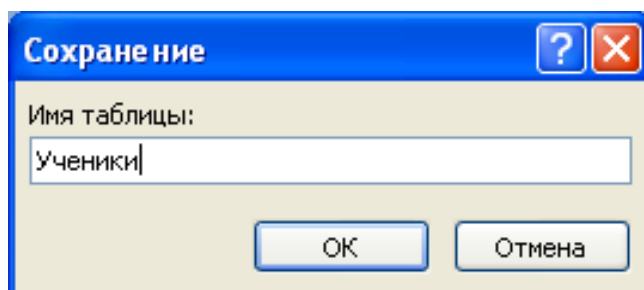


Рис. 4

5. Таким же образом создадим таблицу «КЛАССЫ» (поля таблицы описаны выше).

6. Вводить данные в таблицу можно уже в режиме таблицы. Для перехода в режим таблицы выбираем из списка «Режим таблицы» (Рис. 5) (или 2 раза щелкнуть по пиктограмме «Класс»):

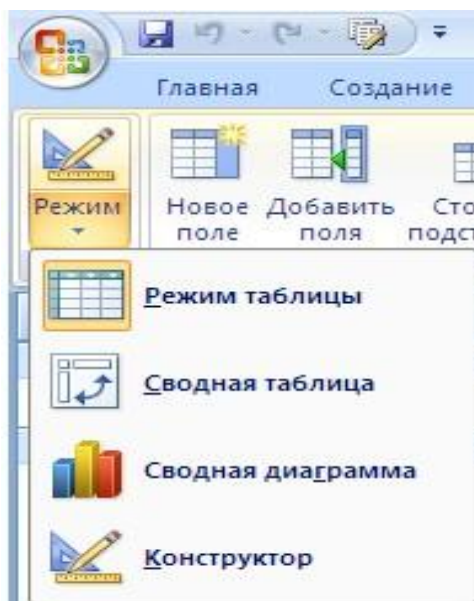


Рис. 5

7. Введем данные в таблицу «КЛАССЫ» (Рис. 6):

Класс	
Класс	Добавить поле
5а	
5б	
6а	
6б	
7а	
7б	
8а	
8б	
9а	
9б	
10а	
11а	

Рис. 6

8. В таблицу «**УЧЕНИКИ**» данные будем вводить через объект «**Форма**», т.к. это гораздо удобнее, тем более что у нас уже имеется список классов.

1.4.2 Проектирование форм

Форма – это наиболее удобное средство отображения данных. Формы удобна и наглядна, она использует графические возможности Windows (шрифты, графика и т.д.)

Для простоты создания формы воспользуемся возможностью автоматического создания автоформы в столбец.

1. Выберем «**Создание**» - «**Форма**», появится окно с формой «**Ученики**» (рис. 7, рис. 8):

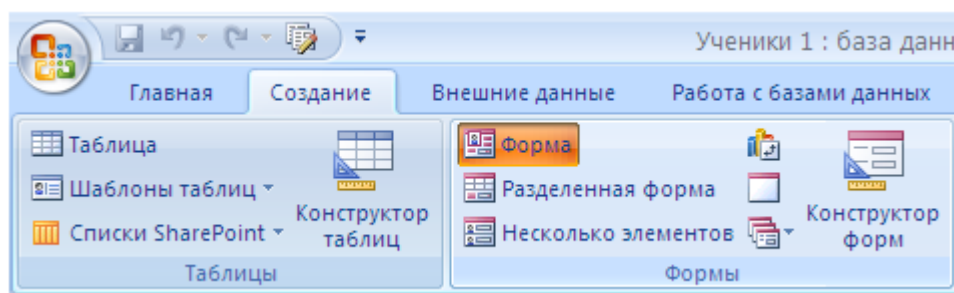


Рис. 7

Рис. 8

Либо создать форму через **«Конструктор форм»**. Выберем **«Создание»** – **«Другие формы»** – **«Мастер форм»** (Рис. 9).

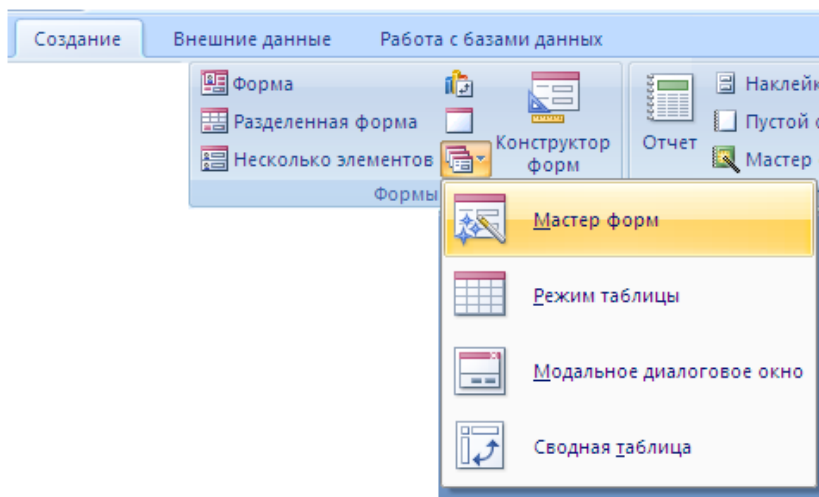


Рис. 9

Появится окно **«Создание форм»** - **«Выберете поле для формы»**. В поле **«Таблицы и запросы»** выбираем нужную нам таблицу (**«Ученики»**). Из окна **«Доступные поля»** переносим все

необходимые для создания формы поля таблицы в окне «**Выбранные поля**» с помощью кнопки  - **Далее** (Рис. 10).

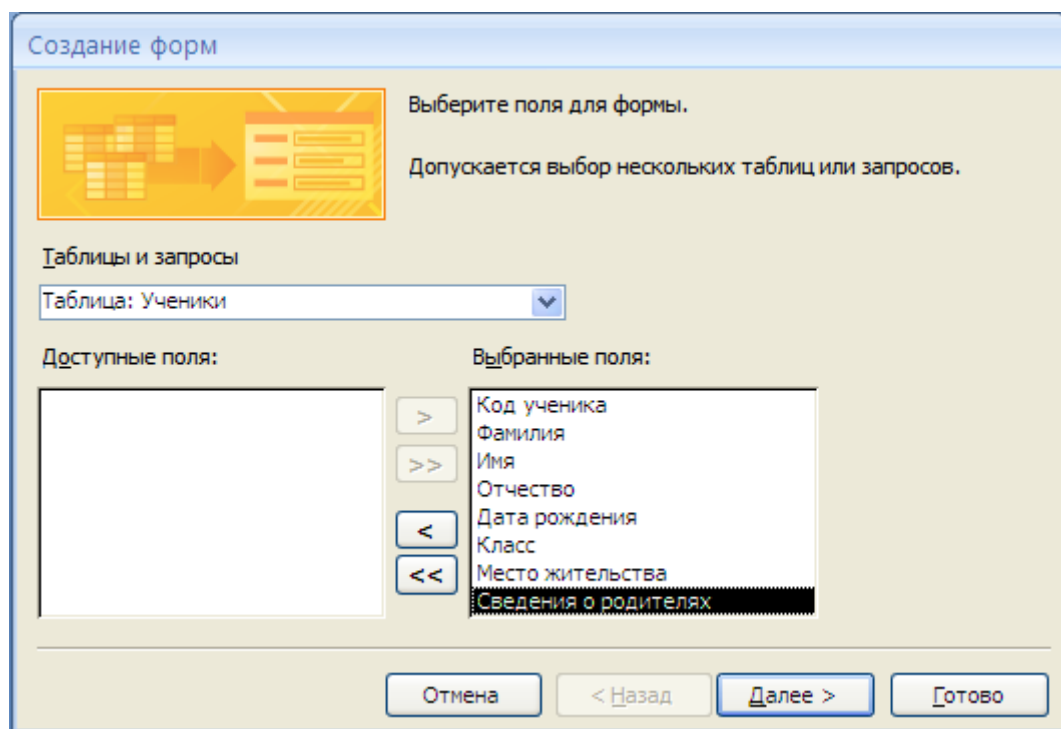


Рис. 10

Появится окно «**Создание форм**» - «**Внешний вид формы**» выбираем из списка вывод формы «**В один столбец**» (рис. 11). После появиться окно, в котором можно выбрать стиль формы.

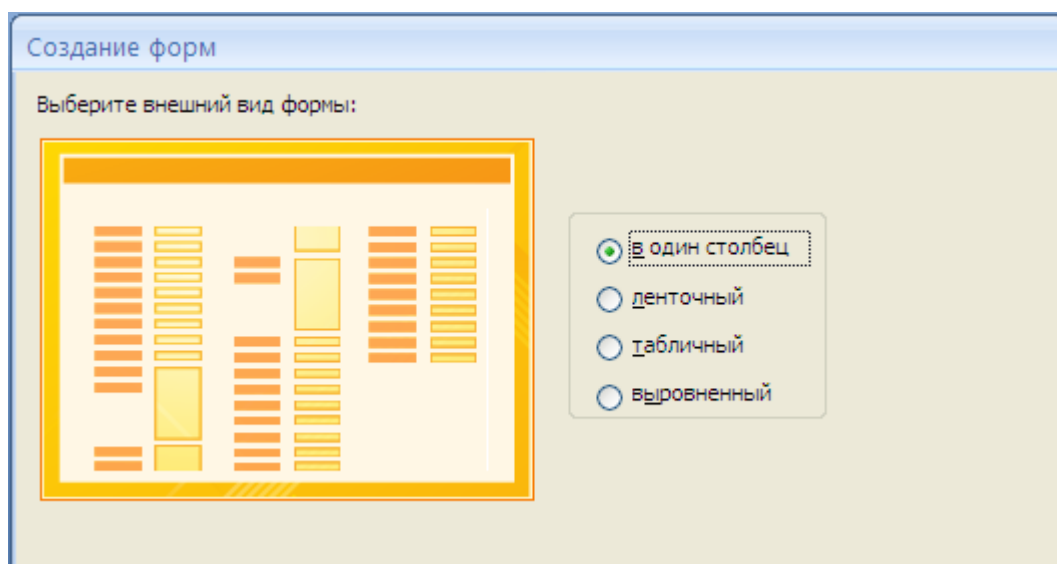


Рис. 11

Появится окно с предложением дальнейших действий. Выберем **«Открыть форму для просмотра и ввода данных»** - **«Готово»** (рис.12).

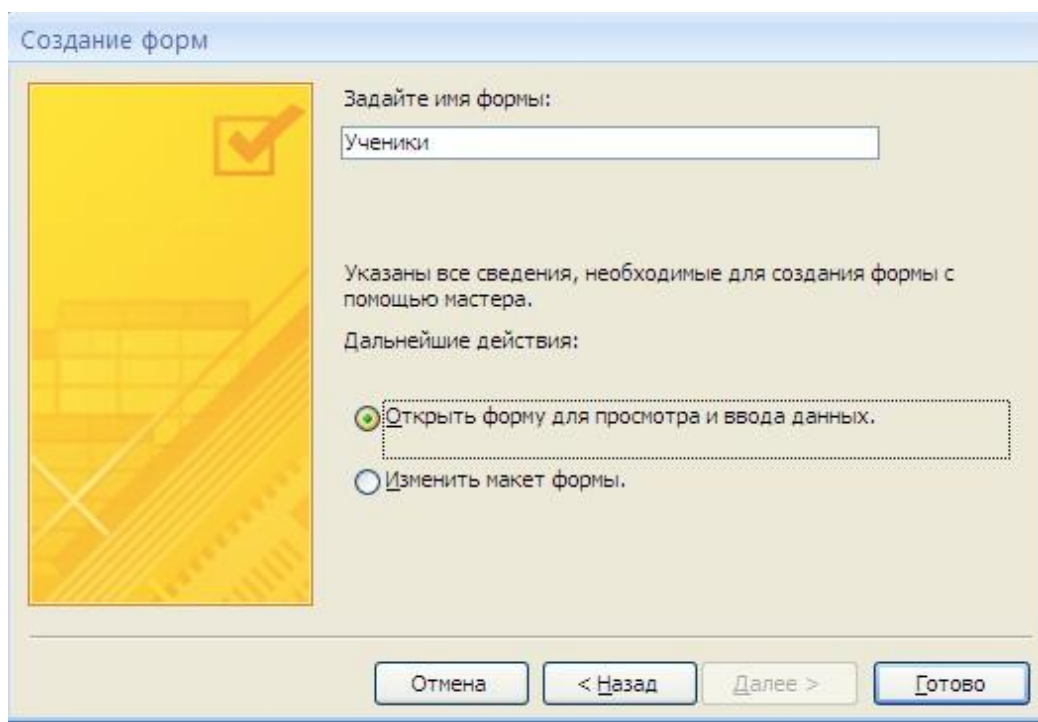


Рис. 12

После этого форма появится на экране (Рис.13).

Рис. 13

2. Доработаем ее в режиме **«Конструктор»** (подключаем пиктограмму ) (Рис. 14).



Рис. 14

3. Вызовем **«Панель элементов»** (подключить пиктограмму ) (Рис. 15).

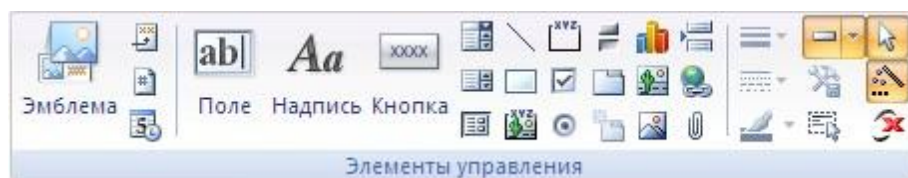


Рис. 15

5. Раздвинем **«Заголовок формы»** и **«Область данных»** и на возникшем месте с помощью пиктограммы  сделаем надпись **«Карточка ученика»**. Выделив эту область (дважды щелкнуть мышкой по рамке надписи), отформатируем надпись (установим размер шрифта) (Рис. 16).

6. Поле **«Код ученика»** перетянем в ту же область.

Рис. 16

7. Для удобства ввода информации создадим поле со списком. Список может содержать несколько значений или записи из связанной таблицы. Исходные данные для формирования списка могут задаваться различными способами: из таблиц, запросов или вводиться через точку с запятой. В нашем случае исходные данные задаются из таблицы **«КЛАССЫ»**:

- выделите поле **«Класс»**;
- удалите полностью это поле;
- выберите пиктограмму  - **«Поле со списком»**;
- вычертите поле в любой свободной части формы.
- появится окно **«Создание поля со списком»**. Выберем строку **«Объект «поле со списком» будет использовать значения из таблицы или запроса»** - Далее (Рис. 18).

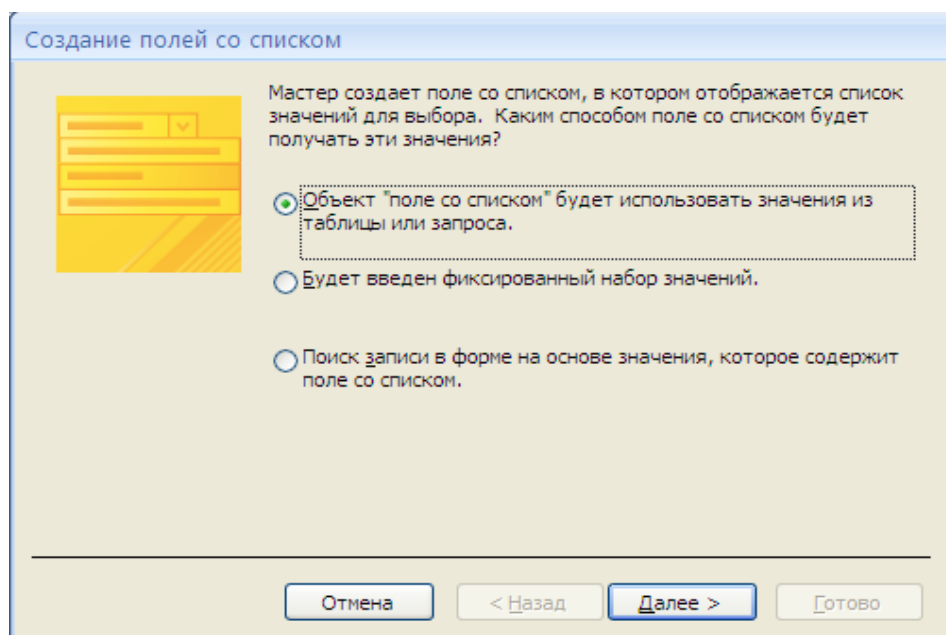


Рис. 18

➤ появится окно «**Создание поля со списком**», где нужно выбрать таблицу, которая будет содержать поле со списком. Выбираем «**Таблица: Класс**» - Далее (Рис. 19).

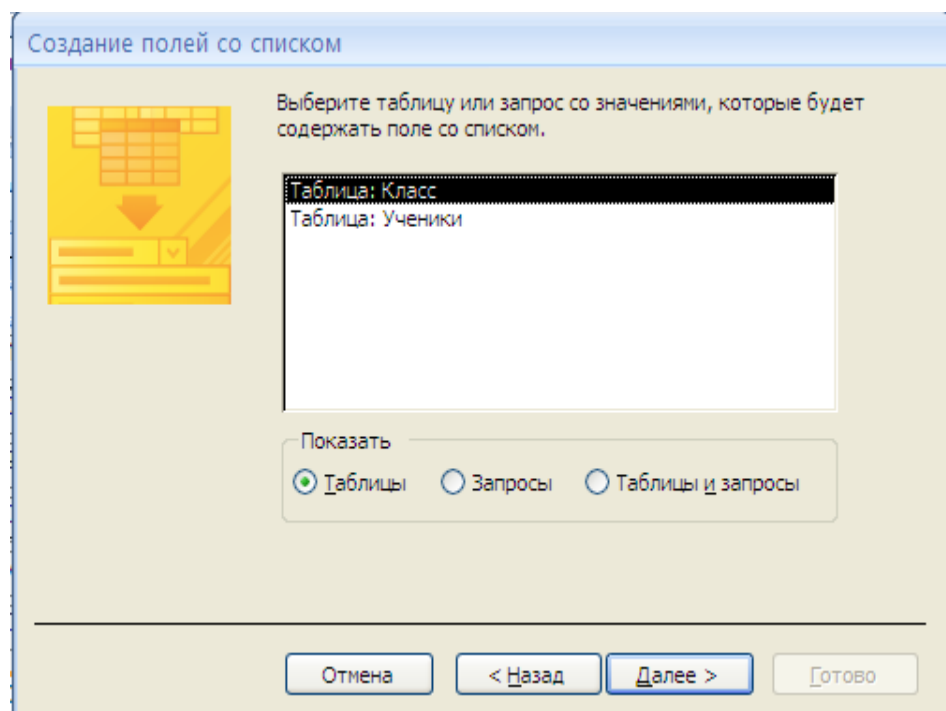


Рис. 19

➤ в следующем появившемся окне «**Создание поля со списком**» необходимо выбрать из доступных полей те поля со

значениями, которые следует включить в поле со списком. Выберем поле «Класс» - Далее (Рис. 20).

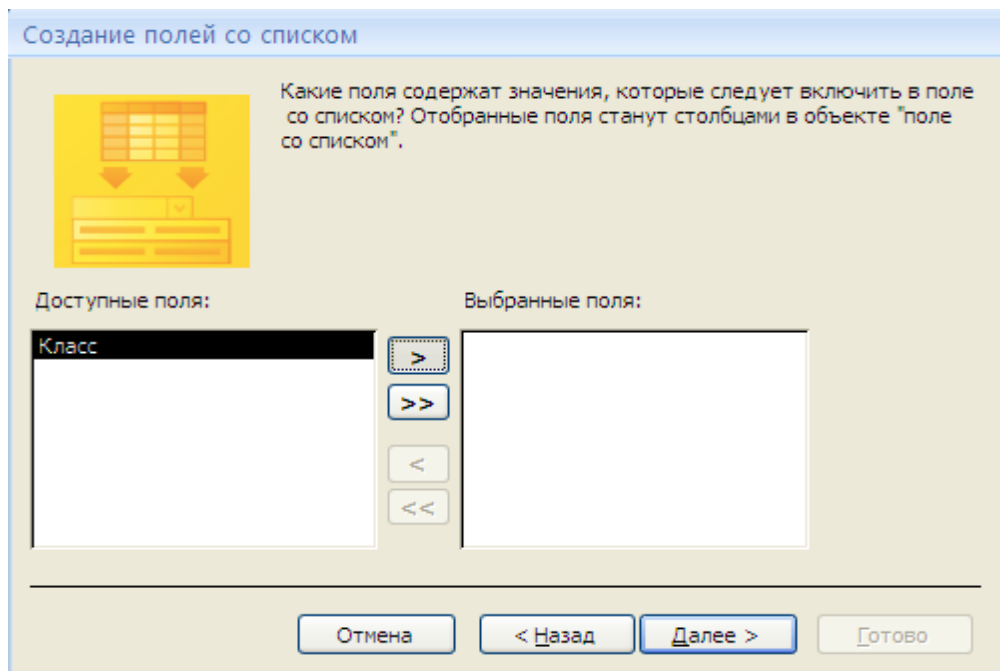


Рис. 20

➤ в появившемся окне можно произвести сортировку записей. Далее.

➤ выбранное значение из объекта поле со списком, необходимо сохранить в поле. Выбираем «Сохранить в поле»- «Класс» - «Далее» (Рис.21).

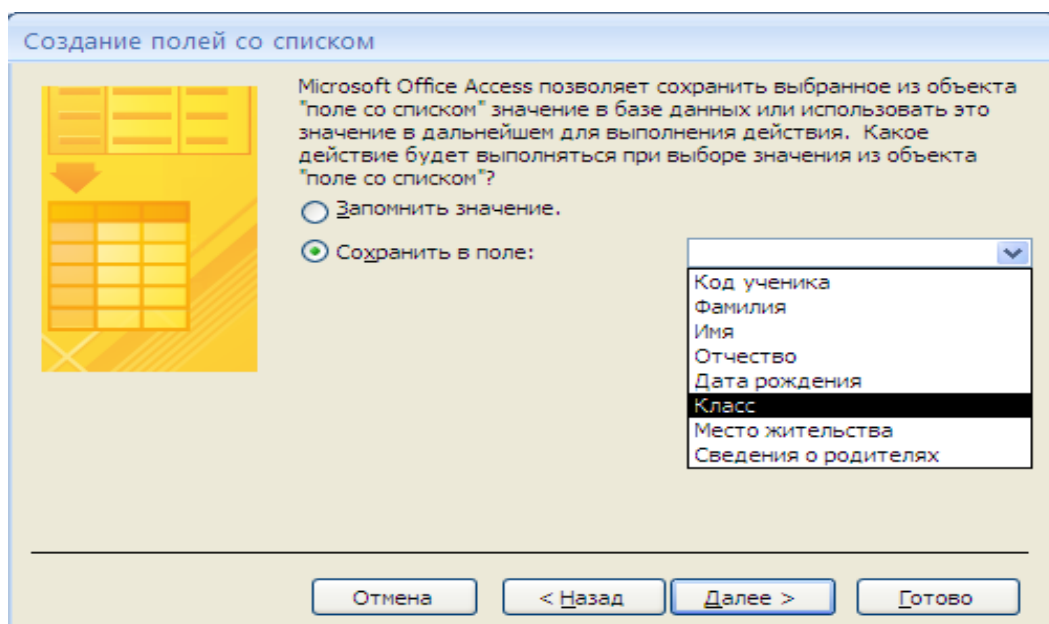


Рис. 21

➤ Полю со списком нужно дать имя. Назовем его «Класс» - **Готово.**

➤ Закрываем окно и даем название форме «**Карточка ученика**» (Рис. 22);

Рис. 22

➤ просматриваем нашу форму (2 раза щелкаем по пиктограмме «**Карточка ученика**») и проверяем работу поля со списком (Рис. 23).

Добавим в форму кнопку «**Закрывает форму**», для этого:

- входим в режим конструктора ;
- на «**Панели элементов**» выбираем пиктограмму кнопка ;
- вычерчиваем кнопку на поле нашей формы.

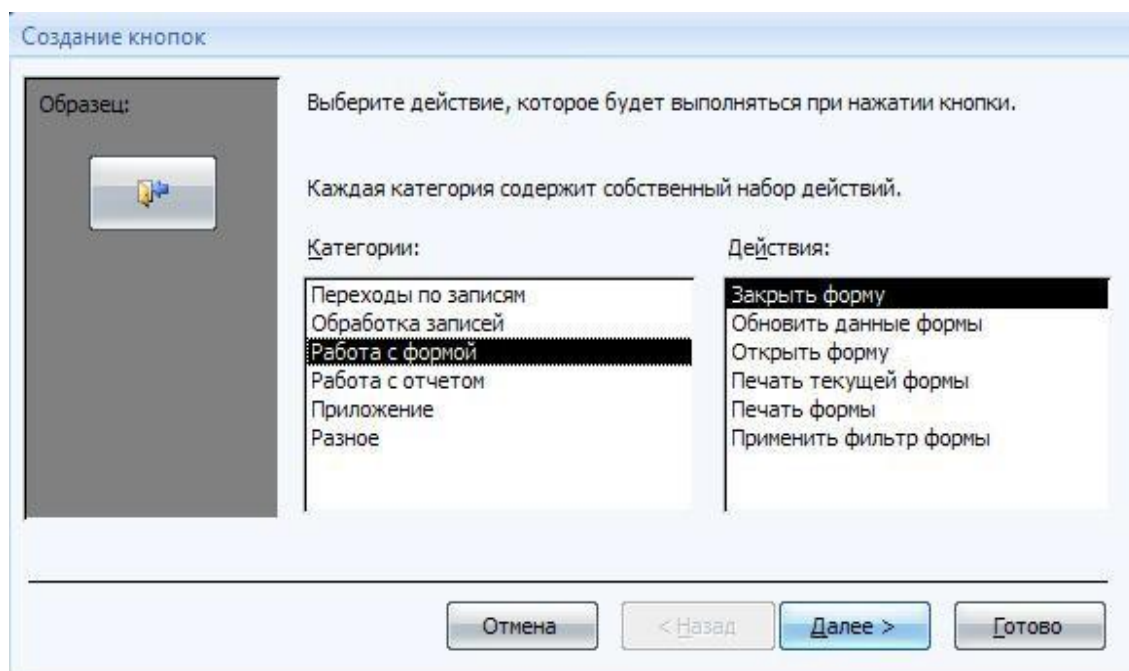


Рис. 23

После рисования кнопки появилось окно «Создание кнопок» (Рис. 24).

➤ Выбираем «Категорию» - «Работа с формой» и «Действия» - «Заккрыть форму», «Далее»;

➤ В окне «Создание кнопок» выбираем «Текст» или «Рисунок», «Далее» (Рис. 24):

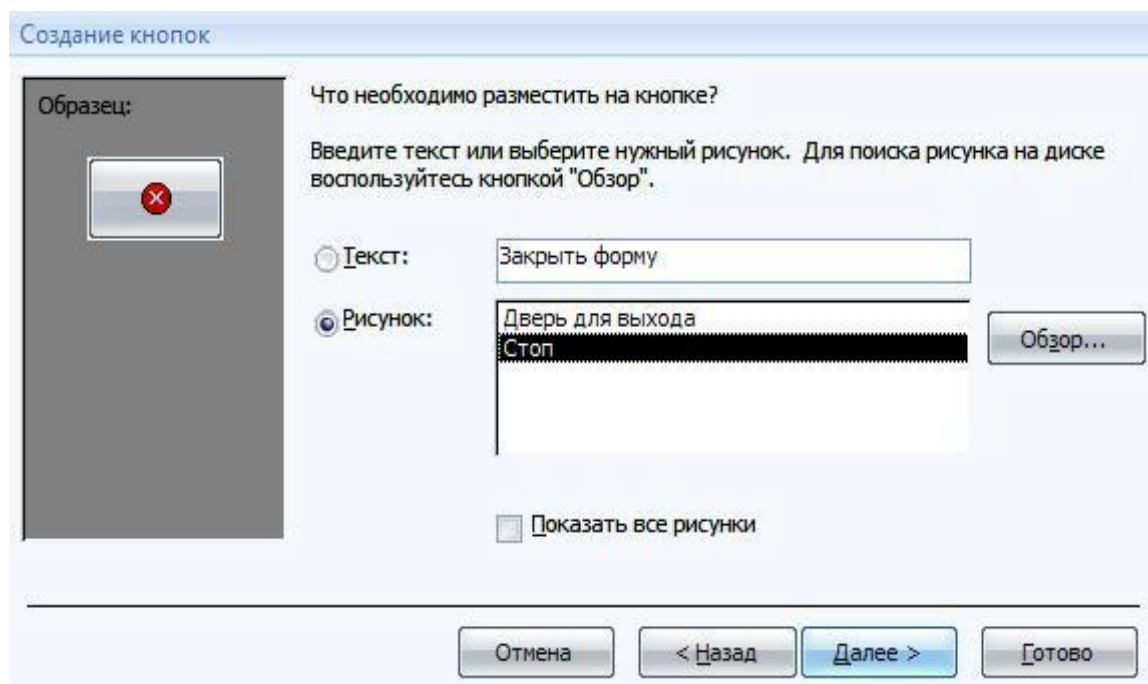


Рис. 24

- Имя кнопке можно не задавать - «Готово».
- Даем имя форме - «**Карточка ученика**».
- Получившаяся форма готова к заполнению (Рис. 25):

Рис. 25

1.4.3 Ввод данных

Мы уже заполнили таблицу «**КЛАССЫ**», и теперь можно приступить к заполнению таблицы «**УЧЕНИКИ**» через форму «**Карточка ученика**».

При заполнении формы после ввода информации для перехода к следующему полю, нужно нажать клавишу «Enter». После заполнения последнего поля формы MS Access автоматически добавит новую запись в таблицу. Переход для просмотра записей осуществляется с помощью кнопок управления, расположенных внизу формы (Рис. 26) **Запись: [Назад] [Вперед] [1 из 1] [Назад] [Вперед] [Новая запись]**:

- переход к первой записи;
- переход к предыдущей записи;
- переход к последующей записи;
- переход к последней записи;
- переход к новой записи.

Карточка ученика	
Код ученика:	1
Фамилия:	Иванов
Имя:	Иван
день рождения:	25.01.1991
класс:	11а
Место жительства:	г. Красноярск
Сведения о родителях:	служащие

Рис. 26

1.4.4 Запросы

С помощью запроса можно быстро и эффективно получить доступ к данным, хранящимся в таблице. Запросы представляют собой важное дополнение к таблицам.

При выполнении запросов считываются данные из таблиц и отображается результат в режиме таблицы. Данные всегда хранятся в таблице, результат выполнения запроса не сохраняется. В запросе хранится только инструкция по организации данных в запросе.

Задание:

- Создать запрос на выборку всех учеников.
- Создать запрос на выборку из списка учеников по признакам.

Для создания запроса выберем режим «Конструктора» (можно использовать «Мастер запросов»).

1. Выбираем «Объекты» - «Запросы», «Создание запроса в режиме конструктора».

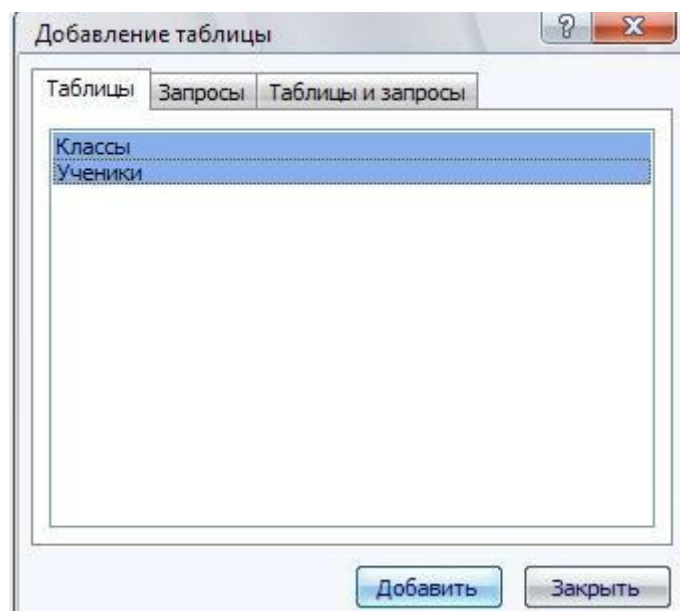


Рис. 26

2. В окне «Добавление таблицы» добавляем таблицы «УЧЕНИКИ» и «КЛАССЫ».

Закрываем окно «Добавление таблицы».

3. В окне конструктора запросов «Запрос1: Запрос на выборку» в бланк запроса переносим имена полей, необходимые в запросе. В нашем случае, это все поля таблицы (Рис. 27):

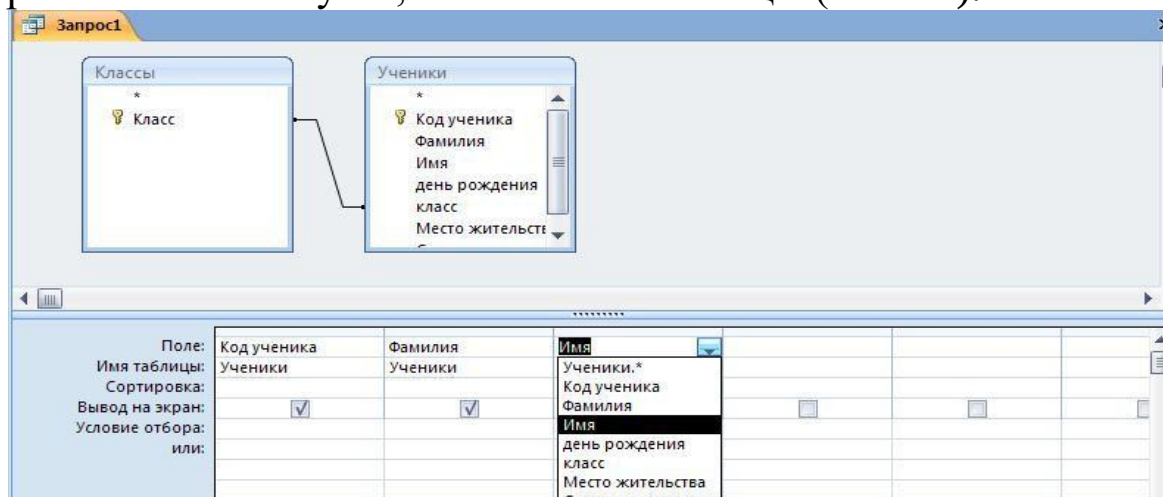


Рис. 27

Закрываем окно конструктора и сохраняем запрос под именем «Запрос Все ученики» (Рис. 28):



Рис. 28

4. Запускаем запрос на выполнение (Рис. 29):

Запрос все ученики						
Код ученик	Фамилия	Имя	день рожд	Место жите	Сведения о	Класс
1	Иванов	Иван	25.01.1991	г. Красноярск	служащие	11а
2	Петрова	Марина	06.01.1997	г.Красноярск	рабочие	7а
3	Залюбовская	Майя	18.01.1995	г.Красноярск	служащие	9а
*						

Рис. 29

5. Создаем второй запрос, в котором нужно найти учеников по признакам, например, по классам. Тогда в строке «Условия отбора» указываем конкретный признак. Это может быть определенный класс или различные условия:

Условие	Поле	Условие отбора
Все ученики с фамилией на букву З	Фамилия	З*
Все ученики 10а класса	Класс	10а
Ученики старше 1992 г.р.	Дата рождения	<01.01.1992
Ученики, имена которых в написании имеют букву я	Имя	*я*
Ученики, у которых третья буква фамилии буква т	Фамилия	??т*

Если нужно провести выборку по нескольким условиям, то условия указываются в разных столбцах одной строки.

Если необходима выборка по нескольким условиям, то в том же столбце на последующих строках необходимо указать остальные условия отбора.

Например, нужно найти всех учеников (фамилия и имя) 10а классов с фамилией начинающейся на букву П (в запросе указываем только те поля которые требуются). Тогда конструктор запросов будет такой (Рис. 30):

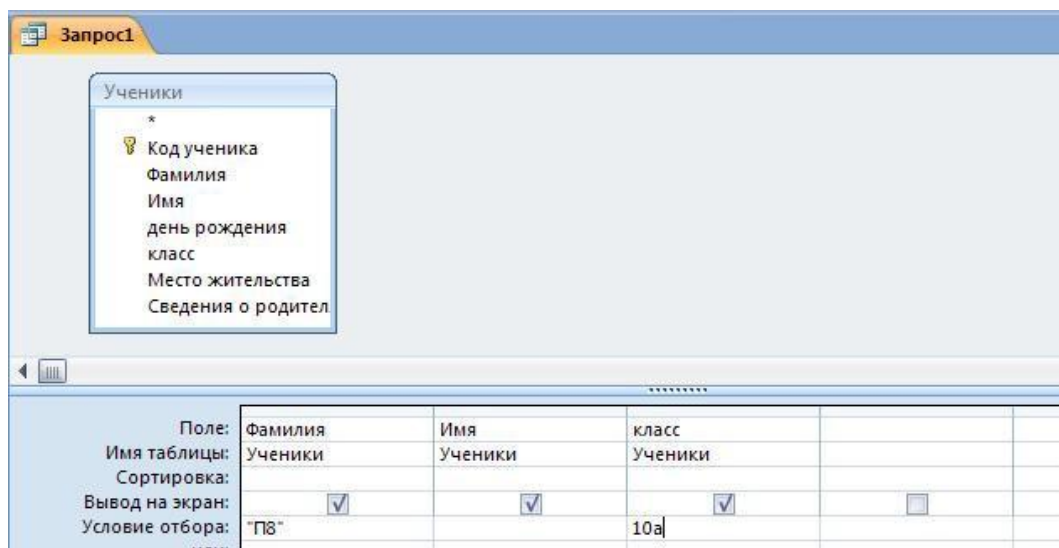


Рис. 30

6. В условии отбора можно ввести запуск диалогового окна. Вопрос должен быть заключен в квадратные скобки. Например, в поле «Класс» в Условии отбора вводим **[Введите класс]**.

Назовем этот запрос «Запрос по классам» (Рис. 31).

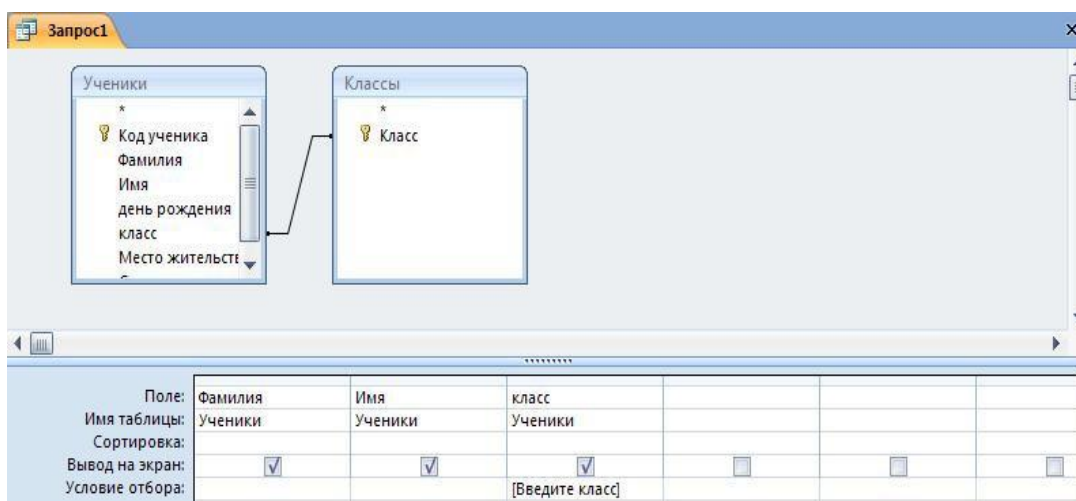


Рис. 31

7. В результате формирования этого запроса мы получим вопрос в диалоговом окне, и по ответу на этот вопрос будет формироваться запрос (Рис. 32 и Рис. 33):

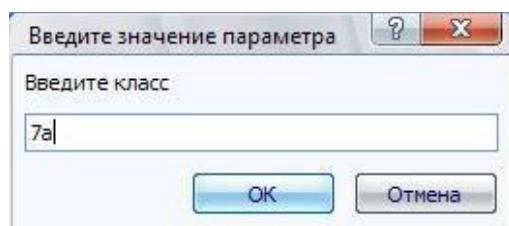
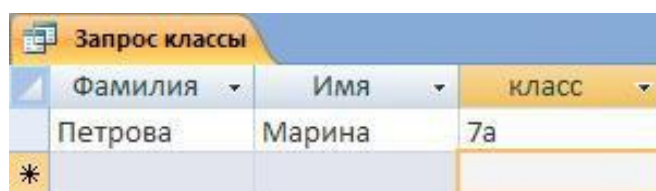


Рис. 32



Фамилия	Имя	класс
Петрова	Марина	7a
*		

Рис. 33

1.4.5 Отчеты

Перед началом конструирования отчета необходимо подготовить его макет. Наиболее простой и рекомендуемый способ создания отчета – с помощью мастера.

1. Выбрать «Создать» - «Отчеты» - режим «Мастер отчетов», далее в окне чуть ниже указать имя таблицы или запроса как источника данных, на основании которых будут создан отчет, Выбрать таблицу «УЧЕНИКИ» (Рис. 34).

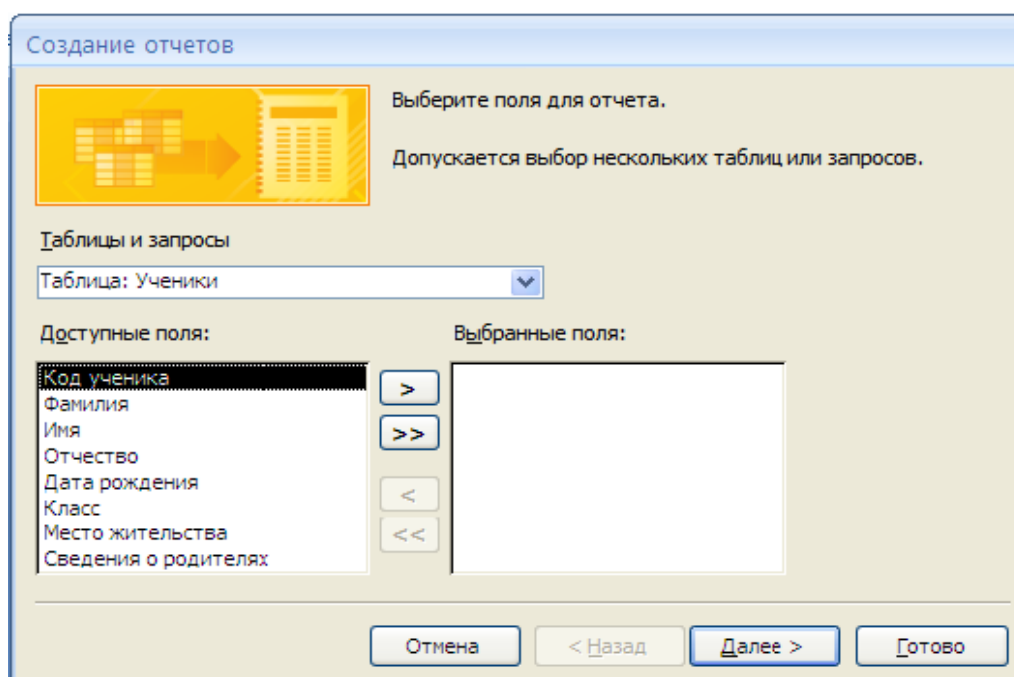


Рис. 34

2. После нажатия кнопки «ОК» на экране разворачивается диалоговое окно для указания имен полей таблицы, которые должны присутствовать в отчете.

3. Выбрать нужные поля: для этого: выделить нужное поле в окне «Доступные поля:» и с помощью кнопки **>** перенести его в окно «**Выбранные поля:**». Нажать кнопку «Далее».

4. Указать уровни сортировки (группировки данных в отчете относительно какого-либо поля таблицы). Имена полей относительно которых будет происходить сортировка выделяются в левой части окна, и нажимается кнопка **>**. В случае использования многоступенчатой сортировки необходимые поля переносятся последовательно (Рис. 35).

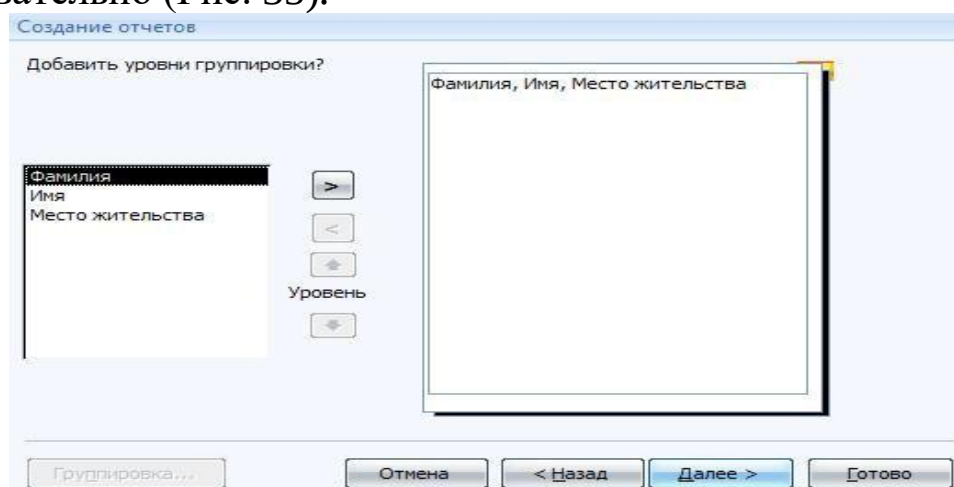


Рис.35

В приведенном примере выбрана сортировка относительно поля «Фамилия» (Рис. 36).

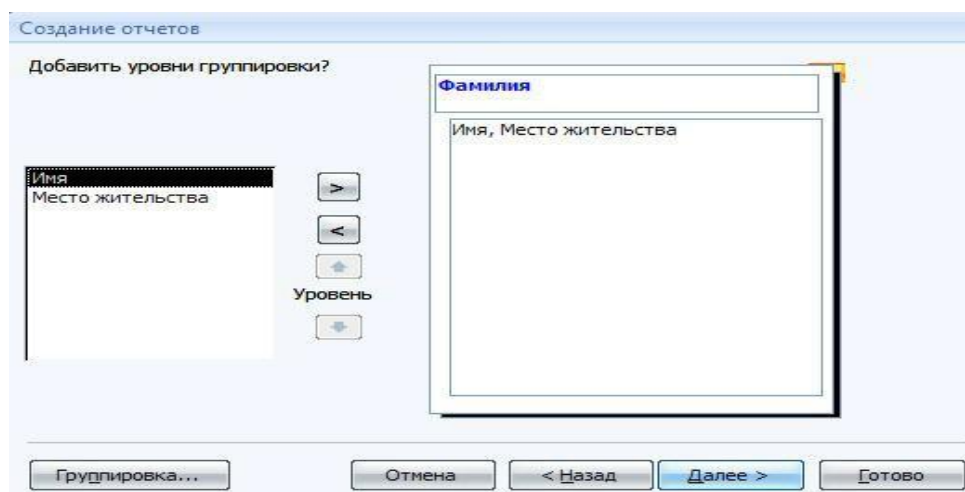


Рис. 36

5. Указав уровни сортировки, нажимаем на кнопку «Далее». На следующем шаге (изображение диалогового окна не приводится) указывается в каком порядке производить сортировку: по возрастанию, то есть в алфавитном порядке или по убыванию. Режим сортировки по умолчанию установлен – по возрастанию.

6. Выбрать макет отчета из предлагаемых стандартных образцов, указать ориентацию бумаги, учитывая примерную ширину записи в таблице. Отступы от края листа устанавливаются автоматически в зависимости от характеристик и типа печатающего устройства (Рис. 37).

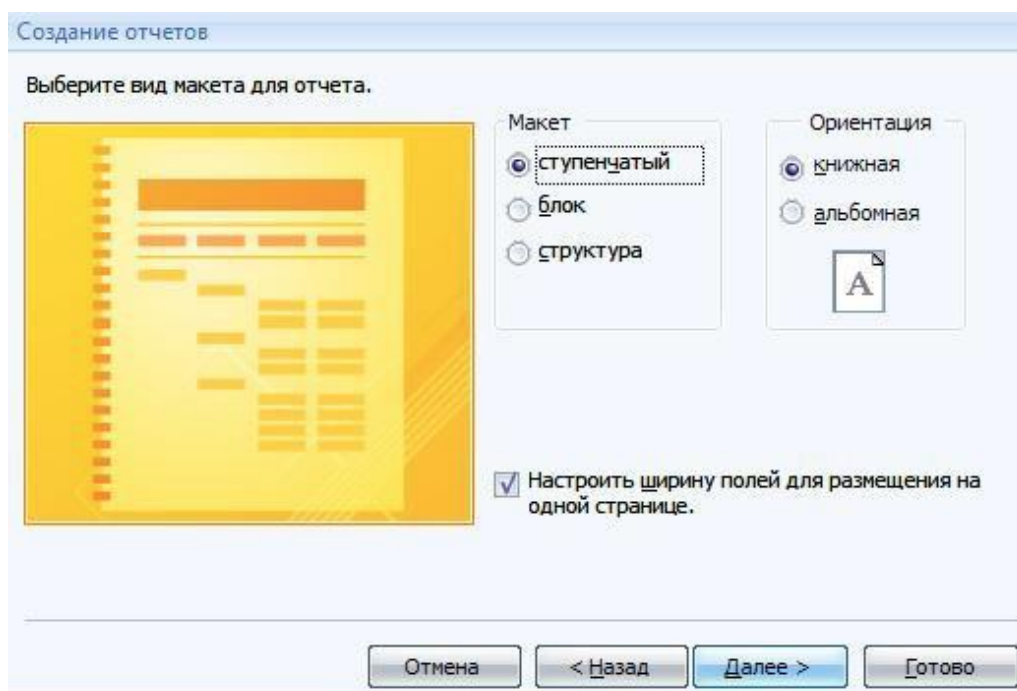


Рис.37

Выбрав макет нажимаем кнопку «Далее».

7. Выбираем стиль заголовка отчетов из стандартного набора (Рис. 38).

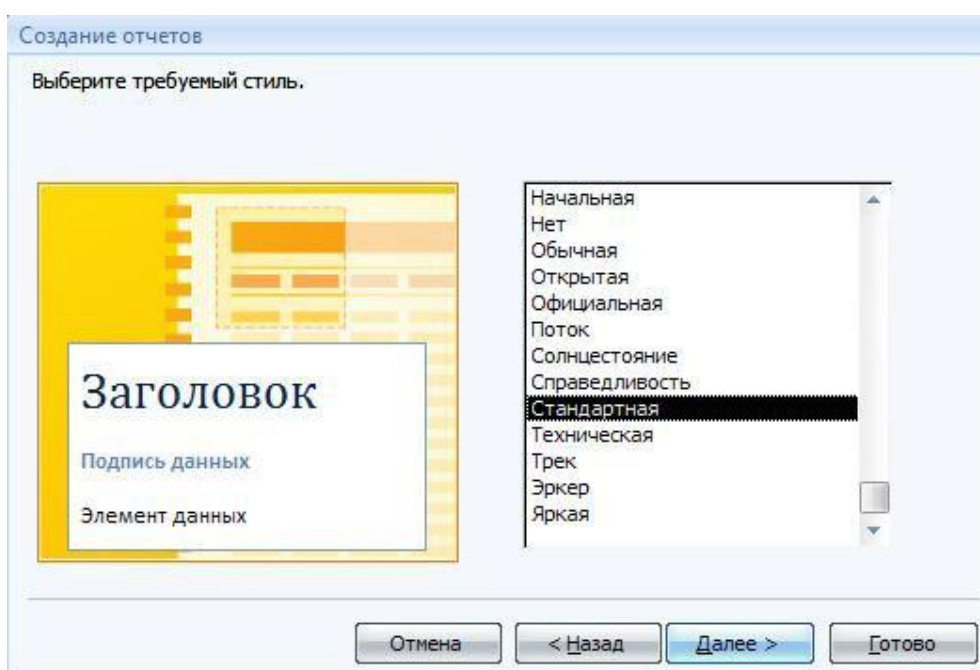


Рис. 38

8. Можно изменить имя созданного отчета (по умолчанию имя отчету присваивается по имени объекта источника данных). В случае, если создается несколько отчетов на один объект, то в имена отчетов автоматически вводятся цифры (Рис. 39).

Создание отчетов

Задайте имя отчета:

Ученики

Указаны все сведения, необходимые для создания отчета с помощью мастера.

Дальнейшие действия:

☒ Просмотреть отчет.

☐ Изменить макет отчета.

Отмена < Назад Далее > Готово

Рис. 39

Нажимаем кнопку «Готово» (Рис. 40).

Фамилия	Имя	Место жительства
Залюбовс	Майя	г.Красноярск
Иванов	Иван	г. Красноярск
Петрова	Марина	г.Красноярск

Рис. 40

9. В готовом отчете можно форматировать поля, для этого необходимо нажать пиктограмму **«Конструктор»**  .

10. Аналогичным образом создаются отчеты на базе запросов.

1.4.6 Кнопочные формы

Заключительным этапом разработки БД является создание **«Кнопочной формы»**. Она служит для объединения всех разработанных объектов (таблиц, форм, запросов, отчетов) в едином диалоговом приложении, что позволяет запускать на выполнение различные режимы работы с базой данных из одной формы. **«Кнопочная форма»** – это панель управления приложением.

Создадим главную кнопку форму, которая будет использоваться как панель управления приложением **«УЧЕНИКИ»**.

1. Выполним команды **«Работа с базами данных»** - **«Диспетчер кнопочных форм»** или на панели инструментов выберем пиктограмму .

В окне **«Страница»** кнопочной формы подтвердим создание кнопочной формы.

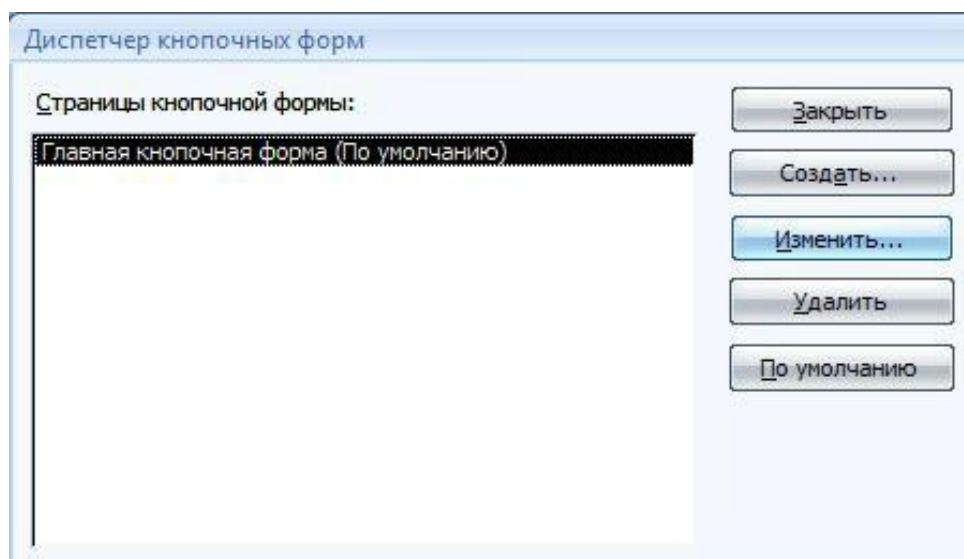


Рис. 41

2. Появится окно со строкой **«Главная кнопочная форма»** (Рис. 41)

В окне **«Диспетчер кнопочных форм»** формируется список кнопочных форм разных уровней. Для формирования элементов главной кнопочной формы нажмем кнопку **«Изменить»**.

3. В окне диалога **«Изменение страницы кнопочной формы»** переименуем строку **«Главная кнопочная форма»** в приложение **«Ученики»**, введя это имя в поле **«Название кнопочной формы»** (Рис. 42).

Рис. 42

4. Для формирования кнопок, обеспечивающих выполнение пользователем своих задач, нажмем кнопку **«Создать»**.

5. На экране появится окно **«Изменение элемента кнопочной формы»**. В строке **«Текст»** наберем подпись для первой кнопки, например, **«Ввод нового ученика»**. В строке **«Команда»** выберем команду **«Открытие формы в режиме добавления»**. В строке **«Форма»** выберем форму **«Карточка ученика»**.

5. Аналогично можно создать другие элементы кнопочной формы, задав соответствующие подписи, команды и названия форм, запросов и отчетов. После завершения работы с диспетчером кнопочных форм в списке форм базы данных появляется кнопочная форма. Главная кнопочная форма будет запускаться при открытии базы данных, так как ссылка на эту форму создается диспетчером

кнопочных форм в параметрах запуска. Для оформления в кнопочную форму можно вставить рисунок (Рис. 43).

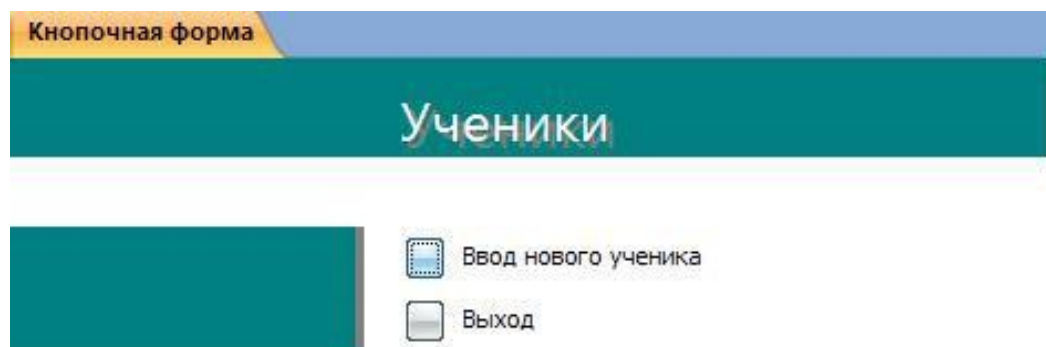


Рис. 43

В приведенном примере кнопочной формы содержатся некоторые объекты созданной БД. Последний элемент кнопочной формы «**ВЫХОД**» формируется по стандартной команде «**Выход из приложения**».

2. ПРОЕКТИРОВАНИЕ БАЗЫ ДАННЫХ «ПРОКАТ ФИЛЬМОВ»

Задание: разработать базу данных «Прокат фильмов», состоящую из трех таблиц:

Таблица 1. ФИЛЬМ

Таблица 2. КЛИЕНТ

Таблица 3. ВЫДАЧА ФИЛЬМОВ (КВИТАНЦИЯ)

2.1 Проектирование логической структуры БД

Определим, из каких полей будет состоять каждая таблица, их типы размер. Описание полей таблицы «КЛИЕНТ»:

Ключ	Имя поля	Тип данных	Размер
⇒	Код клиента	Текстовый	3
	Фамилия	Текстовый	15
	Имя	Текстовый	15
	Отчество	Текстовый	15
	Адрес	Текстовый	40
	Телефон	Текстовый	15

Описание полей таблицы «ФИЛЬМ»:

Ключ	Имя поля	Тип данных	Размер
⇒	Код фильма	Текстовый	3
	Название фильма	Текстовый	40
	Страна	Текстовый	15
	Режиссер	Текстовый	15
	Время показа	Числовой	Длинное
	Жанр	Текстовый	50

Описание полей таблицы «ВЫДАЧА ФИЛЬМОВ»:

Ключ	Имя поля	Тип данных	Размер
⇒.	№	Текстовый	3
	Код фильма	Текстовый	3
	Дата выдачи	Дата\время	15
	Залог	Денежная	Денежная
	Код клиента	Текстовый	3
	Количество дней	Числовой	Длинное
	Цена проката в день	Денежная	Денежная
	Просрочено(дней)	Числовой	Длинное
	Итого (руб.):	Денежная	Денежная

2.2 Создание БД

Запускаем Access и выбираем «Создание базы данных» - «Новая база данных». Даем имя нашей базе «Прокат фильмов».

2.3 Создание таблиц

Создадим таблицы нашей базы в режиме конструктора: «Объекты» - «Таблицы» - «Создание таблицы в режиме конструктора» (Рис.44). Создадим таблицы КЛИЕНТ (Рис.45), ФИЛЬМ (Рис.46), ВЫДАЧА ФИЛЬМОВ (рис.47).

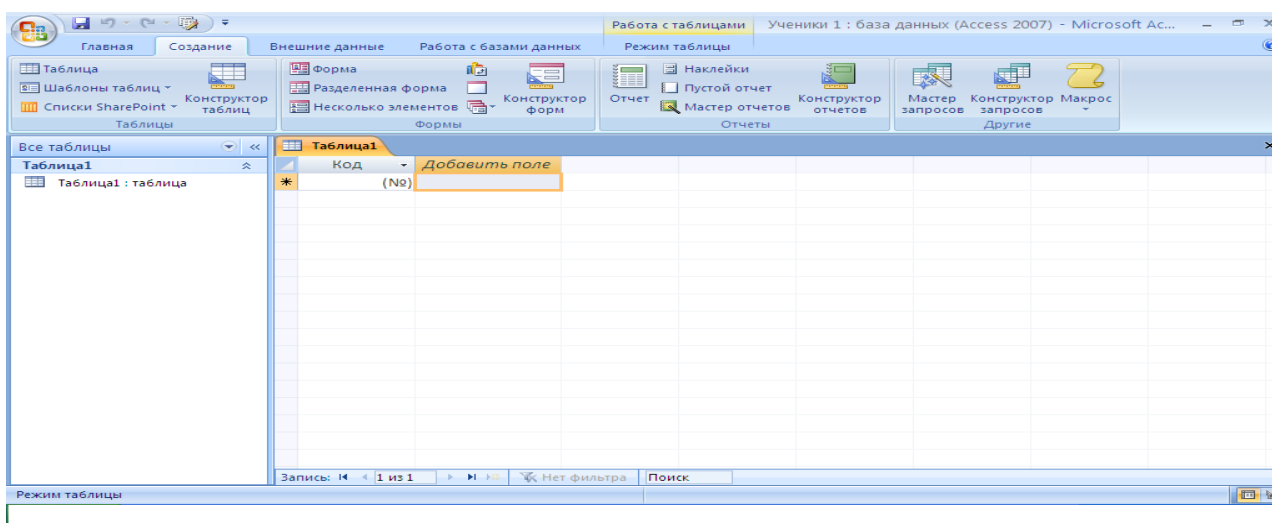


Рис. 44

Таблица КЛИЕНТ

Клиент	
Имя поля	Тип данных
Код клиента	Текстовый
Фамилия	Текстовый
Имя	Текстовый
Отчество	Текстовый
Адрес	Текстовый
Общие Подстановка	
Размер поля	3
Формат поля	
Маска ввода	
Подпись	
Значение по умолчанию	
Условие на значение	
Сообщение об ошибке	
Обязательное поле	Нет
Пустые строки	Да
Индексированное поле	Да (Совпадения не допускаются)
Сжатие Юникод	Нет

Рис. 45

Для удобства ввода информации в таблицу ФИЛЬМ создадим для поля «Страна производитель» поле со списком. Исходные данные для формирования списка могут задаваться различными способами: из таблиц, запросов или вводиться через точку с запятой. В нашем случае исходные данные зададим во вкладке «Подстановка» через точку с запятой (Рис. 46):

Таблица ФИЛЬМ

Фильм	
Имя поля	Тип данных
Код фильма	Текстовый
Название фильма	Текстовый
Страна	Текстовый
Режиссер	Текстовый
Время показа	Числовой
Общие Подстановка	
Размер поля	3
Формат поля	
Маска ввода	
Подпись	
Значение по умолчанию	
Условие на значение	
Сообщение об ошибке	
Обязательное поле	Нет
Пустые строки	Да
Индексированное поле	Да (Совпадения не допускаются)
Сжатие Юникод	Нет
Режим IME	Нет контроля
Режим предложений IME	Нет
Смарт-теги	

Рис. 46

Таблица ВЫДАЧА ФИЛЬМОВ

Выдача фильмов	
Имя поля	Тип данных
№	Текстовый
Код фильма	Текстовый
Дата выдачи	Дата/время
Залог	Денежный
Код клиента	Текстовый

Общие Подстановка	
Размер поля	3
Формат поля	
Маска ввода	
Подпись	
Значение по умолчанию	
Условие на значение	
Сообщение об ошибке	
Обязательное поле	Нет
Пустые строки	Да
Индексированное поле	Да (Совпадения не допускаются)
Сжатие Юникод	Нет
Режим IME	Нет контроля
Режим предложений IME	Нет
Смарт-теги	

Рис. 47

2.4 Установка связей между таблицами

Выбираем в меню команду «Работа с базами данных» - «Схема данных» .

Из окна «Добавление таблицы» добавляем три таблицы в окно «Схема данных» (Рис. 48).

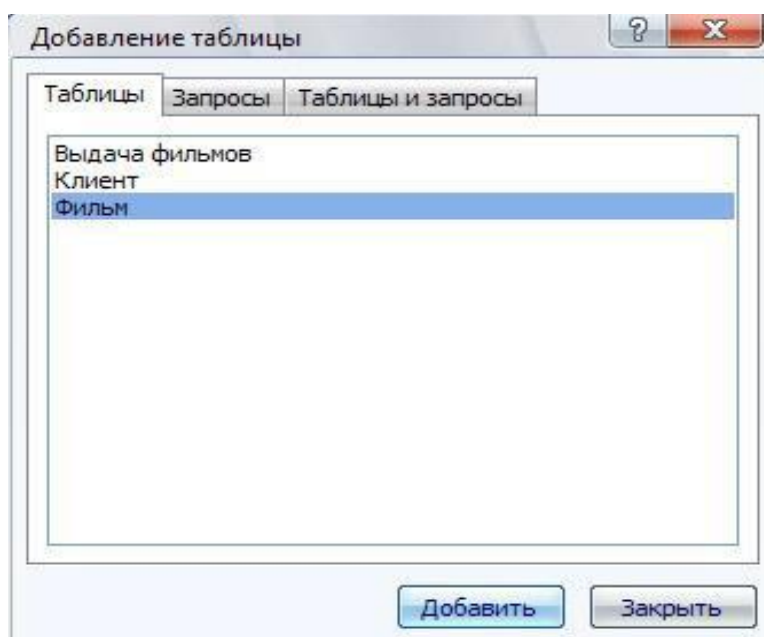


Рис. 48

Ставим указатель на поле «Код клиента» в таблице «КЛИЕНТ» и не отжимая кнопку мыши переносим указатель на то же поле «Код клиента» таблицы «ВЫДАЧА ФИЛЬМОВ».

В появившемся окне отмечаем флажками:

- ✓ Обеспечение целостности данных;
- ✓ Каскадное обновление связанных полей;
- ✓ Каскадное удаление связанных полей;
- ✓ Тип отношения «один-ко-многим»;
- ✓ «Создать» (Рис. 49).

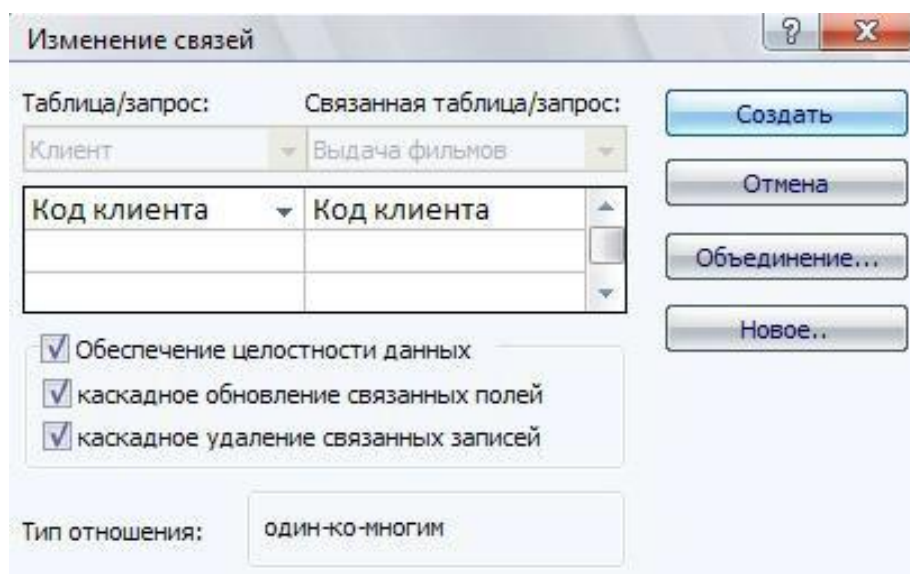


Рис. 49

Такую же операцию производим с полем «Код фильма», связывая таблицы «ВЫДАЧА ФИЛЬМОВ» и «ФИЛЬМ» (Рис. 50).

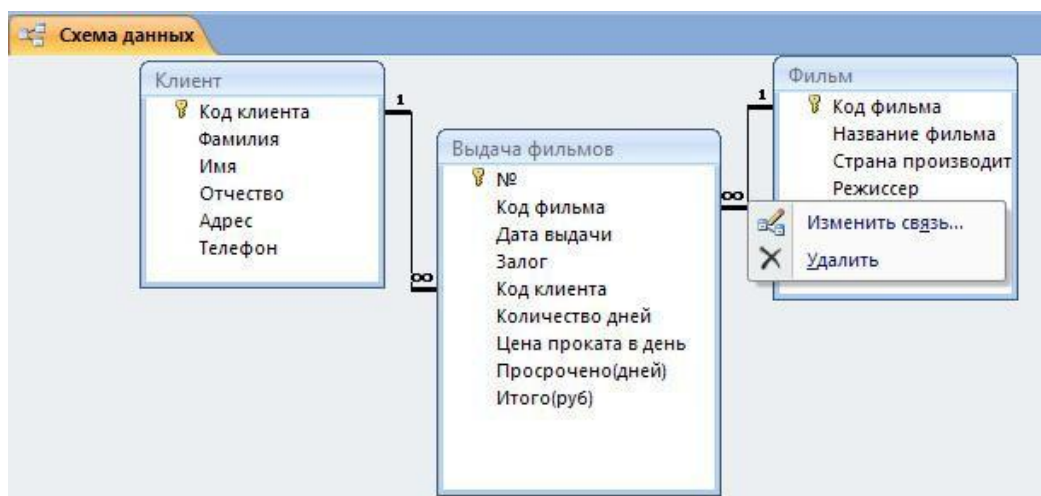


Рис. 50

Для удаления связей достаточно выделить ее и нажать на клавишу «Delete» или вызвать контекстное меню и выбрать команду «Удалить».

Примечание: Для того, чтобы связь один-ко-многим была создана правильно, при заполнении таблиц в режиме конструктора **внимательно** вводите данные в ключах, которые в последствие нужно связать, т.е. тип данных, количество знаков – все должно быть одинаковым, чтобы не нарушалось правило целостности, иначе связь будет установлена, но это будет связь не один-ко-многим, а значит, дальнейшая работа с базой данных не имеет смысла!

2.5 Проектирование форм

Форма – это наиболее удобное средство отображения данных. Форма удобна и наглядна, она использует графические возможности Windows (шрифты, графика и т.д.).

При конструировании формы определяется, из каких таблиц будут отображаться данные, пояснительный текст и т.д.

Для простоты создания формы воспользуемся возможностью автоматического создания автоформы в столбец (можно использовать и «Мастер форм»).

Выберем таблицу ФИЛЬМ, после выполним команду «Создание» - «Форма» появится окно «Form1». Появится форма для таблицы ФИЛЬМ (Рис. 51, Рис. 52):

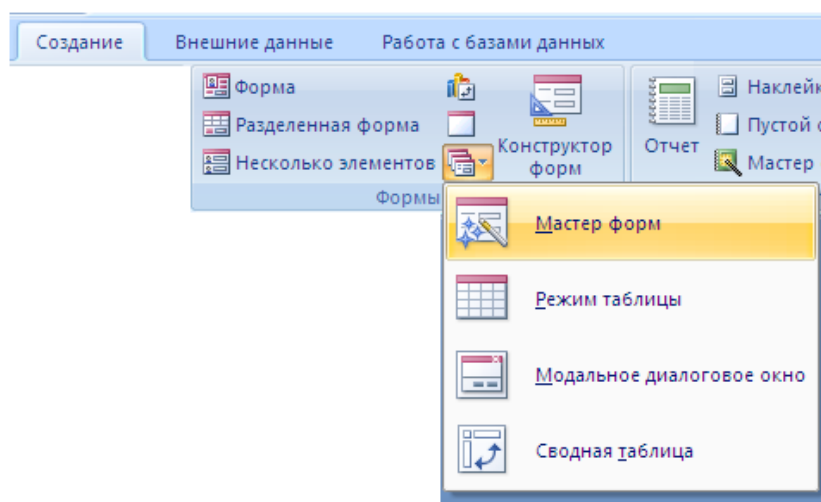


Рис. 51

Фильм

Фильм

Код фильма:

Название фильма:

Страна производитель:

Режиссер:

Время показа:

Жанр:

Рис. 52

Доработаем эту форму в режиме «Конструктора» (Рис. 53):

Фильм

Заголовок формы

Область данных

Код фильма:

Название фильма:

Страна производитель:

Режиссер:

Время показа:

Жанр:

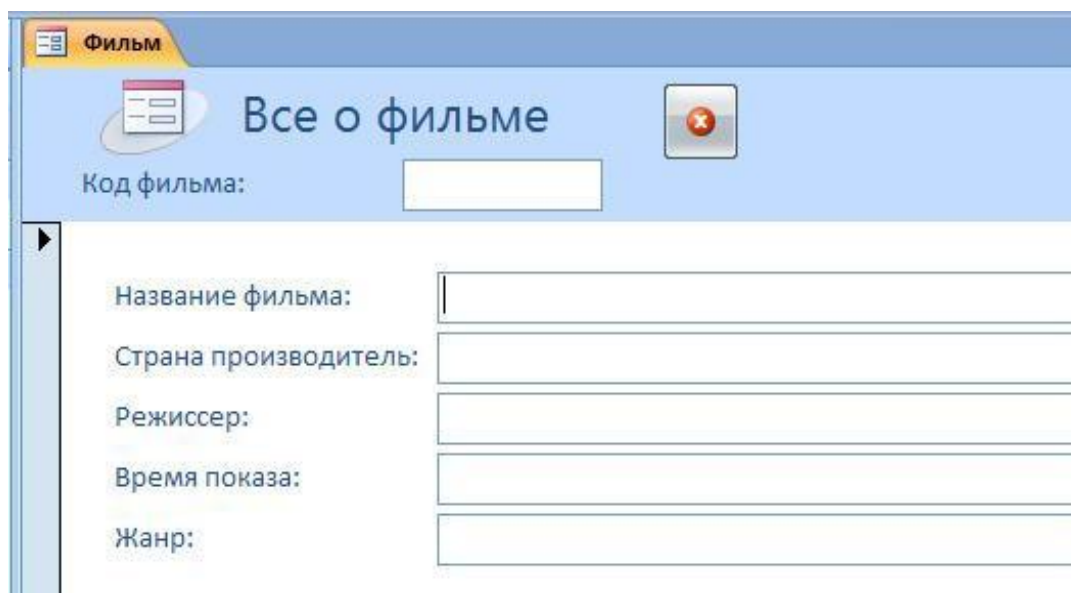
Рис. 53

Вызовем "Панель элементов" . Раздвинем "Заголовок формы" и "Область данных" и на освободившееся место с помощью элемента  сделаем надпись "Все о фильме". Выделив эту область, отформатируем надпись.

Поле "Код фильма" перетянем в область "Заголовок формы" (Либо используя команду Вырезать /Вставить).

Добавим в форму кнопку «Закрѳтия формы», воспользовавшись «Панелью элементов».

- Нажмем на пиктограмму «Кнопка»  и вычертим прямоугольник на поле формы.
- После рисования кнопки появилось окно «Создание кнопок».
- Выбираем «Категории» - «Работа с формой» и «Действия» - «Закрѳть форму» - «Далее»; Получится следующая форма, готовая к заполнению (Рис. 54).



Фильм

Все о фильме

Код фильма:

Название фильма:

Страна производитель:

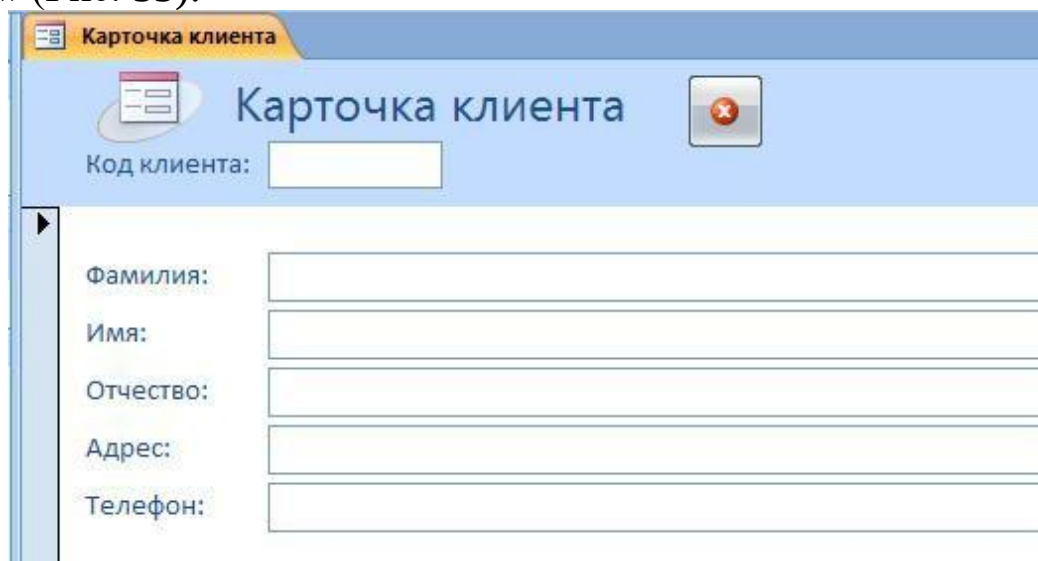
Режиссер:

Время показа:

Жанр:

Рис. 54

Аналогично создадим форму для заполнения таблицы «КЛИЕНТ» (Рис. 55):



Карточка клиента

Карточка клиента

Код клиента:

Фамилия:

Имя:

Отчество:

Адрес:

Телефон:

Рис. 55

Заполните через форму «ВСЕ О ФИЛЬМЕ» таблицу «ФИЛЬМ» (Рис. 56):

Фильм						
	Код фильма	Название ф	Страна про	Режиссер	Время пока	Жанр
+	1	Авиатор	США	Мартин Кросе	129	Драма
+	2	Вокруг света з	США	Френк Корачи	134	Приключения
+	3	Жених напро	США	Клэр Клинер	90	Комедия
+	4	Бумер-2	Россия	Вдовиченко	100	Триллер

Рис. 56

Заполните через форму «КАРТОЧКА КЛИЕНТА» таблицу «КЛИЕНТ» (Рис. 57)

Клиент						
	Код клиент	Фамилия	Имя	Отчество	Адрес	Телефон
+	1	Иванов	Петр	Иванович	Москва	1234568
+	2	Петрова	Ксения	Васильевна	Москва	2255324
+	3	Назаров	Илья	Алексеевич	Москва	2467909
+	4	Лебо	Алексей	Иванович	Москва	2567090

Рис. 57

Создадим форму для заполнения таблицы «ВЫДАЧА ФИЛЬМОВ» (Рис. 58):

Выдача фильмов

Бланк выдачи фильмов

№:

Код фильма:

Дата выдачи:

Залог:

Код клиента:

Количество дней:

Цена проката в день:

Просрочено(дней):

Итого(руб):

Рис. 58

Доработаем форму «**БЛАНК ВЫДАЧИ ФИЛЬМОВ**»:

- Вместо поля «**Код фильма**» укажем название фильма посредством поля со списком. Включим дополнительные поля для фильма: время показа, жанр фильма и режиссера.

- Вместо поля «**Код клиента**» укажем фамилию клиента посредством поля со списком. Включим дополнительные поля для клиентов – укажем имя и телефон.

В режиме конструктора преобразуем поле «**Код фильма**» в поле со списком, т.е. вместо самого кода будем выбирать из списка наименование фильма. В таблицу же будет записываться код:

1. Удалим поле «**Код фильма**» (вместе с надписью и окном ввода).

2. На панели элементов управления выбираем инструмент «**Поле со списком**» .

3. Вычерчиваем на освободившемся месте новое поле. После этого заработает мастер и появится последовательность из 5 диалоговых окон:

4. В первом окне проверяем способ получения значений для поля со списком, должен быть подключен способ «**Объект «поле со**

списком» будет использовать значения из таблицы или запроса» - «Далее» (Рис. 59);

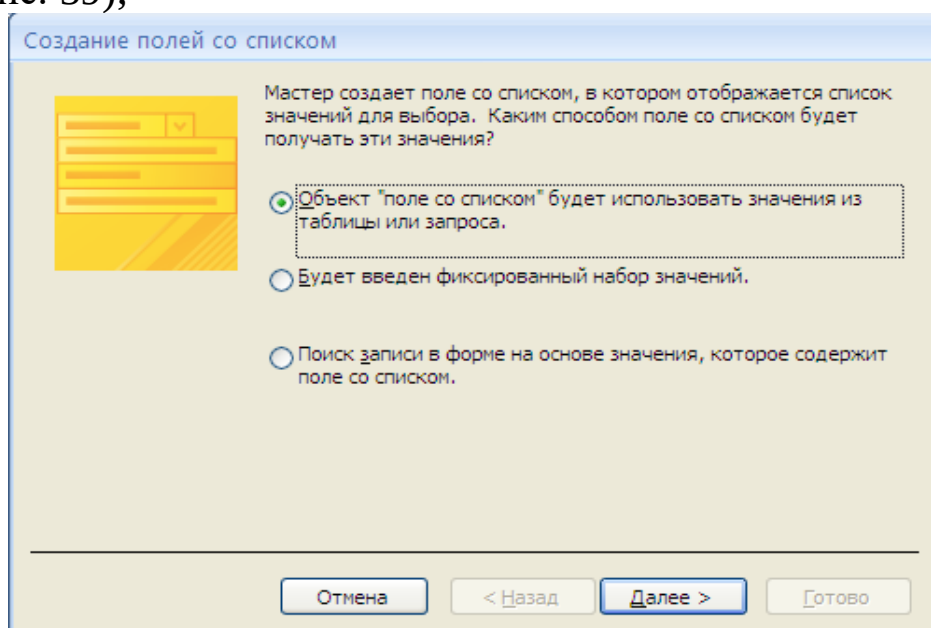


Рис. 59

5. Во втором окне выбираем таблицу из которой будем брать значения поля со списком, так как мы удалили поле «Код фильма», то таблицу должны выбрать «ФИЛЬМ» - «Далее» (Рис. 60);

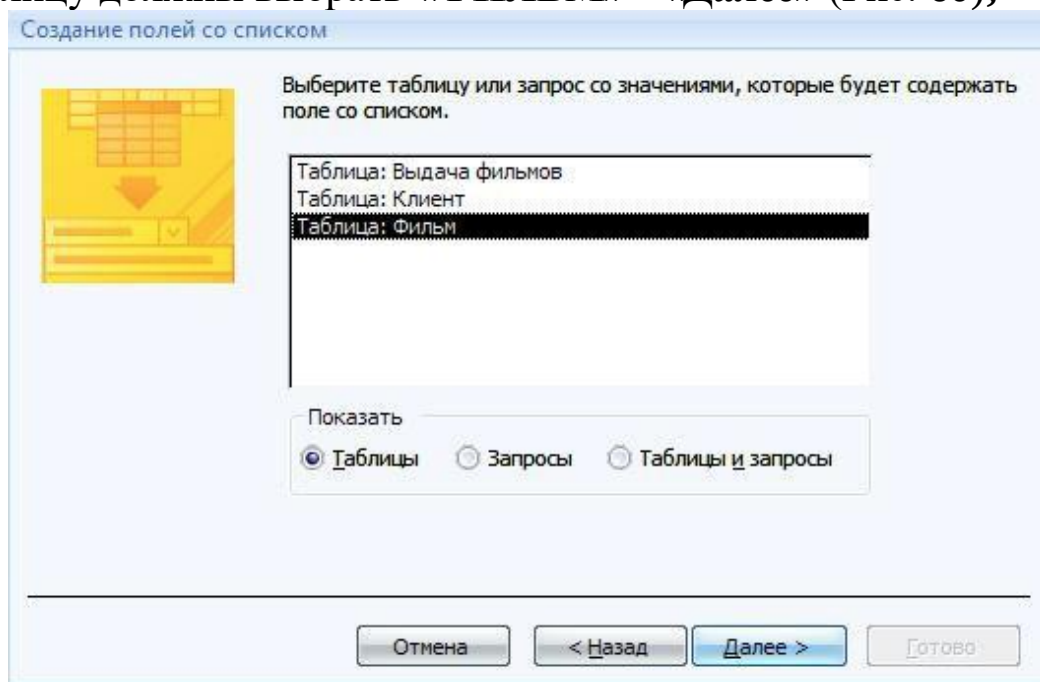


Рис. 60

6. В третьем окне перенесем поле «Код фильма» и «Название фильма» из окна «Доступные поля» в окно «Выбранные поля» с помощью кнопки  (Рис. 61);

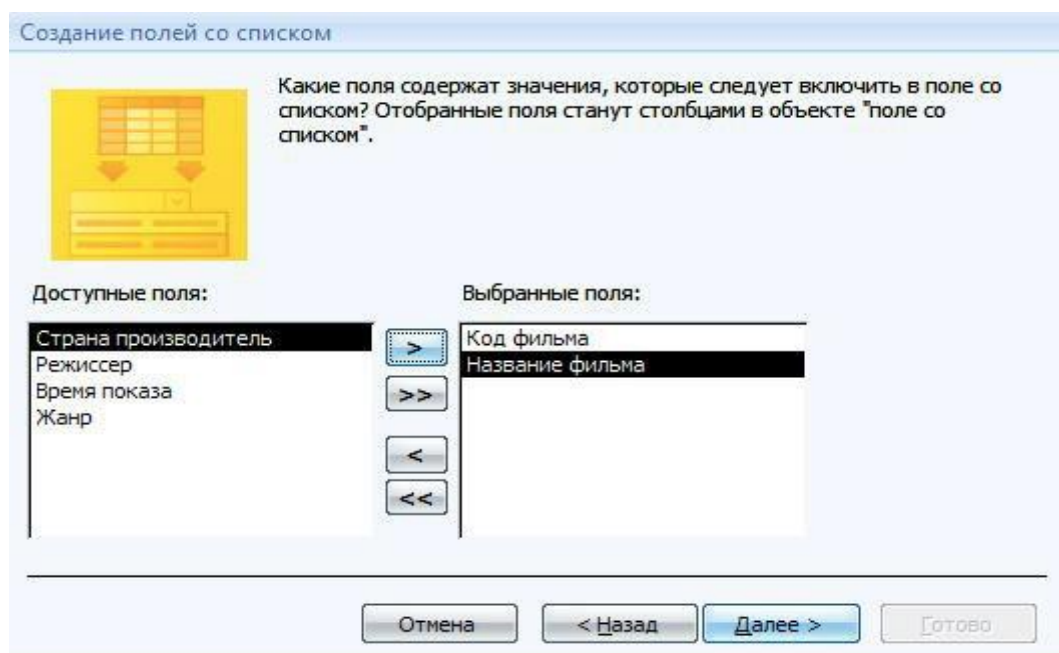


Рис. 61

В четвертом и пятом окне выбираем порядок сортировки фильмов и нужно ли скрывать ключевой столбец - «Далее» (Рис. 62, Рис. 63);

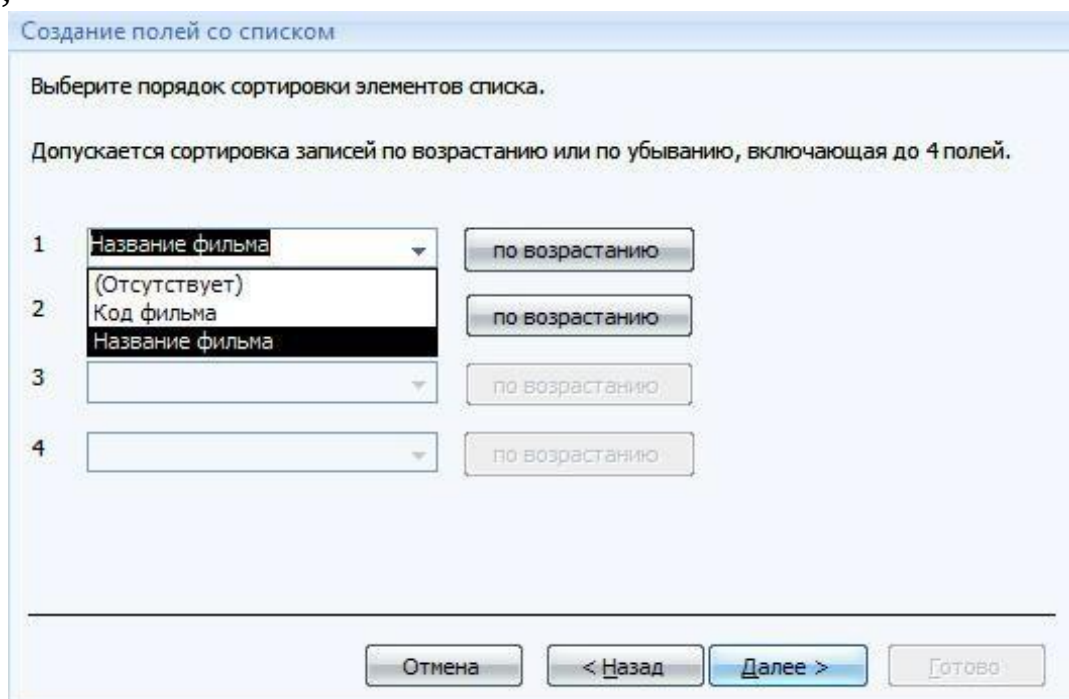


Рис. 62

Создание полей со списком

Задайте ширину столбцов, которые содержит поле со списком.

Перетащите правую границу заголовка столбца на нужную ширину или дважды щелкните ее для автоматического подбора ширины.

☒ Скрыть ключевой столбец (рекомендуется)

Название фильма					
Авиатор					
Бумер-2					
Вокруг света за 81					
Жених напрокат					

Отмена < Назад Далее > Готово

Рис. 63

7. В шестом окне указываем, где сохранить выбранное из объекта поле со списком - в поле «Код фильма» - «Далее» (Рис. 64);

Создание полей со списком

Microsoft Office Access позволяет сохранить выбранное из объекта "поле со списком" значение в базе данных или использовать это значение в дальнейшем для выполнения действия. Какое действие будет выполняться при выборе значения из объекта "поле со списком"?

☐ Запомнить значение.

☒ Сохранить в поле: Код фильма

Отмена < Назад Далее > Готово

Рис. 64

8. В седьмом окне задаем подпись «**Название фильма**» - «**Готово**» (Рис. 65).

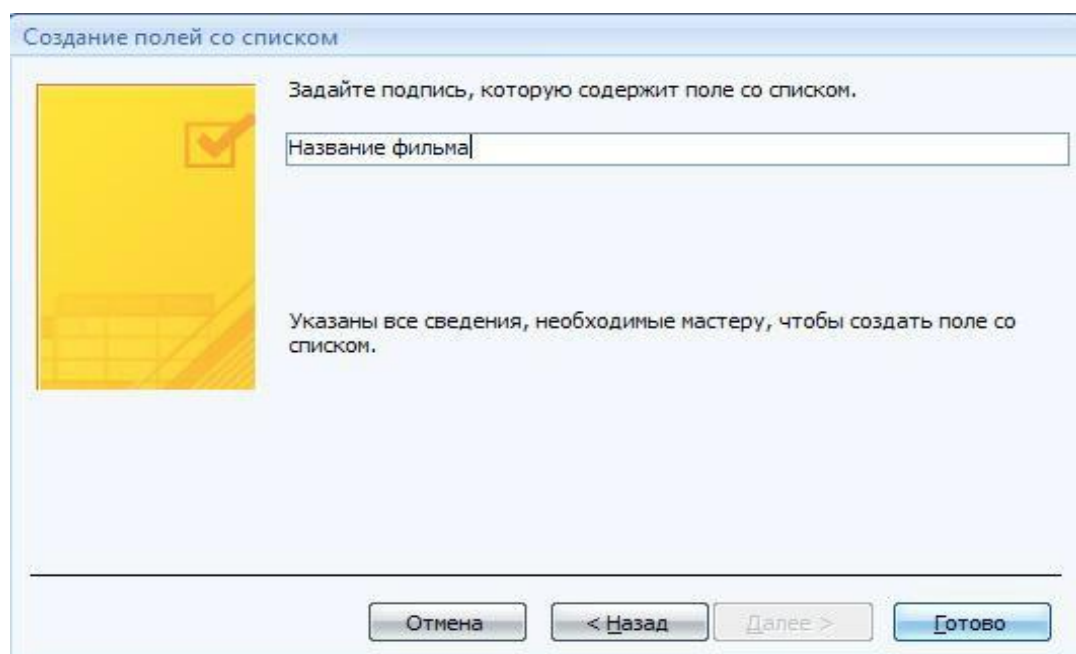


Рис. 65

Форма в режиме «**Конструктора**» будет выглядеть следующим образом (Рис. 66):

Рис. 66

Проверяем работу «Поля со списком» в режиме формы (Рис. 67):

Рис. 67

Теперь включим дополнительные поля для фильма – укажем время показа, жанр фильма и режиссера. Данные в этих полях будут появляться автоматически при смене названия фильма:

- В режиме конструктора посредством нажатия правой клавиши мыши на черный квадрат в левом верхнем углу формы, выбираем команду **«Свойства»** (Рис. 68):

-

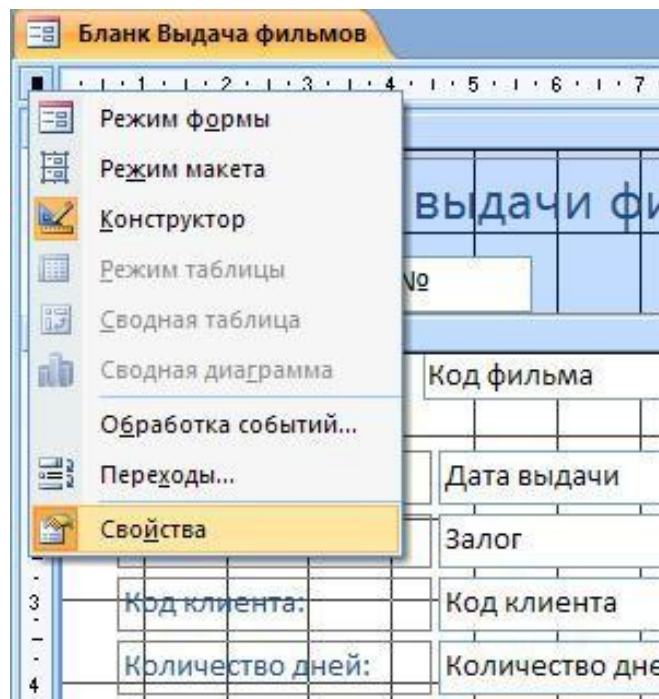


Рис. 68

- В появившемся диалоговом окне **«Форма»** (Рис.69) на закладке **«Данные»** выбираем строку **«Источник записей»** и вызываем **«построитель запросов»** – кнопка .

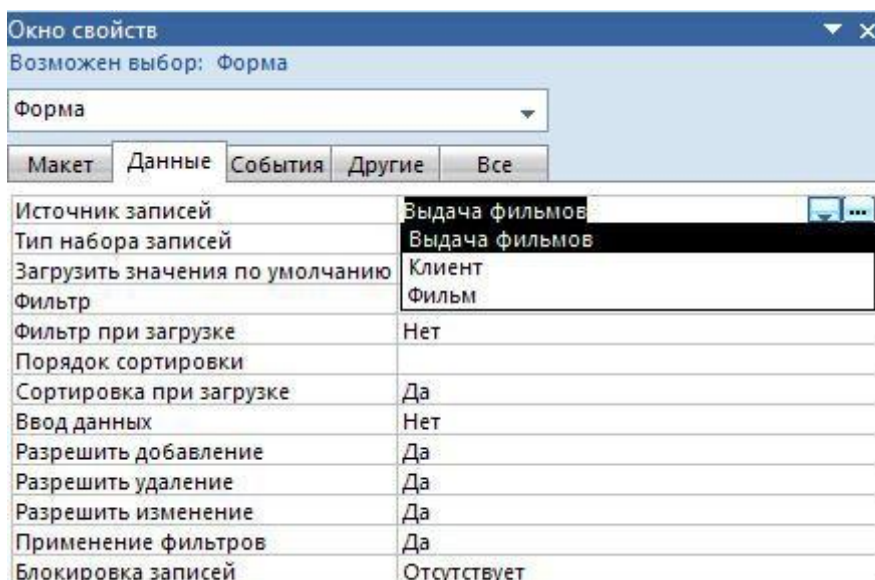


Рис. 69

- Подтверждаем создание запроса на основе этой таблицы (Рис. 70).

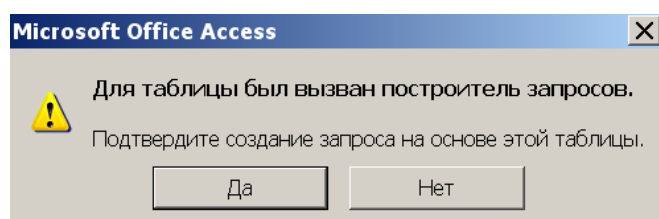


Рис.70

- Выбрав пиктограмму «Отобразить таблицу»  добавляем в рабочее поле построителя таблицу «ФИЛЬМ», т.к. из нее будем брать дополнительные данные, а так же добавляем таблицу «Клиент».

Примечание:

- 1) при построении должны быть связи один-ко-многим, впоследствии эти связи не отображаются.
- 2) Добавить необходимо все таблицы, в противном случае в форме работать будет только часть запроса.

- Перетащим в графу «Поле» все поля из таблицы «ВЫДАЧА ФИЛЬМОВ» и поля «Время показа», «Жанр фильма» и «Режиссер» из таблицы «ФИЛЬМ» (т.е. те поля, которые мы планируем вывести в дополнительные поля формы). Аналогично

включим дополнительные поля для клиентов из таблицы «КЛИЕНТ»
- укажем имя и телефон (Рис. 71)

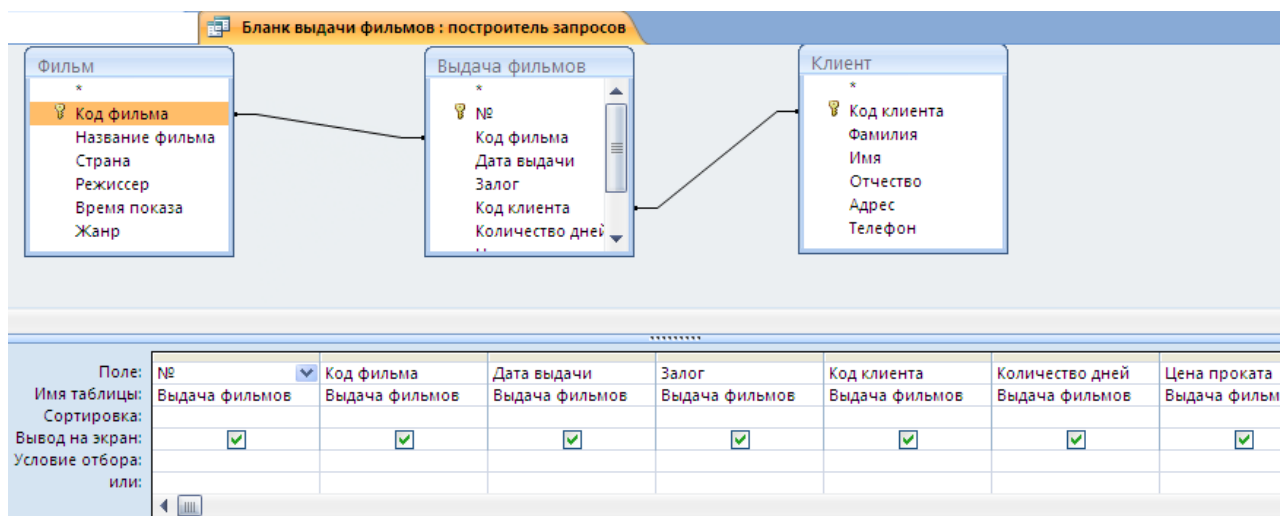


Рис. 71

Сохраним изменения (Рис.72):

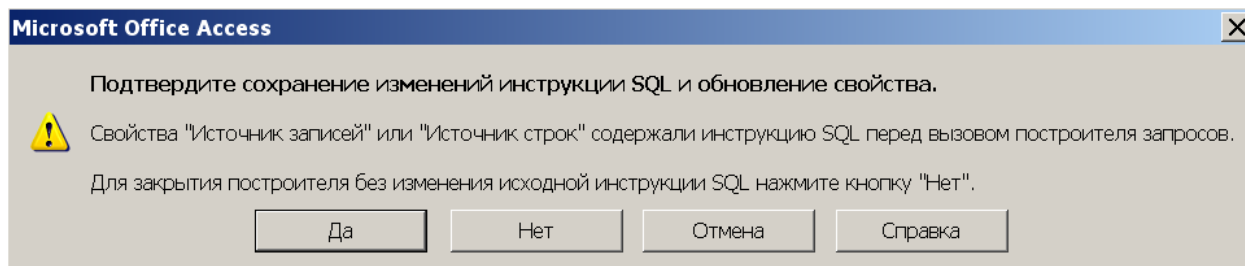


Рис.72

• Подключаем пиктограмму «Список полей»  - и перетягиваем нужные поля в форму «БЛАНК ВЫДАЧИ ФИЛЬМОВ» и красиво размещаем их на рабочем поле. Создаем кнопку для закрытия формы. (Рис.73):

The screenshot shows a software interface for a movie rental system. At the top, there is a tab labeled 'Бланк выдачи фильмов'. Below the tab is a header area with the title 'Бланк выдачи фильмов' and two input fields for '№'. The main form is divided into two sections. The top section contains fields for 'Клиент' (Client), 'Код клиента' (Client code), 'Имя' (Name), and 'Телефон' (Phone). The bottom section, titled 'Область данных' (Data area), contains fields for 'Название фильма' (Movie name), 'Код фильма' (Movie code), 'Режиссер' (Director), 'Дата выдачи' (Issue date), 'Время показа' (Show time), 'Залог' (Deposit), 'Жанр' (Genre), 'Количество дней' (Number of days), 'Цена проката' (Rental price), and 'Просрочено(дней)' (Overdue in days). A red 'X' icon is visible in the top right corner of the form.

Рис.73

Проверяем нашу работу (Рис. 74):

The screenshot shows the same movie rental form as in Figure 73, but with data entered. The 'Клиент' field is set to 'Иванов', 'Имя' is 'Петр', and 'Телефон' is '1234567'. In the 'Область данных' section, 'Название фильма' is 'Бумер-2', 'Режиссер' is 'Петр Буслов', 'Время показа' is '100', and 'Жанр' is 'Триллер'. The 'Дата выдачи', 'Залог', 'Количество дней', 'Цена проката', and 'Просрочено(дней)' fields are empty. A red 'X' icon is visible in the top right corner of the form.

Рис.74

2.6 Макросы

В нашей задаче мы будем программировать макрос подсчета итоговой суммы, которую должен заплатить клиент за прокат

фильма. Эта цена будет считаться по формуле: **(Количество дней + Просрочено)*Цена проката в сутки – Залог**

2.6.1 Процедура создания макроса

- В окне БД выбираем «Создание» - «Макрос».
- Выбираем Макрокоманду «Задать временную переменную» (Рис. 75).

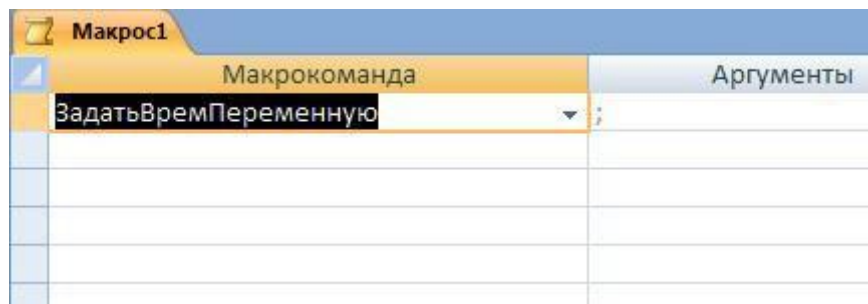


Рис. 75

- Выбираем Аргумент макрокоманды «Элемент» и в «Построителе выражений» [...] выполняем последовательность действий:

- указателем мыши щелкаем 2 раза по команде «Forms»;
- указателем мыши щелкаем 2 раза по команде «Все формы»;
- выбираем форму «БЛАНК ВЫДАЧИ ФИЛЬМОВ» ;
- в среднем окне щелкаем 2 раза по полю «Итого». (Рис. 75)

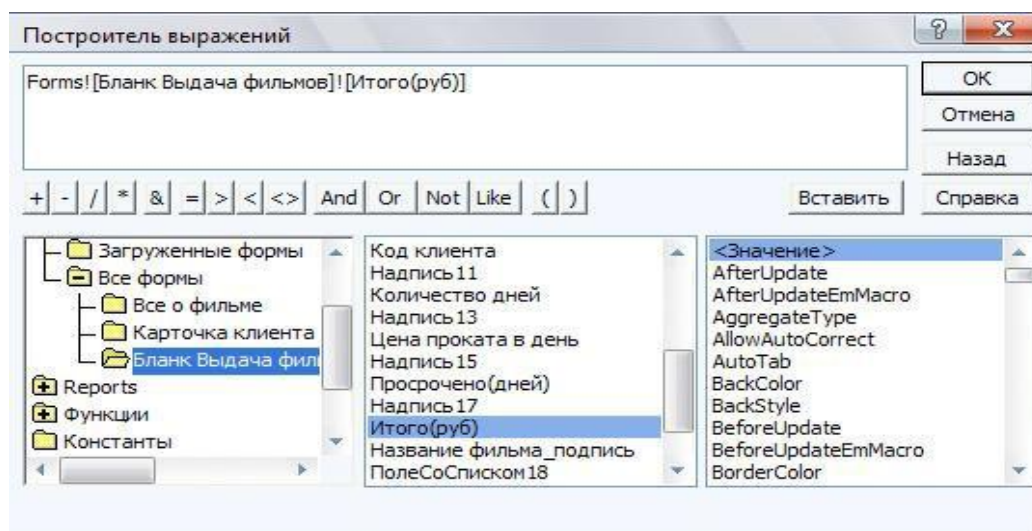


Рис. 75

Выбираем Аргумент макрокоманды **«Выражение»** и с помощью **«построителя выражения»** повторяем те же действия: **«Forms»** - **«Все формы»** - **«БЛАНК ВЫДАЧИ ФИЛЬМОВ»** и записываем формулу подсчета итоговой суммы (Рис. 76):

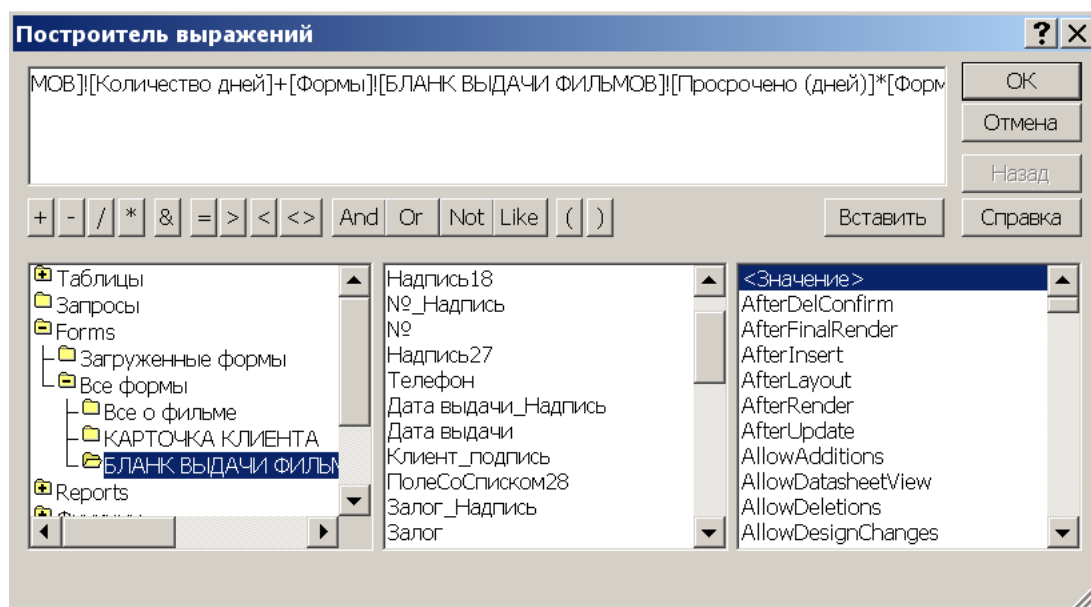
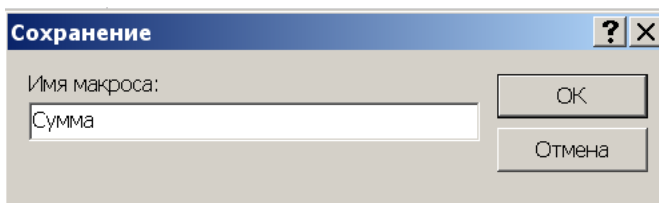


Рис. 76

Сохраняем макрос под именем **«Сумма»** :



Макрос готов. Теперь надо его привязать к полям формы, которые использовались для данного макроса.

Займемся подключением макроса.

2.6.2 Подключение макросов

Срабатывание макроса можно выполнить по нажатию кнопки (которую необходимо поместить на форму) и соответственно определив **«Категорию кнопки»** - **«Разное»**, **«Действие»** - **«Выполнить макрос «Сумма»** (Рис.77).

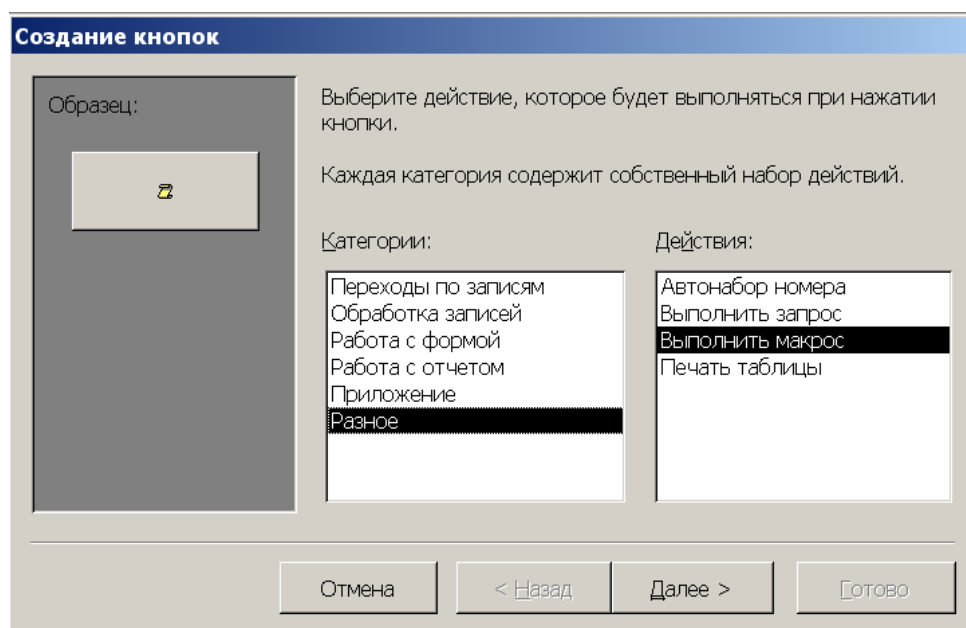


Рис.77

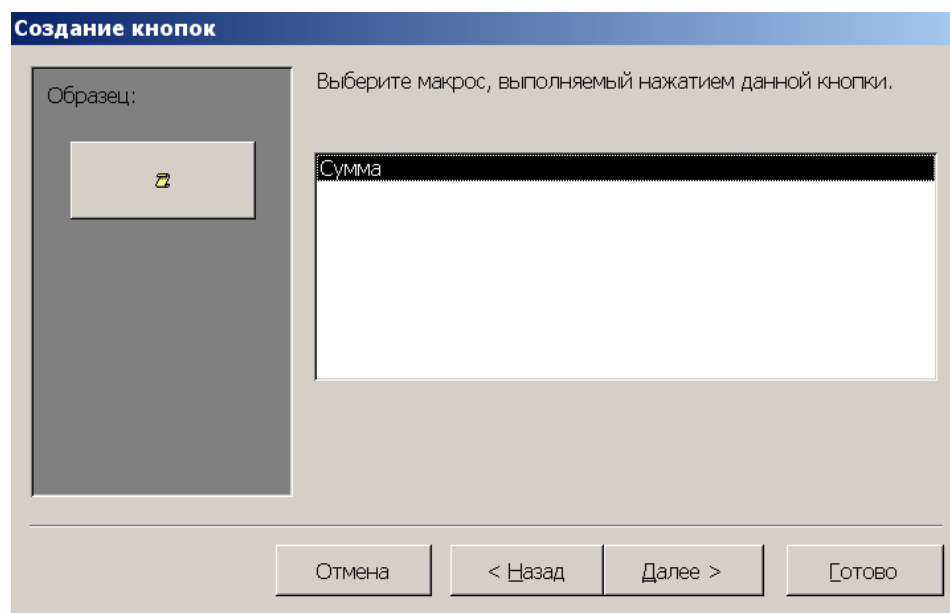


Рис. 78

Форма с макросом СУММА (Рис. 79):

ВЫДАЧА ФИЛЬМОВ

БЛАНК ВЫДАЧИ ФИЛЬМОВ № Дата выдачи

Клиент

Название фильма Режиссер:

Залог Время показа:

Количество дней Жанр:

Цена проката в день

Просрочено (дней)

Итого (руб):

Запись: 1 из 1

Рис.79

Можно выполнить срабатывание макроса автоматически, связав его с изменением событий, от которых зависит макрос. Для этого надо выполнить последовательность действий:

- открыть форму «Бланк выдачи фильмов»;
 - подключить в «Конструктор»;
 - поставить указатель на поле «Итого» и вызвать контекстное меню;
 - выбрать команду «Свойства»;
 - подключить вкладку «События»;
 - в поле «После обновления» подключить макрос «Сумма»
- (Рис. 80).

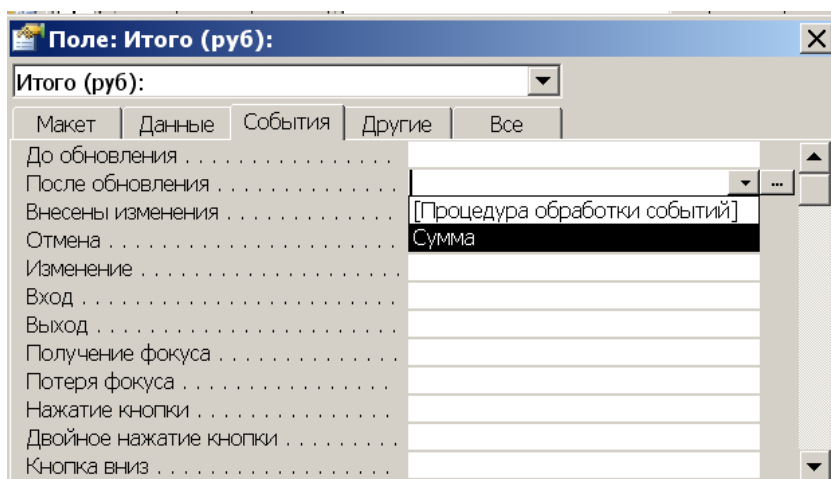


Рис.80

То же самое проделать с полями «Залог», «Количество дней», «Просрочено», «Цена проката в сутки».

Вновь открыть форму «БЛАНК ВЫДАЧИ ФИЛЬМОВ» и проверить работу макроса.

2.7 Запросы

Одной из основных функций обработки данных является поиск данных. Для выборки и просмотра записей из одной или нескольких таблиц предназначены **Запросы**. Их можно сохранять и использовать, позднее, создавать на их основе формы или отчёты. Запросы имеют возможность выбора полей для отображения из основных и связанных таблиц. Это отличает их от фильтров и делает основным инструментом обработки данных в СУБД. В зависимости от функции, которую выполняет запрос, различают:

- запросы на выборку;
- запросы на изменение;
- запросы с параметрами;
- запросы перекрёстные;
- запросы с вычисляемыми полями;
- запросы на добавление записей из одной таблицы в другую;
- запросы на обновление записей;
- запросы на удаление записей.

Последние шесть запросов образуют группу сложных или специальных запросов, используемых для глобального редактирования.

Для создания запросов можно пользоваться Мастером запросов. Он ускоряет процесс создания, однако более подходит для наиболее простых запросов типового характера. Наиболее гибкий способ создания запросов – это режим конструктора. Он позволяет проектировать запрос в соответствии с требованием пользователя.

2.7.1 Создание запроса на выборку

Запросы этого типа предназначены для извлечения записей и полей, удовлетворяющих некоторому условию, из одной или нескольких таблиц и просмотра результата в режиме таблицы.

Создадим несложный запрос на основе таблицы «ФИЛЬМ», который выполняет выборку записей фильмах определенного режиссера, например **Klara Kilnera**.

Для этого в окне базы данных выберем объект «Запросы» и строку «Создание запроса» в режиме «Конструктор запросов» (Рис. 81).

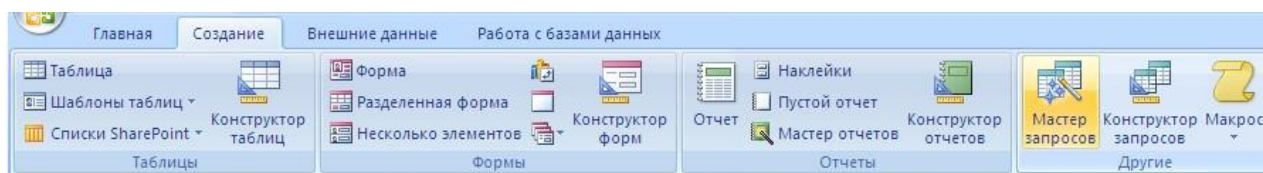


Рис. 81

После этого открывается окно **«Добавление таблицы»**. Выберем таблицу **«Фильм»**, нажмем кнопку **«Добавить»** и затем **«Закрыть»**. Окно запроса будет выглядеть следующим образом (Рис. 82):

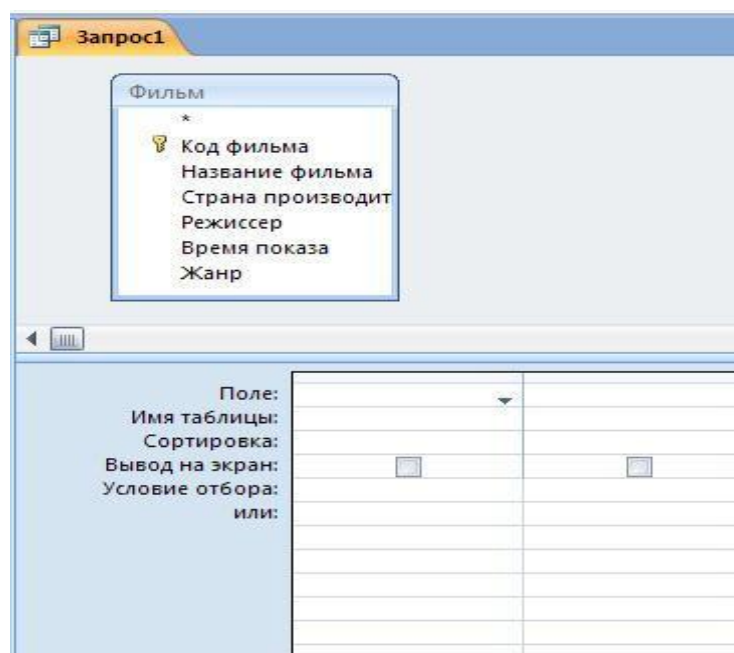


Рис.82

После добавления таблицы в окно запроса нужно добавить поля в бланк запроса. Для этого нужно перетащить поле **«Режиссер»** из списка полей в верхней части окна в строку поле первого столбца бланка запроса. Аналогично перенести остальные поля.

Затем надо ввести условие отбора записей в столбец бланка запроса с полем **«Режиссер»** в строку **«Условие отбора»**. В нашем примере это первый столбец. Введём в строку **«Условие»** отбора Клэр Килнер (Рис. 83).

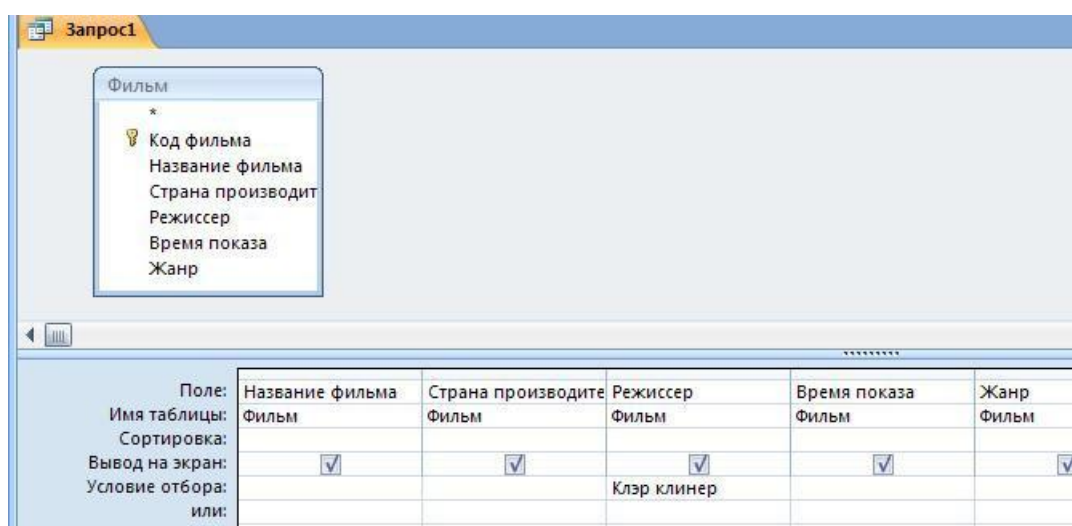
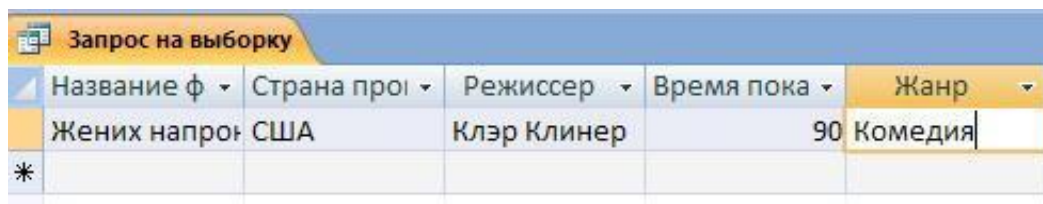


Рис. 83

Для того, чтобы выполнить запрос нужно нажать на панели конструктора кнопку **«Запуск»** или кнопку **«Вид»** на панели

инструментов. Результаты выполнения запроса будут представлены в виде таблицы (Рис. 84):



Название ф ▾	Страна прои ▾	Режиссер ▾	Время пока ▾	Жанр ▾
Жених напро	США	Клэр Клинер	90	Комедия
*				

Рис. 84

Процесс выбора полей для многотабличного запроса точно такой же, как и для запроса с одной таблицей.

2.7.2 Параметрический запрос

Рассмотренные запросы на выборку могут выполняться только с условием, указанным в конструкторе. Для получения сведений по данному полю с разными значениями необходимо использовать **параметрический запрос**. После запуска такого запроса выдаётся диалоговое окно и запросом параметра. Пользователь вводит значение параметра, после чего запрос работает аналогично рассмотренным выше. Построим параметрический запрос **«Информация о фильме»**:

- откроем объект **«Запросы»**;
- выберем команду **«Создать в режиме конструктора»**;
- добавим таблицу **«Фильм»**;
- перенесём во вторую половину окна **«Конструктора запросов»** все поля из таблицы;
- в столбце с полем **«Название фильма»** в строку **«Условие отбора»** введём **[Ведите название фильма]** (Фраза в квадратных скобках);

Окно конструктора при этом будет выглядеть следующим образом (Рис. 85):

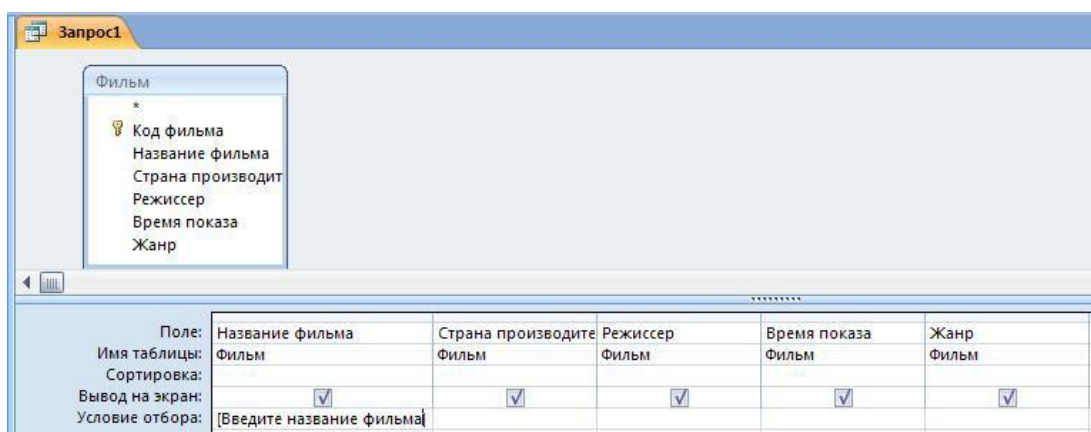


Рис.85

Такой запрос может неоднократно использоваться для всех фильмов из БД.

2.7.3 Перекрёстный запрос

Один из наиболее интересных запросов – перекрёстный. Он представляет собой таблицу, на пересечении столбцов и строчек которой лежат сведения необходимые пользователю.

Посмотрим, например кто из клиентов не вернул вовремя диск, название фильма, дату выдачи сколько дней просрочен возврат. Для начала построим базовый запрос, содержащий необходимую информацию для перекрёстного запроса:

- откроем объект «Запросы»;
- выберем команду «Создать в режиме конструктора»;
- добавим таблицы «ФИЛЬМ», «КЛИЕНТ», «ВЫДАЧА ФИЛЬМОВ»;
- перенесём во вторую половину окна «Конструктора запросов» необходимые поля из таблиц:
 - поля «Фамилия», «Телефон» из таблицы «КЛИЕНТ»;
 - поле «Название фильма» из таблицы «ФИЛЬМ»;
 - поля «Дата выдачи», «Просрочено» из таблицы «ВЫДАЧА ФИЛЬМОВ»;
- по полю «Просрочено» введём условие отбора;

- сохраним запрос с именем **«Базовый запрос»**. Выполним этот запрос:

Базовый запрос является основой для перекрёстного запроса. Следующий шаг – создание самого перекрёстного запроса:

- выберем команду **«Создать» - «Перекрёстный запрос»;**
- в качестве источника укажем **«Запросы» - «Базовый запрос»;**
- выберем в качестве заголовков строк значения следующих полей: **«Фамилия», «Телефон», «Дата выдачи»;**
- выберем в качестве заголовков столбцов значения поля **«Название фильма»;**
- в ячейках, находящихся на пересечении вычислим сумму для значений поля **«Просрочено (дней)»;**
- сохраним запрос (Рис. 86).

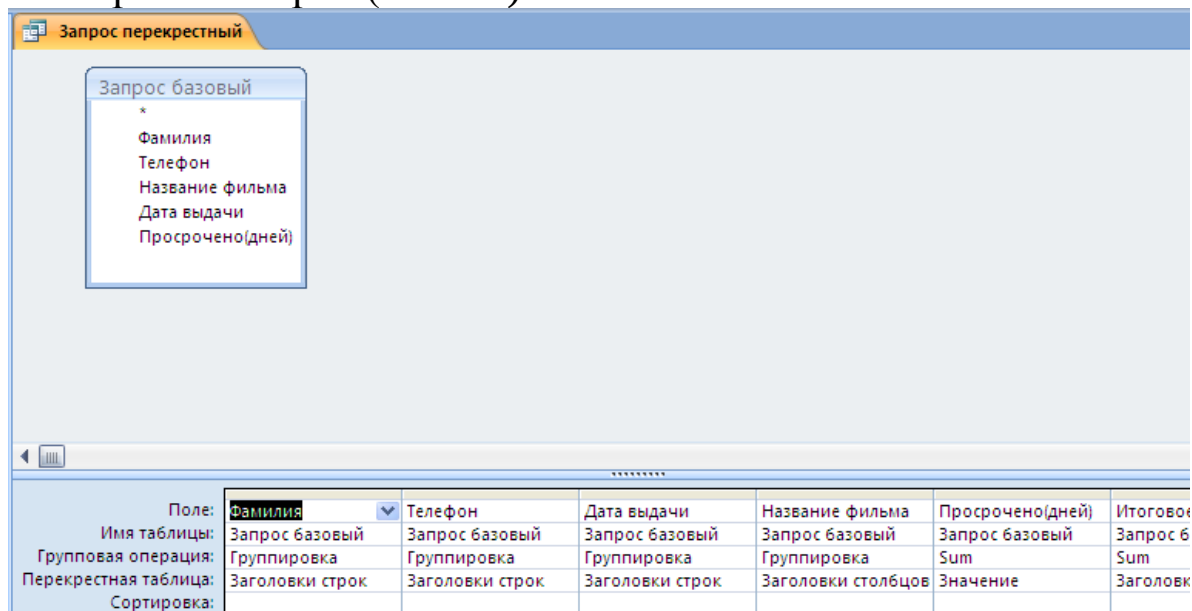


Рис. 86

Следует отметить, что итоговое значение вычисляется для значений, выбранных в качестве заголовков строк.

2.8 Отчеты

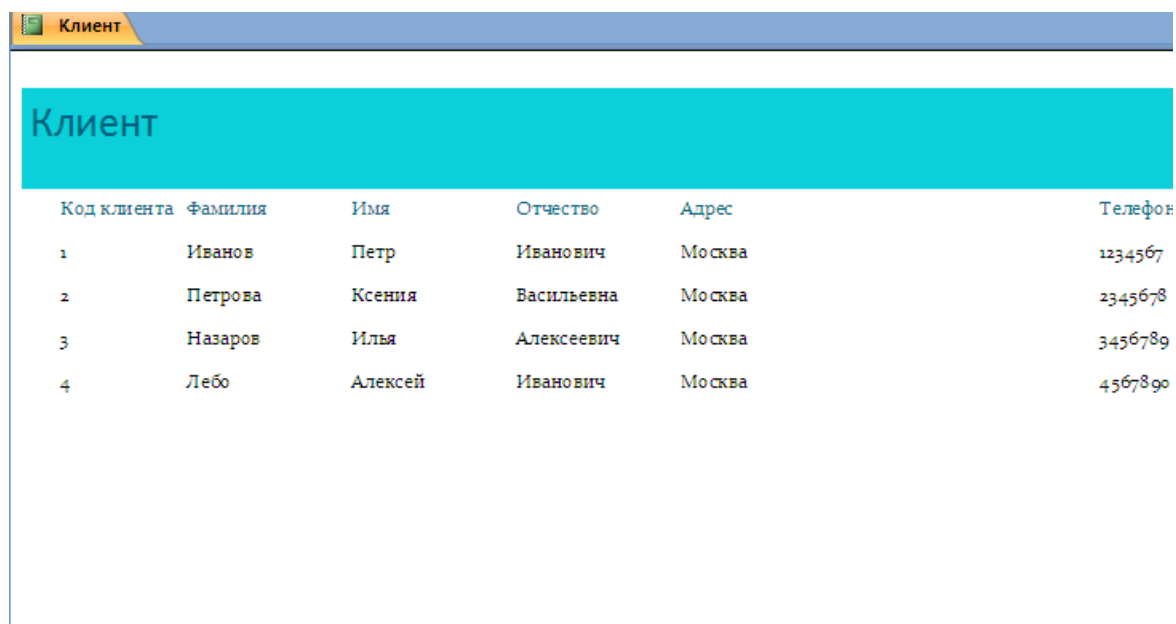
Для вывода поисковой информации на бумагу используется объект **«Отчеты»**. Варианты создания отчета:

Конструктор – отчет создается самостоятельно.

Мастер отчетов – отчет создается автоматически на основе выбранных полей.

Автоотчет – работает быстрее, чем мастер. Надо указать имя таблицы или запроса и при желании выбрать брать тип автоотчета - ленточный или в столбец.

Автоотчет в столбец – отображает записи из таблицы или запроса и располагает их по столбцам. Оформление отчета носит чисто функциональный характер и отличается простотой (Рис.87):



The screenshot shows a software window titled 'Клиент' with a light blue header bar. Below the header is a table with a light blue background. The table has six columns: 'Код клиента', 'Фамилия', 'Имя', 'Отчество', 'Адрес', and 'Телефон'. There are four rows of data.

Код клиента	Фамилия	Имя	Отчество	Адрес	Телефон
1	Иванов	Петр	Иванович	Москва	1234567
2	Петрова	Ксения	Васильевна	Москва	2345678
3	Назаров	Илья	Алексеевич	Москва	3456789
4	Лебо	Алексей	Иванович	Москва	4567890

Рис. 87

Автоотчет ленточный подходит для отчета с небольшим количеством полей:

Мастер диаграмм – создается отчет с диаграммой:

5 Выбираем поля с данными, которые надо отобразить в диаграмме:

Выбираем тип диаграммы для представления наглядности полей:

Над отчетами можно поработать в окне конструктора.

Создание наклеек – создаются стандартные или пользовательские наклейки.

2.9 Главная кнопочная форма

Кнопочные формы это специальные формы, выполняющие функции меню. Они представляют собой формы с кнопками, при помощи которых можно открывать обычные формы или же кнопочные. В последнем случае мы говорим о **иерархической** кнопочной форме. В Access существует специальное средство, облегчающее создание кнопочных форм - **«Диспетчер кнопочных форм (Switchboard Manager)»**. Для его запуска нужно выполнить команды:

«Сервис» - «Служебные программы» - «Диспетчер кнопочных форм».

Каждая кнопочная форма в приложении называется **страницей** главной кнопочной формы. Естественно, что первой страницей, которую добавляет диспетчер в диалоговое окно после ответа «Да» на предложение **«Создать кнопочную форму?»**, является страница **Main Switchboard**.

Далее необходимо добавить в эту страницу элементы – кнопки. Для этого нужно выделить эту страницу и щёлкнуть по кнопке **«Изменить»**. После этого открывается окно **«Изменение страницы кнопочной формы»**. Чтобы добавить элемент нужно щёлкнуть по кнопке **«Создать»**.

В поле **«Текст»** нужно набрать имя элемента. Например: **«Новый фильм»**.

В раскрывающемся списке **«Команда»** выбрать команду **«Открыть форму для добавления»**.

В раскрывающемся списке **«Форма»** выбрать форму **«ФИЛЬМ»**.

В результате описанных действий создастся кнопка, при нажатии на которую будет осуществляться добавление нового фильма в базу данных.

Аналогично, можно создать кнопки для добавления нового клиента, нового бланка выдачи.

- В главной кнопочной форме могут содержаться страницы **«Запросы»** или **«Отчеты»** для выполнения функций печати или просмотра запросов. Для этого необходимо:
- в диалоговом окне **«Диспетчер кнопочных форм (Switchboard Manager)»** щёлкнуть по кнопке **«Создать»**;

- в диалоговом окне «Создание» в поле **«Имя страницы кнопочной формы»** ввести текст, например, **«Запросы»**;
- щёлкнуть по кнопке **«ОК»**.

После создания новой страницы в неё нужно добавить соответствующие элементы. Отметим, что для открытия запросов нужно предварительно создать макросы. Для этого нужно открыть объект **«Макросы»**, выбрать команду **«Создать»**, выбрать макрокоманду **«Открыть Запрос»**. Затем в поле **«Запрос»** указать имя запроса (например, параметрический), сохранить этот макрос с соответствующим именем **«Параметрический запрос»**.

Теперь добавим элемент в страницу **«Запросы»**. Для этого нужно:

- выделить страницу **«Запросы»** в окне **«Диспетчер кнопочных форм»**;
- щёлкнуть по кнопке **«Изменить»**;
- в открывшемся окне **«Изменение страницы кнопочной формы»** щелкнуть по кнопке **«Создать»**;
- в диалоговом окне **«Изменение элемента кнопочной формы»** в поле **«Текст»** ввести **«Параметрический запрос»**;
- в раскрывающемся списке **«Команда»** выбрать команду **«Выполнить Макрос»**;
- в раскрывающемся списке **«Макрос»** выбрать макрос **«Параметрический запрос»**;
- щёлкнуть по кнопке **«ОК»** для возврата к предыдущему диалоговому окну.

В результате описанных действий на странице запросы будет создана кнопка, при нажатии на которую открывается параметрический запрос. Однако, для перехода на страницу запросы нужно на странице **Main Switchboard** добавить элемент.

Следует предусмотреть кнопку на странице **«Запросы»** для возврата на страницу **Main Switchboard**, а также кнопку **«Выйти из приложения»**.

Созданная кнопочная форма разместится в объекте **«Формы»** с именем **«Switchboard»**. Её можно отредактировать, добавить графические объекты, переместить элементы, добавить гиперссылки.

В конечном варианте она может выглядеть следующим образом (Рис.88):

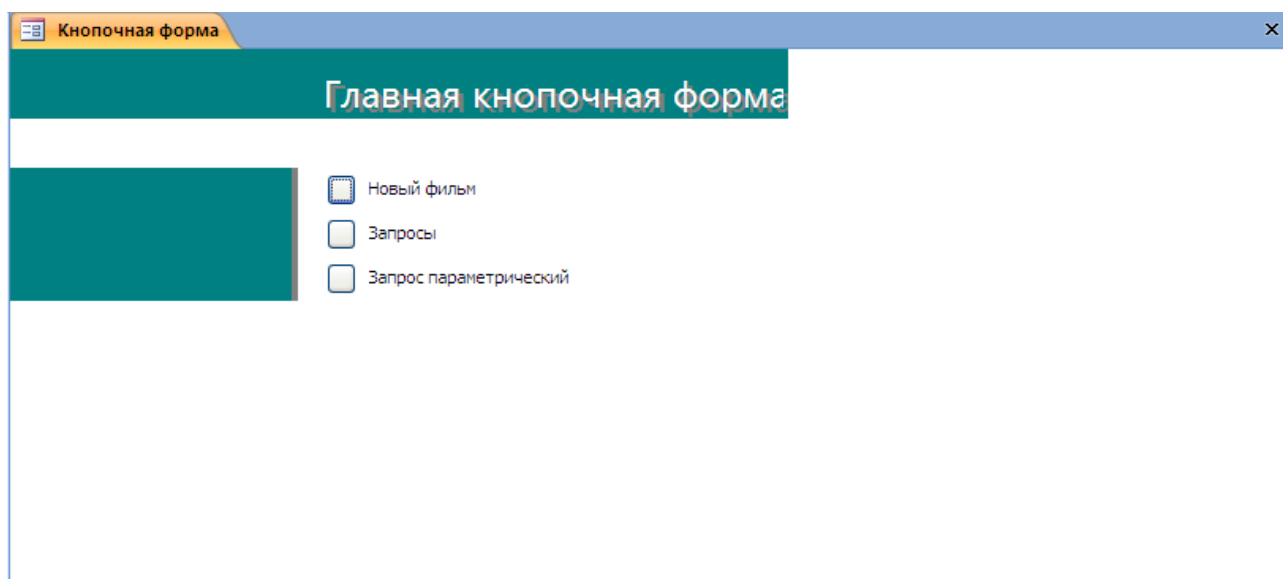


Рис.88

Для добавления графических объектов необходимо выполнить последовательность действий:

- «Объекты» - «Формы» - «Switchboard»;
- подключить Конструктор;
- на «Панели элементов» подключить пиктограмму «Рисунок» — и нарисовать прямоугольник на поле нашей формы. Появится окно «Выбор рисунка»;
- найти файл, в котором хранится рисунок и щелкнуть по нему 2 раза мышкой.

3.ГЛОССАРИЙ

Access	- система управления БД, работающая в среде Windows. Это универсальный комплекс прикладных программ, которые обслуживают и создают БД. Access -гибкая система, позволяющая работать с простыми и сложными БД, это реляционная БД, которая позволяет определить отношения между различными категориями информации
База данных (БД)	- совокупность специальным образом организованных наборов данных, хранимых, обрабатываемых и обновляемых средствами ЭВМ
Запись	- информация о конкретном экземпляре информационного объекта. Например, информация об одном сотруднике. В таблице каждая строка - это запись
Запрос	- выборка и просмотр нужных данных из одной или нескольких таблиц. Его можно сохранять, чтобы позднее вновь использовать или создать на его базе форму или отчет
Запрос на выборку	- запросы этого типа предназначены для извлечения записей и полей, удовлетворяющих некоторому условию, из одной или нескольких таблиц и просмотра результата в режиме таблицы

Запрос параметрический	- запрос для получения сведений по данному полю с разными значениями
Запрос перекрестный	- сведения необходимые пользователю, которые лежат на пересечении столбцов и строчек таблицы
Информационно-логическая модель	- отображает данные предметной области в виде совокупности информационных объектов и связи между ними
Информационный объект	- информационное описание реального процесса, объекта, явления или события. Каждому информационному объекту присваивается уникальное имя. Примеры информационных объектов - СОТРУДНИКИ, КЛИЕНТЫ
Ключ	- одно или несколько полей, однозначно определяющих конкретную запись в таблице
Ключевой реквизит	- это реквизит, однозначно определяющий отдельный конкретный экземпляр объекта. Так, например, для информационного объекта СОТРУДНИКИ ключевой реквизит - это код сотрудника, который однозначно определяет конкретного сотрудника
Кнопка	- элемент управления
Кнопочные формы	- специальные формы, выполняющие функции меню

Макрос	- небольшие программы, в которых содержатся макрокоманды Access, предназначенные для выполнения одного или нескольких действий (например: открыть форму, распечатать отчет, запустить запрос или импортировать данные и т.д.)
Модуль	- программа на языке VB, которая разрабатывается пользователем для реализации нестандартных процедур при создании приложения
Отношение «один-ко-многим»	- обозначает отношение между двумя таблицами, при котором для каждого элемента с одной стороны отношения находится несколько связанных с ним элементов по другую сторону отношения
Отношение «один-к-одному»	- обозначает отношение между объектами, при котором каждому элементу первого объекта соответствует только один элемент второго объекта и каждому элементу второго объекта соответствует не более одного элемента второго объекта
Отношение «многие-ко-многим»	- обозначает отношение между объектами, при котором каждому элем первого объекта соответствует несколько элементов второго объекта и оборот каждому элементу второго объекта соответствует несколько элементов первого объекта
Отчет	- выходной документ для вывода на печать

Поле	- наименьший элемент информации. Каждый тип сведений хранится в отдельном поле. Примеры полей - фамилия, имя. В таблице каждый столбец это поле
Поле со списком	- элемент управления, сочетающий в себе функции двух элементов управления: поля и списка. Позволяет вводить данные путём выбора значения раскрывающегося списка
Предметная область	- часть реального мира, подлежащая автоматизации
Проект	- совокупность объектов Access
Реквизиты	- это количественные и качественные характеристики реального объекта. Например, реквизиты информационного объекта Сотрудники - код сотрудника, фамилия, имя, отчество, дата рождения
Реляционная БД	- «relation» (англ.- отношение) множество связанных между собой двумерных таблиц. В каждой таблице содержатся сведения об одном разделе информации
Форма	- вывод данных из таблиц или запросов в форме удобной для пользователя



4. РЕКОМЕНДУЕМАЯ ЛИТЕРАТУРА

Основная

1. Советов Б. Я. Базы данных: учебник для прикладного бакалавриата, М.: Издательство Юрайт, 2017.
2. Стружкин Н.П. Базы данных: проектирование: учебник для академического бакалавриата, М.: Издательство Юрайт, 2017.
3. Стружкин Н.П. Базы данных: проектирование. Практикум: учебное пособие для академического бакалавриата, М.: Издательство Юрайт, 2017.

Дополнительная

1. Илюшечкин, В. М. Основы использования и проектирования баз данных: учебник для академического бакалавриата , М.: Издательство Юрайт, 2017
2. Титовская Н.В. Титовский С.Н. Проектирование баз данных в СУБД Microsoft Office Access 2007, - Красноярск: КрасГАУ, 2018.
3. Миндалев И.В. Моделирование данных с помощью Data Modeler за 7 дней: методические указания к лабораторным работам. - Красноярск : КрасГАУ, 2017

ПРИЛОЖЕНИЕ 1.

Задания для доработки базы данных «Прокат фильмов».

Задание 1

Доработайте форму «**БЛАНК ВЫДАЧИ ФИЛЬМОВ**»:

1. Добавьте вычисляемое поле «Дата Возврата» (Дата возврата = Дата возврата + Количество дней)
2. Сделайте поле «**Просрочено**» автоматически вычисляемым.
3. Создайте запрос для вывода всех должников на текущую дату.

Задание 2

Измените структуру таблицы «**ВЫДАЧА ФИЛЬМОВ**», используя логическое поле, связанное с возвратом фильма.

Задание 3

Измените инфологическую модель задачи так, чтобы учитывалось количество копий каждого фильма в прокате и наличие данного фильма на текущий момент.

ПРИЛОЖЕНИЕ 2.

Разноуровневые задания для самостоятельной проектной работы

Уровень «А»

Задача № 1

Разработать базу данных «**Абитуриенты**» для приемной комиссии института:

1. Разработать карточку учета Абитуриентов института, в которой должны храниться следующие сведения:
 - фамилия;
 - имя;
 - отчество;

- дата рождения;
- номер аттестата;
- год окончания школы;
- факультет;
- контактный телефон;
- домашний адрес.

Абитуриенты поступают на факультеты: экономический, математический, физический, химический, гуманитарный.

2. Создать запрос на выборку всех абитуриентов по факультетам.

3. Создать запрос на выборку абитуриентов из общего списка, фамилия которых начинается на букву «В» и отсортировать их по возрастанию.

4. Создать отчет по всему списочному составу абитуриентов.

5. Создать отчет о распределении абитуриентов по факультетам.

Задача № 2

Разработать базу данных «Сотрудники» для отдела кадров:

1. Разработать карточку учета сотрудников предприятия, в которой должны храниться следующие сведения:

- фамилия;
- имя;
- отчество;
- дата рождения;
- отдел;
- контактный телефон;
- место жительства.

На предприятии имеются следующие отделы: бухгалтерия, производственный, научно-исследовательский, маркетинговый, транспортный.

2. Создать запрос на выборку сотрудников по всем отделам.

3. Создать запрос на выборку сотрудников из общего списка, фамилия которых начинается на букву «М» и отсортировать их по возрастанию.

4. Создать отчет по всему списочному составу сотрудников предприятия.

5. Создать отчет о распределении сотрудников по отделам.

Задача № 3

Разработать базу данных «**Фильмы**» для пункта проката фильмов:

1. Разработать карточку учета фильмов, в которой должны храниться следующие сведения:

- фамилия заказчика;
- название фильма;
- страна изготовитель;
- продолжительность;
- дата получения;
- дата возврата;
- жанр.

В пункте проката имеются фильмы следующего жанра: комедийные, детективы, военные, лирические, триллеры, трагедии, мелодрамы.

2. Создать запрос на выборку фильмов по каждому из жанров.

3. Создать запрос на выборку фильмов из общего списка, название которых начинается на какую-либо букву и отсортировать их по возрастанию.

4. Создать отчет по всем фильмам, разбив их по жанрам.

5. Создать отчет о фамилиях заказчиков, бравших один и тот же фильм.

Уровень «В»

Задача № 4

Спроектируйте и создайте фрагмент базы данных «**Командировки**» для отдела кадров. Представьте информационно-логическую модель предметной области. Создайте две взаимосвязанные таблицы: «**СОТРУДНИКИ**» (фамилия, имя, отчество, отдел, адрес, телефон) и «**КОМАНДИРОВКИ**» (код

командировки, код сотрудника, название города, дата выезда, количество дней, суточные, стоимость дороги, общая стоимость).

1. Создайте формы для ввода данных, в форме **«КОМАНДИРОВКИ»** создайте поле со списком. Общая стоимость рассчитывается по формуле: *количество дней * суточные + 2 * стоимость дороги*. Для расчета общей стоимости создайте макрос.

2. Введите 3 записи в таблицу **«СОТРУДНИКИ»** и 5 записей в таблицы **«КОМАНДИРОВКИ»**.

3. Создайте запрос на выборку для вывода списка сотрудников, отправленных в командировку в г. Киев.

4. Создайте перекрестный запрос для анализа общей стоимости, полученных денег от всех сотрудников, отправленных в командировку в разные города.

5. Создайте кнопочную форму, содержащую две кнопки:

- запуск макроса;
- перекрестный запрос.

Задача № 5

Спроектируйте и создайте фрагмент базы данных **«Прокат фильмов»** для видеопроката, осуществляющего выдачу видеофильмов в прокат. Представьте информационно-логическую модель предметной области. Создайте две взаимосвязанные таблицы: **«ФИЛЬМЫ»** (название, жанр, страна производитель) и **«ОБРАЩЕНИЯ»** (название фильма, дата выдачи, дата возврата, суточные, общая стоимость).

1. Создайте формы для ввода данных, в форме **«ОБРАЩЕНИЯ»** создайте поле со списком.

2. Общая стоимость рассчитывается по формуле: *дата возврата - дата выдачи * суточные*. Для расчета стоимости создайте макрос.

3. Введите 3 записи в таблицу **«ФИЛЬМЫ»** и 5 записей в таблицу **«ОБРАЩЕНИЯ»**.

4. Создайте запрос на выборку для вывода списка фильмов, выданных определенного числа.

5. Создайте перекрестный запрос для анализа общей суммы денег, полученных видеопрокатом по всем обращениям и всем фильмам.

6. Создайте кнопочную форму, содержащую две кнопки:

- запуск макроса;
- перекрестный запрос.

Задача № 6

Спроектируйте и создайте фрагмент базы данных «Отпускники» для бухгалтерии. Представьте информационно-логическую модель предметной области. Создайте две взаимосвязанные таблицы: «СОТРУДНИКИ» (фамилия, имя, отчество, отдел, телефон) и «ОТПУСКА» (дата начала, сотрудник, количество дней, дата окончания, выплачено отпускных).

1. Создайте формы для ввода данных, в форме «ОТПУСКА» создайте поле со списком.
2. Дата окончания отпуска рассчитывается по формуле: *дата начала + количество дней*. Для расчета даты окончания создайте макрос.
3. Введите 3 записи в таблицу «СОТРУДНИКИ» и 5 записей в таблицу «ОТПУСКА».
4. Создайте запрос на выборку для вывода списка отпусков определенного сотрудника.
5. Создайте итоговый запрос для анализа общей суммы отпускных денег, полученных всеми сотрудниками.
6. Создайте кнопочную форму, содержащую две кнопки:
 - запуск макроса;
 - итоговый запрос.

Задача № 7

Спроектируйте и создайте фрагмент базы данных «Автосервис». Представьте информационно-логическую модель предметной области. Создайте три взаимосвязанные таблицы: «КЛИЕНТЫ» (фамилия, марка автомобиля, год выпуска,

регистрационный номер), **«ОБРАЩЕНИЯ»** (дата, клиент, причина обращения, заключение, время работы, стоимость) и **«ПРИЧИНЫ ОБРАЩЕНИЯ»** (причины обращения, цена часа работы).

1. Создайте формы для ввода данных, в форме **«ОБРАЩЕНИЯ»** создайте 2 поля со списком и дополнительные поля.

2. Стоимость рассчитывается по формуле: цена часа работы*время работы. Для расчета стоимости создайте макрос. Введите 3 записи в таблицу **«КЛИЕНТЫ»** и **«ПРИЧИНЫ ОБРАЩЕНИЯ»**, 5 записей в таблицу **«ОБРАЩЕНИЯ»**.

3. Создайте запрос на выборку для вывода списка обращений, определенного клиента.

4. Создайте перекрестный запрос для анализа общей стоимости услуг, оказанных автосервисом всем клиентам по датам.

5. Создайте кнопочную форму, содержащую три кнопки:

- **«НАЙДИ КЛИЕНТА»** (по нажатию на кнопку должно появляться окно с текстом **«Введите фамилию клиента»**), после ввода фамилии должна быть выдана информация о клиенте из таблицы **«КЛИЕНТЫ»**;
- перекрестный и итоговый запросы.

Задача № 8

Спроектируйте и создайте фрагмент базы данных **«Почта»**, осуществляющей подписку клиентов на периодические издания. Представьте информационно-логическую модель предметной области. Создайте три взаимосвязанные таблицы: **«КЛИЕНТЫ»** (фамилия, адрес, телефон), **«ОФОРМЛЕНИЕ»** (дата оформления, клиент, издание, с какого числа оформляется подписка, срок подписки в месяцах, общая стоимость) и **«ИЗДАНИЯ»** (наименование, количество страниц, цена за месяц).

1. Создайте формы для ввода данных, в форме **«ОФОРМЛЕНИЕ»** создайте 2 поля со списком и дополнительные поля.

2. Стоимость рассчитывается по формуле: цена за месяц* срок подписки в месяцах. Для расчета стоимости создайте макрос.

3. Введите 3 записи в таблицу «КЛИЕНТЫ» и «ИЗДАНИЯ», 5 записей в таблицу «ОФОРМЛЕНИЕ».

4. Создайте запрос на выборку для вывода списка изданий, выписанных определенным клиентом.

5. Создайте перекрестный запрос для анализа общей стоимости услуг, оказанных почтой всем клиентам по датам.

6. Создайте кнопочную форму, содержащую три кнопки:

- «НАЙДИ КЛИЕНТА» (по нажатию на кнопку должно появляться окно с текстом «Введите фамилию клиента», после ввода фамилии должна быть выдана информация о клиенте из таблицы «КЛИЕНТЫ»;
- перекрестный и итоговый запросы.

СПИСОК ВОПРОСОВ ПО БАЗАМ ДАННЫХ

1. Понятие базы данных. Типы. Области применения.
2. Иерархические и сетевые базы данных. Примеры.
3. Разработка реляционных баз данных.
4. СУБД. Назначение. Типы. Области применения.
5. Понятия: запись, поле, ключ в базе данных.
6. Автоматизированный банк данных. Назначение. Структура.
7. Проектирование базы данных.
8. Технология редактирования структуры таблицы.
9. Технология создания таблиц.
10. Главная и подчиненная таблица.
11. Технология создания и редактирования схемы данных.
12. Типы связей между информационными объектами.
13. Связывание таблиц.
14. Типы экранных форм.
15. Технология создания экранных форм.
16. Технология редактирования экранных форм.
17. Технология создания многотабличных экранных форм.
18. Ввод и редактирование данных.

19. Типы объектов.
20. Обеспечение целостности данных в СУБД.
21. Технология создания полей со списком в экранных формах.
22. Технология создания макросов.
23. Технология привязывания макросов к полям основной формы.
24. Вставка подчиненных форм и отчетов в основную форму.
25. Технология создания кнопочных форм.
26. Типы запросов.
27. Типы фильтров.
28. Технология создания фильтров.
29. Технология создания запросов с параметром.
30. Технология создания перекрестных запросов.
31. Технология создания запросов на обновление и удаление.
32. Технология создания итоговых запросов.
33. Технология создания запросов на выборку.
34. Технология создания запросов на добавление.
35. Сортировка данных.
36. Быстрая сортировка данных.
37. Вложенная сортировка данных с помощью запроса.
38. Типы отчетов.
39. Технология создания отчетов.
40. Однотабличный отчет.
41. Технология создания диаграмм.
42. Способы поиска данных.