

МИНИСТЕРСТВО СЕЛЬСКОГО ХОЗЯЙСТВА
РОССИЙСКОЙ ФЕДЕРАЦИИ

ФГБОУ ВПО «Кубанский государственный
аграрный университет»

К. С. Галиев, Е. К. Печурина

БАЗЫ ДАННЫХ И СУБД

Учебно-методическое пособие

Краснодар
КубГАУ
2016

УДК 004.658 (078)

ББК 32.97

Г15

Рецензенты:

Г. А. Аршинов – доктор технических наук, профессор
(Кубанский государственный аграрный университет);

Е. В. Луценко – доктор экономических наук, профессор
(Кубанский государственный аграрный университет)

Галиев К. С.

Г15 Базы данных и СУБД : учеб.-метод. пособие /
К. С. Галиев, Е. К. Печурина. – Краснодар : КубГАУ, 2016. –
76 с.

Учебно-методическое пособие посвящено одному из разделов дисциплины «Информатика»: базам данных и СУБД. Рассматриваются понятия информационной системы, базы и модели данных, СУБД, а также вопросы проектирования базы данных с использованием СУБД Microsoft Access (версия 2010).

Предназначено для студентов бакалавров начальных курсов, изучающих дисциплину «Информатика».

УДК 004.658 (078)

ББК 32.97

© Галиев К. С.,
Печурина Е. К., 2016
© ФГБОУ ВПО «Кубанский
государственный аграрный
университет», 2016

ОГЛАВЛЕНИЕ

ВВЕДЕНИЕ.....	5
ЧАСТЬ 1. ЛЕКЦИИ ПО БАЗАМ ДАННЫХ И СУБД	6
1 ОСНОВНЫЕ СВЕДЕНИЯ О БАЗАХ ДАННЫХ И СУБД.....	6
1.1 Информационная система	6
1.2 База данных	9
1.3 Системы управления базами данных (СУБД)	10
1.4 Классификация	10
1.5 Модели данных	12
1.6 Нормализация	15
1.7 Типы связей	17
1.8 Индексы	18
1.9 Целостность базы данных	18
Контрольные вопросы.....	19
2 ПРОЕКТИРОВАНИЕ БАЗЫ ДАННЫХ.	
СУБД MS ACCESS.....	20
2.1 Краткая технология разработки базы данных.....	20
2.2 Основные понятия в СУБД MS Access	22
2.3 Основные приемы работы в СУБД MS Access	24
2.3.1 Таблицы	24
2.3.2 Схема данных	25
2.3.3 Заполнение таблиц.....	25
2.3.4 Формы	25
2.3.5 Отчеты.....	26
Контрольные вопросы.....	26
ЧАСТЬ 2. ЛАБОРАТОРНЫЕ РАБОТЫ.	
СУБД MS ACCESS 2010	28
1 КОНСТРУИРОВАНИЕ ТАБЛИЦ И ИХ ЗАПОЛНЕНИЕ	28
1.1 Создание новой базы данных	29
1.2 Создание таблицы Факультеты.....	29
1.3 Создание таблицы Студенты.....	30
1.4 Создание схемы данных	31
1.5 Заполнение таблицы Факультеты.....	32
1.6 Заполнение таблицы Студенты.....	32
1.7 Редактирование таблицы Студенты	33

2 СОЗДАНИЕ ПРОСТЫХ ФОРМ, ЗАПРОСОВ И ОТЧЕТОВ	34
2.1 Конструирование формы Список студентов	34
2.2 Конструирование формы Список факультетов	36
2.3 Работа с формой Список студентов	37
2.4 Создание простого запроса по выбору пола	37
2.5 Создание простого запроса по выбору группы	38
2.6 Создание простого отчета на основе запроса	40
2.7 Создание отчета с итогами	42
2.8 Редактирование отчета с итогами	43
3 СОЗДАНИЕ СЛОЖНЫХ ФОРМ, ЗАПРОСОВ И ОТЧЕТОВ	45
3.1 Создание формы Средний балл	46
3.2 Создание запроса со сложным условием отбора	47
3.3 Создание отчета на основе сложного запроса	49
3.4 Создание формы с кнопкой для просмотра отчета	51
3.5 Создание формы, имеющей поле со списком	54
3.6 Создание запроса на основе поля со списком, включенного в форму	55
3.7 Создание отчета со списком студентов	57
3.8 Просмотр отчета через кнопку формы	58
4 СОЗДАНИЕ ГЛАВНОЙ ФОРМЫ БД. СОЗДАНИЕ НОВОЙ ТАБЛИЦЫ И ВЫВОД ФОТОГРАФИЙ	62
4.1 Создание главной формы с кнопками	62
4.2 Вставка рисунка в Главную форму	66
4.3 Создание новой таблицы	67
4.4 Создание формы для вывода фотографий	69
СПИСОК ЛИТЕРАТУРЫ	75

ВВЕДЕНИЕ

В настоящее время компьютеры используются практически во всех сферах деятельности человека. В пособии речь идёт об основных понятиях в базах данных и системах управления базами данными (СУБД), составляющих ядро автоматизированных информационных систем, использующих компьютеры.

Даются понятия об информационных системах, информационной технологии, базах данных, СУБД и моделях данных, а также кратко описываются этапы проектирования баз данных. Подробно рассмотрен пример выполнения лабораторных работ по разработке учебной базы данных с использованием СУБД MS Access. Пособие состоит из двух частей.

В первой части пособия представлены содержания двух лекций. Рассматриваются основные понятия в базах данных и СУБД MS Access. В конце лекций приводятся контрольные вопросы для проверки степени усвоения учебного материала.

Во второй части пособия подробно рассматривается создание учебной базы данных с использованием СУБД MS Access (версия 2010). Вторая часть пособия оформлена в виде четырех лабораторных работ. Выполнению лабораторных работ предшествует алгоритм действий для достижения требуемого результата. К лабораторным работам прилагается готовая база данных, создание которой описывается в тексте работ.

Нумерация рисунков и таблиц в первой и второй частях начинается заново, потому что в лабораторных работах нет прямых ссылок на лекционный материал. Кроме того, нумерация рисунков в лабораторных работах соответствует номеру темы.

Данное учебно-методическое пособие предназначено для студентов бакалавров начальных курсов, изучающих дисциплину «Информатику».

ЧАСТЬ 1. ЛЕКЦИИ ПО БАЗАМ ДАННЫХ И СУБД

1 ОСНОВНЫЕ СВЕДЕНИЯ О БАЗАХ ДАННЫХ И СУБД

1.1 Информационная система

Любые рассуждения в дисциплине «Информатика» всегда предполагают наличие компьютера. Компьютер – это техническое средство для обработки информации. В связи с понятием «обработка информации» существуют еще два понятия: информационная технология и информационная система.

Информационная технология – это взаимосвязанная совокупность **двух** частей: 1) правила, методы, способы, алгоритмы и т. п. для обработки информации; и 2) технические средства для обработки информации.

Информационная система – это взаимосвязанная совокупность **трех** частей: 1) правила, методы, способы, алгоритмы и т. п. для обработки информации; 2) технические средства для обработки информации; и 3) грамотный персонал по обработке информации.

Другими словами, информационная система = информационная технология + грамотный персонал. Зачастую в определении информационной системы умалчивают наличие третьей составляющей, сводя, таким образом, оба понятия в одно. Однако, известно, что, во-первых, любая совершенная и опробованная технология не дает требуемого результата при неграмотном персонале; во-вторых, целью любой системы является получение нового свойства, которое отсутствует у составляющих.

Цель любой информационной системы – обработка информации конкретной предметной области **с выдачей результата**.

Приведем пример, используя простейшие рассуждения. Если, что-то *выдает* требуемый результат, т. е. работает, тогда это система. Если, что-то может выдать требуемый результат, но

не выдает, потому что чего-то не хватает (грамотного персонала), тогда это технология.

Вообще, **все**, что нас окружает в этом мире, это различные системы. *Система* – это нечто целое, составленное из взаимосвязанных частей; система обладает *новым свойством*, которое отсутствует у составляющих. Именно ради приобретения нового свойства создаются системы. Сама система является подсистемой более сложного образования; любая составляющая системы, в свою очередь, является тоже системой. Например, велосипед – это система, составленная из рамы, колес, руля, педали и т. п., которые специальным способом взаимосвязаны. Новое свойство велосипеда, как системы – это способность передвигать человека. Если составляющие взаимосвязаны по-другому, например, сваркой, тогда новое свойство (передвигать человека) не возникает. Далее, колесо, в свою очередь, тоже является системой – состоит из обода, покрышки, камеры, спиц, втулки и т. п.

Заметим также, что в «Информатике» встречаются много словосочетаний, использующих понятие *системы* – операционная система, система счисления, двоичная система, видеосистема, звуковая система, система программирования, информационная система и т. д.

Отметим еще одно обстоятельство, имеющее отношение не только к «Информатике», а к образовательной *системе* вообще. Целью обучения любого студента является приобретение знаний, чтобы стать грамотным специалистом в своей предметной области. Другими словами, обучаются для того, чтобы стать полезной составляющей, в виде грамотного специалиста, для какой-то реально работающей системы в своей предметной области.

Понятия *информационная технология* и *информационная система* явно обозначились в связи с появлением нового технического средства (компьютера) для обработки информации. До появления компьютера информация обрабатывалась **при** непосредственном участии человека. Компьютер позволяет обрабатывать информацию **без** непосредственного участия человека, с помощью заранее составленного алгоритма (компьютерной программы). Иначе говоря, компьютер позволяет *автоматизиро-*

вать обработку информации. Автоматизация – это замена человека техническим средством для выполнения работы.

Приведем распространенные определения понятия информационной системы:

Информационная система – организационно упорядоченная совокупность документов (массивов документов) и информационных технологий, в том числе с использованием средств вычислительной техники и связи, реализующих информационные процессы. (Закон РФ «Об информации, информатизации и защите информации» от 20.02.1995 № 24-ФЗ – устарел).

Информационная система – совокупность содержащейся в базах данных информации и обеспечивающих ее обработку информационных технологий и технических средств. (Закон РФ «Об информации, информационных технологиях и защите информации» от 27.07.2006 № 149-ФЗ).

Информационные технологии – процессы, методы поиска, сбора, хранения, обработки, предоставления, распространения информации и способы осуществления таких процессов и методов. (Закон РФ «Об информации, информационных технологиях и защите информации» от 27.07.2006 № 149-ФЗ).

Информационной системой называется комплекс, включающий вычислительное и коммуникационное оборудование, программное обеспечение, лингвистические средства и информационные ресурсы, а также системный персонал и обеспечивающий поддержку динамической информационной модели некоторой части реального мира для удовлетворения информационных потребностей пользователей [1].

Информационная система – система обработки информации, работающая совместно с организационными ресурсами, такими как люди, технические средства и финансовые ресурсы, которые обеспечивают и распределяют информацию [2]. Это взято из стандарта ISO – *International Standardization Organization* (международная организация по стандартам).

Информационная система в узком смысле – это программно-аппаратная система, предназначенная для автоматизации целенаправленной деятельности конечных пользователей, обеспечивающая, в соответствии с заложенной в нее логикой обработ-

ки, возможность получения, модификации и хранения информации [3].

Другими словами, **информационная система** – это аппаратные и программные средства, предназначенные для того, чтобы своевременно обеспечить пользователей нужной информацией.

Любая информационная система выполняет две основные задачи:

- 1) хранение данных;
- 2) доступ к данным, т. е. возможность искать, делать запросы и изменять данные.

Массивы данных в информационных системах имеют большой объем и хранятся во внешних запоминающих устройствах компьютера. Данные должны храниться в таком виде, чтобы их можно было легко искать и делать запросы. Такие массивы данных называются *базами данных* (БД).

1.2 База данных

База данных (БД, database) – это информационные структуры, содержащие взаимосвязанные данные о реальных объектах.

База данных – это структурированный набор данных, удобный для выполнения запроса и поиска необходимых данных.

Пример из реальной ситуации. В одном из сельхозпредприятий (СХП) надо было составить список работников, которым можно выделить земельный пай. Секретарша СХП, которая недавно сменила пишущую машинку на компьютер, создала в Word текст со всеми необходимыми сведениями. В районном управлении этот текст вернули на доработку с замечанием, что из этого документа нельзя извлечь требуемые данные. Вывод – данные не были структурированы.

Разделив сведения на столбцы, используя Excel, получили структурированный список. Но это все равно не база данных.

Особенностями совокупности данных в БД являются:

- достаточно большие объемы информации;
- максимально возможная компактность хранения данных;
- возможность извлечения из базы данных разнообразной информации в определенной предметной области;

- удобные для пользователя вид и форма извлекаемой информации;
- высокая скорость доступа к данным;
- надежность хранения информации и возможность предоставления санкционированного доступа к данным для отдельных пользователей;
- удобство и простота конструирования пользователем запросов, форм и отчетов для выборки данных.

Создание базы данных, ее поддержка и обеспечение доступа пользователей к ней осуществляется с помощью специального программного инструмента – системы управления базой данных. (СУБД).

1.3 Системы управления базами данных (СУБД)

Система управления базами данных (СУБД) – это программное обеспечение для создания и редактирования баз данных, просмотра и поиска информации в них.

Хотя термины СУБД и «база данных» обозначают различные понятия, они неразрывно связаны: свойства базы данных определяются СУБД, которая ею управляет, и наоборот. Комплекс «БД + СУБД» называется информационной системой в узком смысле. Поэтому можно сказать, что база данных является ядром информационной системы.

СУБД решают все задачи, связанные с управлением данными, в том числе:

- поиск данных;
- редактирование данных;
- выполнение расчетов;
- обеспечение целостности (корректности, непротиворечивости) данных;
- восстановление данных после сбоев.

1.4 Классификация

По технологии обработки базы данных делятся на централизованные и распределенные.

В простейшем случае база данных и СУБД находятся на одном компьютере. Такая информационная система называется **локальной**, с ней работает один пользователь. Недостаток такой системы проявляется тогда, когда с БД должны работать несколько пользователей на своих компьютерах:

- базу данных надо обновлять на каждом компьютере;
- невозможно стыковать изменения, вносимые пользователями на своих компьютерах.

В современных информационных системах применяются **удаленные базы данных**, расположенные на серверах (специально выделенных компьютерах) локальной или глобальной сети. В этом случае несколько пользователей могут одновременно работать с базой данных и вносить в неё изменения.

СУБД, работающие с удаленными базами данных, различаются двумя способами разделения функций обработки:

- файл-серверные СУБД;
- клиент-серверные СУБД.

Файл-серверные СУБД расположены на компьютерах пользователей, а сервер хранит только файлы базы данных, но не участвует в обработке данных (рисунок 1.1).

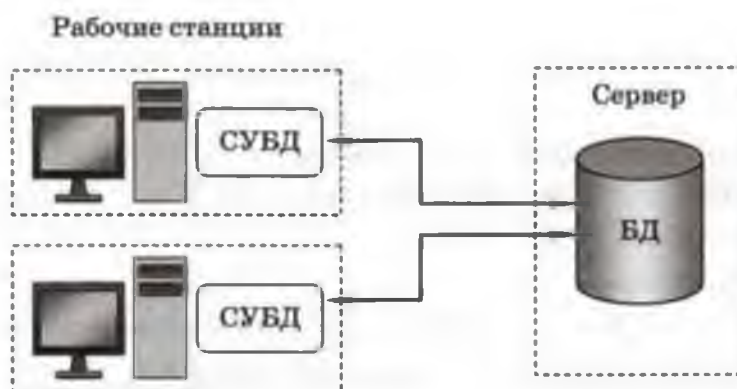


Рисунок 1.1 – Файл-серверные СУБД

Клиент-серверная СУБД расположена на том же компьютере, где находится база данных (рисунок 1.2).

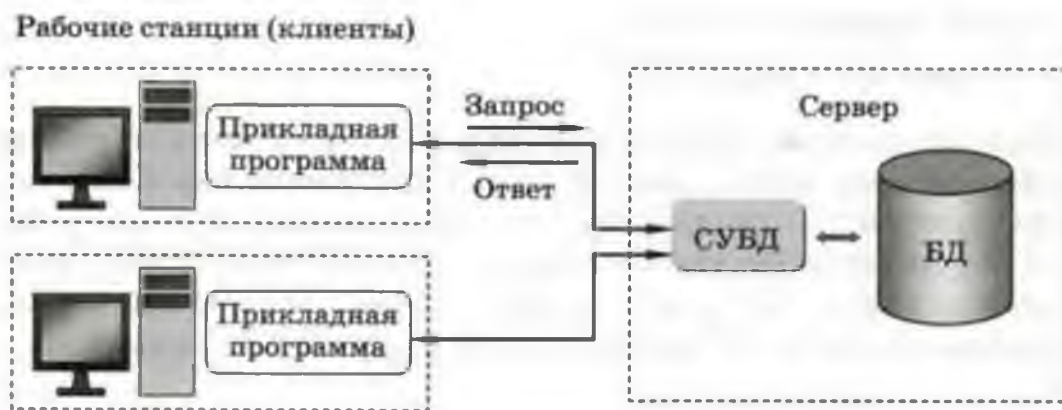


Рисунок 1.2 – Клиент-серверная СУБД

На компьютере пользователя расположена только прикладная программа-клиент, которая по сети обращается с запросом к СУБД для выполнения операций с данными.

В современных клиент-серверных СУБД для управления данными используют язык SQL (*Structured Query Language* – язык структурированных запросов). Он содержит все команды, необходимые для работы с данными.

Самые известные клиент-серверные СУБД следующие: *Microsoft SQL Server, Oracle, Firebird, PostgreSQL, MySQL*.

1.5 Модели данных

Модель – это замена реального объекта (оригинала). Модель снабжают только существенными свойствами оригинала, которые важны для данного исследования.

Моделирование – это создание и исследование модели с целью изучения оригинала.

Практически все, что делается с помощью компьютера, – это моделирование. Например, база данных продуктового магазина – это модель реального магазина.

Основой базы данных является **модель данных**. Существуют два уровня моделей данных:

- инфологическая модель данных;
- даталогическая модель данных.

Инфологическая (информационно-логическая) модель данных отражает предметную область в виде совокупности информационных объектов и их структурных связей.

Информационный объект – это описание некоторой сущности (явления, реального объекта, процесса) в виде совокупности логически связанных данных (реквизитов).

Например, информационный объект **Студент** описывает некоторую сущность – *студент*. Реквизитный состав этого информационного объекта, т. е. его структура, следующая: *№ группы, ФИО, № зачетной книжки, дата рождения*.

Информационный объект имеет множество реализаций – экземпляров.

Инфологические модели данных для описания базы данных могут быть иерархическими, сетевыми и реляционными.

Иерархическая (древовидная) модель данных представляет собой иерархию элементов, называемых узлами. Узел – это совокупность атрибутов данных, описывающих информационный объект. На самом верхнем уровне имеется только один узел – корень. Каждый узел кроме корня связан только с одним узлом на более высоком уровне, называемом исходным узлом для данного узла. Каждый узел может быть связан с одним или несколькими узлами более низкого уровня, называемыми порожденными (подчиненными). Узлы, не имеющие порожденных, называются листьями. К каждой записи базы данных существует только один путь.

Например, модель управления работой университета. Во главе университета находится ректор; ректору подчиняются несколько проректоров; проректор управляет работой факультетов или других подразделений (отдел кадров, бухгалтерия, библиотека и т. п.); декан управляет работой кафедр, студенческих групп и т. д.

Сетевая модель также основывается на понятиях узел, уровень, связь. Сетевая модель данных – это модель, в которой порожденный узел может иметь более одного исходного узла. В сетевой структуре любой элемент любого уровня может быть связан с любым другим элементом.

Например, модель управления работой автобусных сообщений города или региона (края).

Реляционная модель базы данных состоит из одного или нескольких взаимосвязанных сущностей (таблиц) с атрибутами

(столбцами), каждая из которых соответствует одной узкой части предметной области.

Например, инфологическая модель обучения студентов в университете может быть описана набором связанных таблиц (рисунок 1.3).

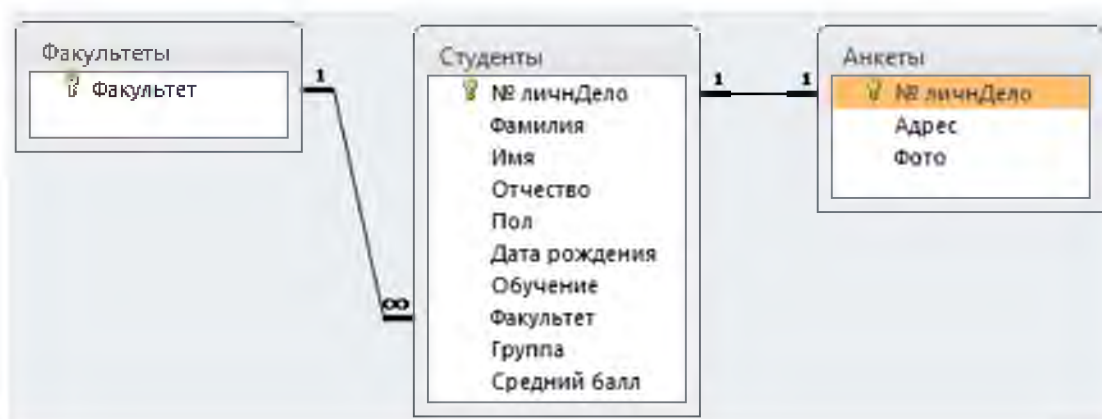


Рисунок 1.3 – Инфологическая модель

Отметим ещё раз (в виде повтора): **Инфологическая** модель данных отражает предметную область в виде совокупности информационных объектов и их структурных связей. Здесь не идет речь о программной реализации базы данных.

В дальнейшем речь будет идти только о *реляционной* модели данных. В *реляционной* модели данных приняты следующие термины:

- *сущность* – это название таблицы;
- *атрибут* или *поле* – это название столбца таблицы;
- *запись* или *кортеж* – это строка данных в таблице;
- *ключ первичный* – поле (одно или более), значение которого однозначно определяет запись в таблице; каждая таблица **обязательно** должна иметь первичный ключ;
- *ключ внешний* – поле, через которое осуществляется связь с другой таблицей; внешний ключ связан с первичным ключом другой таблицы; таблица может не иметь внешних ключей.

Даталогическая (англ. *data* – данные, т. е. логика данных) модель данных отражает информационные объекты базы данных в терминах выбранного средства программирования.

Выбрав средство программирования необходимо переписать таблицы, используя термины данного средства программирования, т.е. таблицы и их атрибуты описать как идентификаторы, указать типы данных, их размер, указать связи между таблицами.

Например, даталогическая модель данных сущности **Анкеты** в терминах MS Access может быть представлена в следующем виде (таблица 1.1).

Таблица 1.1 – Даталогическая модель данных

Сущность Анкеты			
Атрибуты	Тип данных	Размер	Связь
№ личнДело	текстовый	5	Ключ первичный. Связь с сущностью Студенты .
Адрес	текстовый	25	
Фото	текстовый	20	

1.6 Нормализация

Допустим, что данные о студентах собраны в одной таблице (таблица 1.2)

Таблица 1.2 – Сведения о студентах

ФИО	Обучение	Факультет	Адрес
Кучерин В. Л.	очное	энергетики	Общежитие 3
Курочкин Л. Д.	очное	энергетики	Общежитие 3
Копыл А. А.	заочное	энергетики	г. Кропоткин
Иванова М. Ф.	очное	ветеринарный	Общежитие 5

Мы видим, что в таблице имеется избыточность (дублирование): многие данные повторяются (хранятся) несколько раз. Кроме того, при вводе данных можно сделать опечатку, например, вместо «энергетики» ввести «энергитеки». В этом случае нару-

шается *целостность* базы данных, потому что в ней хранится название несуществующего факультета.

Чтобы избежать таких проблем, выполняют *нормализацию* базу данных.

Нормализация – это изменение структуры базы данных, которое устраняет избыточность и предотвращает возможные нарушения целостности. Нормализация достигается разбиением таблицы на ряд составляющих.

Существуют несколько уровней нормализации (называются нормальными формами). Рассмотрим некоторые принципы нормализации на примерах.

1) **Любое поле должно быть неделимым.** Это значит, что таблицу 1.2 необходимо переделать: поле *ФИО* разделить на три поля – *Фамилия*, *Имя*, *Отчество* (таблица 1.3).

Таблица 1.3 – Неделимые поля

Фамилия	Имя	Отчество
Кучерин	Владимир	Львович
Курочкин	Леонид	Дмитриевич
Копыл	Анна	Александровна
Иванова	Мария	Фроловна

2) **Повторяющиеся значения в поле выделить в отдельную таблицу.** Это значит, что таблицу 1.2 надо переделать (таблица 1.4).

Таблица 1.4 – Выделение отдельных таблиц

Фамилия	КодО	КодФ	КодО	Обучение	КодФ	Факультет
Кучерин	1	1	1	очное	1	энергетики
Курочкин	1	1	2	заочное	2	ветеринарный
Копыл	2	1			3	агрономический
Иванова	1	2				

3) **Таблица должна иметь поле с первичным ключом.** Это значит, что любая запись в таблице – уникальна, т. е. нет одинаковых строк. Если таблица в принципе может иметь одинаковые записи, например, одинаковые ФИО, то следует в таблицу доба-

вить поле, выполняющую роль первичного ключа. В данном случае добавим поле с номером личного дела студента (таблица 1.5).

Таблица 1.5 – Сведения о студентах

† № личнДело	ФИО	...
2135	Кучерин В. Л.	
2124	Курочкин Л. Д.	
0423	Копыл А. А.	
0018	Иванова М. Ф.	

Здесь значок † означает, что это поле является ключевым.

1.7 Типы связей

В базе данных используются два вида связей «один ко многим» и «один к одному».

Связь «один ко многим» обозначается 1:N или 1-∞. Это значит, что с одной записью в первой таблице могут быть связаны несколько записей во второй таблице. При этом на стороне «1» в связи участвует первичный ключ, а на стороне «∞» – внешний ключ.

Связь «один к одному» обозначается 1:1, каждой записи в первой таблице соответствует одна запись в связанной таблице. В этом случае таблицы связаны через свои первичные ключи. Такая связь используется для разделения большой таблицы на две части.

Иногда при анализе данных (на этапе проектирования базы данных) появляются связи типа «многие ко многим». Например, преподаватель может преподавать несколько разных дисциплин, а некоторые дисциплины обычно преподают несколько преподавателей. Поэтому связь между таблицами **Преподаватели** и **Дисциплины** – это связь типа «многие ко многим». Современные СУБД не поддерживают связи типа «многие ко многим», вместо этого используется дополнительная (промежуточная) таблица и к ней приставляются две связи «один ко многим».

1.8 Индексы

Индекс – это вспомогательная (виртуальная) таблица, которая служит для ускорения поиска в основной таблице.

Простейший индекс – это вспомогательная таблица из двух столбцов (полей), в которой хранятся значения интересующего нас поля основной таблицы (например, *Фамилия*) и список номеров записей. Записи в индексе упорядочены (отсортированы) по нужному полю. При изменении основной таблицы индексы создаются заново. Поэтому индексные поля надо создавать только по необходимости.

1.9 Целостность базы данных

Целостность базы данных означает, что БД содержит полную и непротиворечивую информацию и удовлетворяет всем заданным ограничениям.

Прежде всего, нужно обеспечить *физическую целостность* БД, т. е. защитить данные от разрушения в случае отказа оборудования (например, при отключении питания).

СУБД выполняет все изменения данных с помощью *транзакций*, которые позволяют в случае сбоя «откатить назад» все начатые операции.

Транзакция (англ. *transaction* – сделка) – это группа операций, которая представляет собой одно законченное действие. Транзакция должна быть выполнена целиком или не выполнена вообще.

Транзакция выполняется СУБД путем создания копии всех данных, которые будут изменяться, и записывает в специальный журнал все операции, которые нужно выполнить. Затем эти операции выполняются фактически и, если всё завершено успешно, запись удаляется из журнала. Если произошел сбой, в журнале будет найдена последняя информация о тех операциях, которые уже были завершены, и база данных восстанавливается в исходное состояние (транзакция не выполнена).

Кроме транзакции, периодически администраторы создают резервные копии всех данных. При сбое восстанавливается самая последняя сохраненная версия БД.

Контрольные вопросы

1. Что такое информационная система?
2. Что такое информационная технология?
3. Что такое база данных?
4. Что означает СУБД и что это такое?
5. Почему база данных является ядром информационной системы?
6. Какие основные особенности у базы данных?
7. Что такое модель данных?
8. Что значит моделирование?
9. Какие два уровня моделей различают при разработке базы данных?
10. Что значит инфологическая модель данных?
11. Что значит даталогическая модель данных?
12. Что значит иерархическая модель данных?
13. Что значит сетевая модель данных?
14. Что значит реляционная модель данных?
15. Какие основные термины используются в реляционной модели?
16. Что означает понятие сущности и атрибута в реляционной модели?
17. Что обозначает «схема данных»?
18. Что значит связи «один ко многим» и «один к одному», как обозначаются такие связи в схемах данных?
19. Как реализуется связь «многие ко многим»?
20. Что значит нормализация базы данных?
21. Что такое индексы в базах данных?
22. Что такое транзакция?

2 ПРОЕКТИРОВАНИЕ БАЗЫ ДАННЫХ. СУБД MS ACCESS

2.1 Краткая технология разработки базы данных

1) Изучить предметную область выбранного объекта. Например, учет товаров в продуктовом магазине.

2) Изучить состав документооборота и/или потоки информации в выбранной предметной области. Например, что имеется в данный момент: типы товаров – хлебные изделия, бакалея, напитки и т. п.; затем в каждом типе – наименование товаров, цена, вес, тара, упаковка, производитель; как организовано движение товаров – поставщики, заказы поставщикам, реализация товара и т. п. Изучить источники и структуру входной и выходной информации. Определить (сформулировать) основные функции автоматизации задачи (выбранной предметной области). Операции, выполненные в данном пункте, называются составлением или написанием **«Постановки задачи»**.

3) Выписать все основные компоненты, определяющие документооборот и/или потоки информации. Сгруппировать эти компоненты в виде отдельных таблиц. При этом некоторые компоненты одной таблицы будут повторяться в других таблицах, или другими словами, данные для таблицы можно будет выбирать из другой таблицы. Таким образом, можно выделить основные и второстепенные таблицы. Таблицы со своими компонентами (атрибутами) следует расположить рядом таким образом, чтобы повторяющиеся атрибуты таблиц связать линиями, т. е. установить связи между таблицами. Операции, выполненные в данном пункте, называются разработкой информационно-логической модели (**инфологическая модель**). Для графической интерпретации употребляют термин **«схема данных»**. Следует отметить, составление инфологической модели является самой ответственной частью разработки базы данных. Инфологическая модель является основой (фундаментом) базы данных. Процесс составления инфологической модели требует от студента многократного обдумывания и уточнения.

4) После составления инфологической модели (схемы данных) следует выбрать средство для программирования. Этим средством может быть алгоритмический язык Бейсик, Паскаль, Си или др. Наиболее удобным средством для первоначального изучения технологии разработки базы данных считается MS Access. Выбрав средство программирования необходимо переписать таблицы, используя термины данного средства программирования, т. е. таблицы и их атрибуты описать как идентификаторы, указать типы данных, их размер, указать связи между таблицами. Операции, выполненные в данном пункте, называются разработкой **дatalogической модели**. В Access даталогическая модель создается с помощью *конструктора* таблиц. Связи между таблицами устанавливаются при выборе типа данных «*Мастер подстановок*». Вид связи уточняется через инструмент «*Схема данных*».

5) После разработки даталогической модели (после конструирования всех таблиц) следует заполнить таблицы данными. Заполнение таблиц начинают с второстепенных (они на схеме данных расположены с краю). Здесь уместно сделать следующее важное замечание: В хорошей базе данных заполнение основных таблиц выполняется выбором данных из второстепенных таблиц путем сведения к минимуму набора текста через клавиатуру. Кроме этого следует избегать избыточности данных в базе. Как было сказано выше в пункте 3, это зависит от качества инфологической модели. В теории баз данных этим вопросам посвящается раздел «*нормализация отношений*». **Нормализация** – это разбиение таблицы на две или более, обладающих лучшими свойствами при включении, изменении и удалении данных. Окончательная цель нормализации сводится к получению такого проекта базы данных, в котором каждый факт появляется лишь в одном месте, т. е. исключена избыточность информации.

6) После заполнения всех таблиц данными следует формирование (конструирование) запросов, форм и отчетов.

2.2 Основные понятия в СУБД MS Access

Особенностью данной СУБД является то, что вся информация базы данных хранится в одном файле (*.mdb или *.accdb для версии Access 2007, 2010) [4].

Объектами СУБД MS Access являются:

- *таблицы* для хранения данных;
- *запросы* на поиск и извлечение только необходимых данных;
- *формы* для просмотра, добавления и обновления данных в таблицах;
- *отчеты* для анализа или печати данных.

Все основные сведения базы данных находятся в записях *таблиц*.

С помощью *запроса* можно найти и извлечь данные (в том числе и данные из нескольких таблиц), соответствующие указанным условиям. Запросы также используются для обновления или удаления нескольких записей одновременно и выполнения предопределенных или пользовательских вычислений на основе данных.

Формы можно использовать для просмотра, ввода и изменения данных в одной строке. Кроме того, с их помощью можно выполнять такие действия, как отправка данных другим приложениям. Формы обычно содержат элементы управления (поля, кнопки и т. п.), связанные с полями базовых таблиц. При открытии формы Access извлекает данные из одной или нескольких таких таблиц и отображает их в выбранном при создании формы формате.

Отчет можно использовать для быстрого анализа данных или их отображения в определенном виде в печатном или другом формате. Например, можно создать отчет, в котором сгруппированы данные и вычислены итоги.

Приведем основные характеристики указанных объектов (таблица 2.1).

Таблица 2.1 – Характеристики объектов MS Access

Атрибут	Максимальное значение
<i>База данных</i>	
Размер файла базы данных	2 Гбайт
Число объектов в базе данных	32 768
Число знаков в имени объекта	64
<i>Таблица</i>	
Число знаков в имени таблицы	64
Число знаков в имени поля	64
Число полей в таблице	255
Число открытых таблиц	2048
Размер таблицы	2 Гбайт
Число знаков в текстовом поле	255
Число знаков в поле MEMO (длинный текст)	65 535 или 1 Гбайт
Размер поля объекта OLE	1 Гбайт
Число индексов в таблице	32
Число полей в индексе	10
<i>Запрос</i>	
Число установленных связей	32 на одну таблицу
Число таблиц в запросе	32
Число полей в наборе записей	255
Размер набора записей	1 Гбайт
<i>Форма и отчет</i>	
Число знаков в надписи	2 048
Число знаков в поле	65 535
Ширина формы или отчета	22 дюйма (55,87 см)
Высота раздела	22 дюйма (55,87 см)
Число уровней вложения форм или отчетов	7

В таблице 2.2 приведены сведения о типах данных, которые могут иметь поля в таблицах.

Таблица 2.2 – Типы данных в таблице Microsoft Access

Тип данных полей таблицы	Использование	Размер
Текстовый	Текст из символов Unicode (2 байта на символ)	До 255 символов
Поле МЕМО	Текст в кодировке Unicode	До 64000 символов
Логический	Поле, содержащее только Да/Нет, Истина/Ложь, Вкл/Выкл.	1 бит
Числовой	Числовые данные	1, 2, 4, 8 или 16 байтов
Дата/время	Дата и время	8 байтов
Денежный	Значения валют	8 байтов
Счетчик	Автоматическая вставка последовательных чисел	4 байта
Поле объекта OLE	Объекты Word, Excel и др.	До 1 Гбайта
Гиперссылка	Гиперссылка может иметь вид пути, либо URL-адреса	До 64000 символов
Мастер подстановок	Выбор значения из другой таблицы, запроса или из списка фиксированных значений	Размер первичного ключа

2.3 Основные приемы работы в СУБД MS Access

2.3.1 Таблицы

Сначала надо создать таблицу в режиме *Конструктор*. При этом надо указать имена полей, тип данных, размер поля и другие свойства (обязательность значения в поле, создание индекса поля и т. п.). Обязательно указать *ключевое поле*, под которым подразумевается *первичный ключ*, который однозначно идентифицирует запись в таблице. Для ключевого поля контролируется отсутствие дублирования его значений при вводе данных. *Внешний ключ* обозначается как *Индексированное поле*, допускаются совпадения.

2.3.2 Схема данных

Схема данных создается в меню *Работа с базами данных*. Надо в схему добавить все таблицы и установить связи между ними.

2.3.3 Заполнение таблиц

Затем надо заполнить **таблицы** известными значениями. Заполнение выполняется в режиме *Таблица*, при этом некоторые значения берутся из связанных таблиц или из заранее созданного списка фиксированных значений.

Если обратно открыть таблицу в режиме *Конструктора*, то можно подробно изучить ее структуру. Например, можно найти параметры типа данных для каждого поля или узнать, используются ли в таблице поля подстановок – поля, которые с помощью запросов извлекают данные из других таблиц. В окне *Свойства поля* на вкладке *Подстановка* в строке *Источник строк* можно увидеть текст запроса на языке SQL.

2.3.4 Формы

Экранные **формы** позволяют организовать наглядную и удобную работу с базой данных. Если форма составлена из нескольких связанных таблиц, то на одной форме можно организовать работу с главной и подчиненными таблицами, выбор данных из таблиц-справочников с использованием раскрывающихся списков, режимы поиска и отбора информации, печати необходимых отчетов на принтере и пр.

Для разработки имеется **Мастер форм**, который позволяет легко создавать формы нескольких видов (простые для работы с данными одной таблицы, более сложные для работы с несколькими таблицами с использованием подчиненных форм). Полученные формы далее, как правило, приходится дополнять и модифицировать в **Конструкторе форм** для реализации всех необходимых условий работы с данными.

На форме можно устанавливать *элементы управления* в виде надписей, полей, кнопок, рисунков и т. п.

Для формы и всех ее элементов можно открыть *Окно свойств*, выбрав в меню значок *Страница свойств*. Все свойства в окне разбиты на группы:

- *Макет* – расположение, шрифт, цвет и прочее, связанное с внешним видом объекта;
- *Данные* – важнейшее свойство в этом разделе – *Источник записей* – для объектов, связанных с редактированием каких-либо данных (связанные объекты – поля, списки...);
- *События* – макросы, выражения и программы, выполняющиеся для объекта при наступлении определенных событий (*Загрузка, Открытие, До обновления, После обновления* и пр.);
- *Другие* – прочие свойства.

2.3.5 Отчеты

Для разработки отчетов, отражающих информацию базы данных в печатных документах, используется *Мастер отчетов* и *Конструктор отчетов*. Отчеты могут быть созданы на основе всей информации, присутствующей в таблицах базы, но чаще для отчетов необходимо отобрать нужную информацию из базы с использованием запроса и на основе его создать отчет. Важным свойством отчетов является возможность группировки данных и получения итоговых данных для групп и всего отчета.

Контрольные вопросы

1. Что означает постановка задачи при проектировании базы данных?
2. Что такое инфологическая модель данных?
3. Что такое даталогическая модель данных?
4. Какое главное назначение СУБД MS Access?
5. Какое расширение имеет файл базы данных, созданный в Access?
6. Какие основные объекты создаются в Access?
7. Какие обязательные атрибуты должны быть определены для каждого поля таблицы реляционной базы данных?
8. Какие типы могут иметь поля таблицы Access?

9. Что такое запрос в Access, что является источником запросов?
10. Что такое форма в Access, какие элементы (объекты) могут на ней присутствовать?
11. Как открыть окно свойств формы или отчета?
12. Какие вкладки содержит окно свойств формы или отчета?
13. Что такое отчет в Access, какие элементы (объекты) могут на ней присутствовать?

ЧАСТЬ 2. ЛАБОРАТОРНЫЕ РАБОТЫ. СУБД MS ACCESS 2010

1 КОНСТРУИРОВАНИЕ ТАБЛИЦ И ИХ ЗАПОЛНЕНИЕ

Цель: научиться создавать базу данных в СУБД Microsoft Access; определять структуру и взаимосвязь таблиц; вводить, корректировать, сортировать и фильтровать данные в таблицах.

БД «Университет» состоит из двух таблиц: **Факультеты** и **Студенты** (рисунок 1.0):

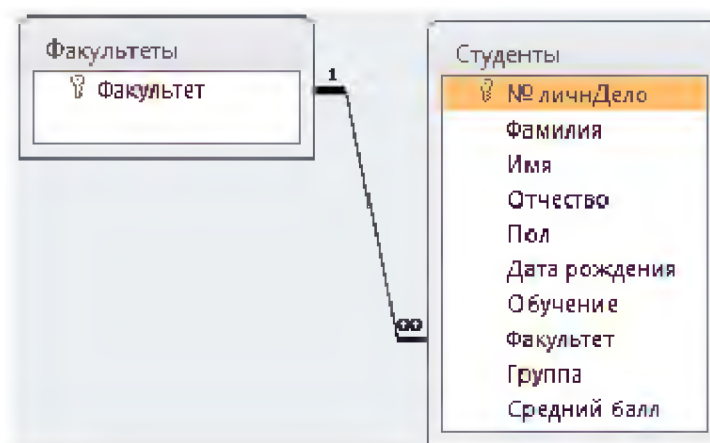


Рисунок 1.0 – Таблицы БД «Университет»

Связь 1-∞ означает, что в таблице **Факультеты** название факультета может встретиться только один раз, а в таблице **Студенты** название факультета может встретиться много раз (ряд студентов учится на одном факультете).

Алгоритм действий в лабораторной работе № 1:

1. Создать новую БД с именем **Университет** в папке **C:/Work** или в папке **C:/User**.
2. Конструировать таблицу **Факультеты**.
3. Конструировать таблицу **Студенты**.
4. Связать таблицы через атрибут (поле) **Факультет**.
5. Заполнить данными таблицу **Факультеты**.
6. Заполнить данными таблицу **Студенты**.

7. Выполнить сортировку, замену, фильтрацию данных в таблице **Студенты**.

1.1 Создание новой базы данных

Создайте новую базу данных, выполнив следующие действия:

- Выполните команду **Новая пустая база данных – Новая база данных (Ok)**.
- Справа внизу **Новая база данных** укажите имя файла: **Университет** (расширение не указывать) в папке **C:\Work**.
- Нажмите **Создать (Ok)**.

1.2 Создание таблицы Факультеты

Создайте таблицу Факультеты, выполнив следующие действия:

- Меню **Создание – Конструктор таблиц**.
- В открывшемся окне **Таблица1** определить поля следующим образом (рисунок 1.1):
 - введите в ячейку столбца **Имя поля** имя первого поля – **Факультет**;
 - в ячейке столбца **Тип данных** оставьте значение **Текстовый**;
 - переключитесь на панель **Свойства поля** (клавишей **F6** или мышью) и введите размер поля **30**;
 - поле **Факультет** сделайте ключевым, используя кнопку **ключа**.

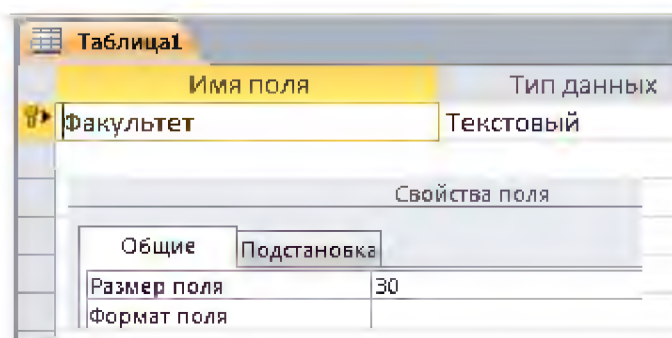


Рисунок 1.1 – Конструктор таблицы

- Закройте окно, т.е. выйти из конструктора, переименовав **Таблицу1** в **Факультеты**.

1.3 Создание таблицы Студенты

Аналогично создайте таблицу **Студенты** в соответствии со структурой (таблица 1).

Таблица 1 – Структура таблицы **Студенты**

Имя поля	Тип данных	Свойства поля
№ личнДело	Текстовый	Ключевое поле; размер поля – 5 ; обязательное поле – да ; пустые строки – нет ; индексированное поле – да (Совпадения не допускаются)
Фамилия	Текстовый	Размер поля – 15 ; обязательное поле – да ; пустые строки – нет
Имя	Текстовый	Размер поля – 10 ; обязательное поле – да ; пустые строки – нет
Отчество	Текстовый	Размер поля – 15 , обязательное поле – да ; пустые строки – нет
Пол	Мастер подстановок	В мастере подстановки: фиксированный набор значений – число столбцов – 1 ; ячейка1 – м ; ячейка2 – ж . Размер поля – 1 , обязательное поле – да ; пустые строки – нет . (Затем Тип данных станет Текстовый , так как на вкладке Подстановка источником строк будет текст «м»; «ж»).
Дата рождения	Дата/время	Формат поля – краткий формат даты
Обучение	Мастер подстановок	Фиксированный набор значений: очное, заочное . Размер поля – 7 , обязательное поле – да ; пустые строки – нет
Факультет	Мастер подстановок	Объект из другой таблицы или запроса ; Показать таблицы Факультеты ; Выбранные поля Факультет , Далее... Готово. <i>Связь пока не создавать.</i> Размер поля – 30 , обязательное поле – да ; пустые строки – нет
Группа	Текстовый	Размер поля – 6 ; обязательное поле – да ; пустые строки – нет ;
Средний балл	Числовой	Одинарное с плавающей точкой, формат – основной , число десятичных знаков – 2

- Выйти из конструктора, сохраняя таблицу под именем **Студенты**.

1.4 Создание схемы данных

С рабочего поля удалите (закройте) таблицы **Студенты** и **Факультеты**.

- Вызовите окно **Схема данных** и добавьте таблицы **Факультеты** и **Студенты**. Установите связь между таблицами (перетащите атрибут **Факультет** из таблицы **Факультеты** в таблицу **Студенты** к строке **Факультет**). Установите тип отношения **Один-ко-многим** (один факультет может быть указан у многих студентов). Установите параметры: **Каскадное обновление связанных полей** и **Каскадное удаление связанных записей** (рисунок 1.2).

Примечание: Для изменения связи или свойств полей в таблицах, необходимо разорвать связь между таблицами, сделав правый щелчок на линии связи.

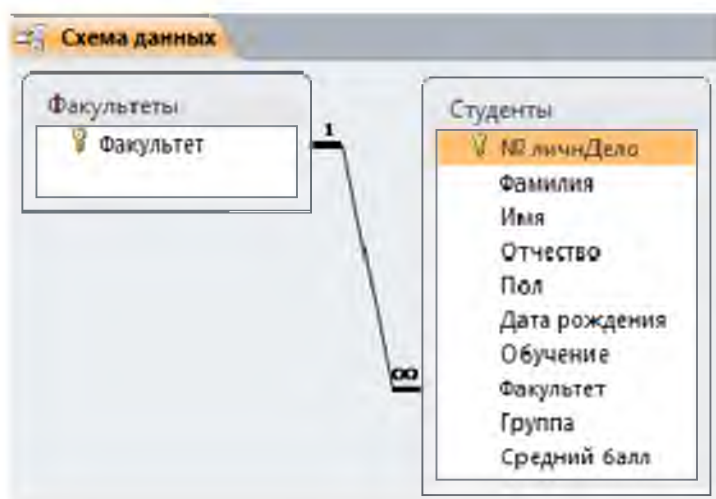


Рисунок 1.2 – Схема данных

- Сохраните макет **схемы данных**.

1.5 Заполнение таблицы Факультеты

Откройте таблицу **Факультеты** двойным щелчком.

- Введите данные в таблицу **Факультеты** и закройте окно с таблицей (рисунок 1.3).

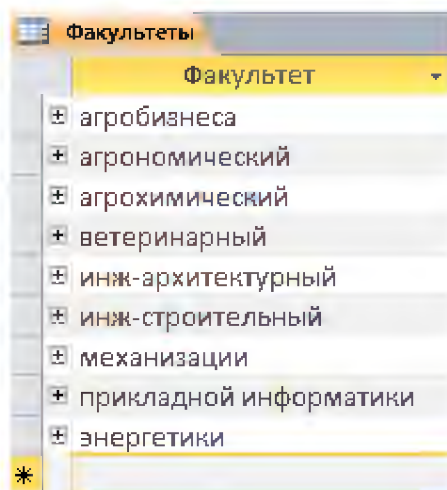


Рисунок 1.3 – Таблица **Факультеты**

1.6 Заполнение таблицы Студенты

Откройте таблицу **Студенты**.

- Введите данные (10–12 строк записей) в таблицу и закройте окно с таблицей (рисунок 1.4).

Студенты									
№ личн.д.	Фамилия	Имя	Отчество	Пол	Дата рожд.	Обучен	Факультет	Группа	Средний
0016	Иваненко	Наталья	Семеновна	ж	30.06.1996	очное	агрохимический	АХ 1402	4,38
0018	Иванова	Мария	Фроловна	ж	15.08.1997	очное	ветеринарный	ВЭ 1501	4,25
1224	Долгих	Павел	Петрович	м	13.06.1997	очное	прикладной информатики	ПИ 1501	4,21
1508	Гуревич	Григорий	Алексеевич	м	27.11.1995	очное	прикладной информатики	ПИ 1521	4,35
2124	Курочкин	Леонид	Дмитриевич	м	15.04.1997	очное	энергетики	ФЭ 1402	4,15
2135	Кучерин	Владимир	Львович	м	03.07.1998	очное	энергетики	ЭС 1502	3,75
0423	Копыл	Анна	Александровна	ж	07.02.1997	очное	ветеринарный	ВЭ 1501	4,15
0245	Чабан	Антон	Пантелеевич	м	05.11.1995	заочное	энергетики	ФЭ 1441	3,75
1013	Онищенко	Константин	Георгиевич	м	11.04.1996	заочное	энергетики	ФЭ 1441	3,25

Рисунок 1.4 – Таблица **Студенты**

1.7 Редактирование таблицы Студенты

Откройте таблицу **Студенты** и отредактируйте данные:

- в поле **Фамилия** замените "Долгих" на "Журавлев"; затем опять замените обратно "Журавлев" на "Долгих";
- в поле **Дата рождения**, замените цифры года "1997" на "1998";
- удалите четвертую запись (строку) таблицы (*лучше не удалять; если удалили, то восстановите*).
- Отсортируйте записи:
 - по возрастанию данных поля **Фамилия**;
 - по убыванию поля **Дата рождения**.
- Используйте фильтрацию по выделению для вывода на экран записей по мужскому полу. Установите курсор на поле **Пол**. Далее команды **Главная – Выделение – Равно «м»**.
- Удалите установленный фильтр, нажав **Фильтр**.
- Используя расширенный фильтр, в диалоговом окне выберите поле **Группа**, укажите критерий "**ПИ 1501**" и установите сортировку поля **Дата рождения** по убыванию. Для этого используйте команду **Главная – Дополнительно – Расширенный фильтр**. После изменения параметров фильтра необходимо выполнить команду **Применить фильтр** (рисунок 1.5).

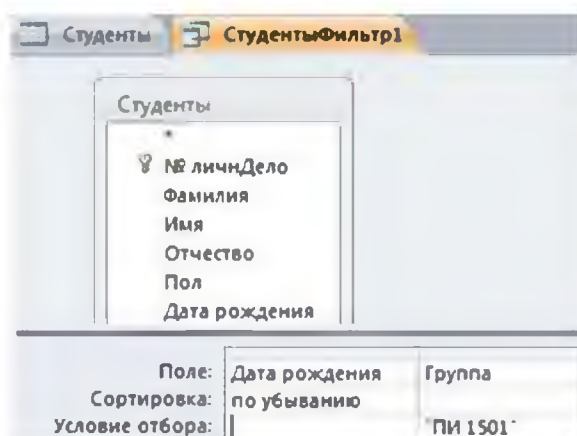


Рисунок 1.5 – Расширенный фильтр для таблицы **Студенты**

- Закройте таблицу **СтудентыФильтр1**.

2 СОЗДАНИЕ ПРОСТЫХ ФОРМ, ЗАПРОСОВ И ОТЧЕТОВ

Цель: научиться создавать простые пользовательские формы, вводить, корректировать и осуществлять поиск данных в формах; научиться создавать простые запросы на выборку данных и готовить списки данных в виде документов, готовых к передаче другим пользователям.

Алгоритм действий в лабораторной работе № 2:

1. Конструировать простую форму **Студенты** (вид в один столбец).
2. Конструировать простую форму **Факультеты** (вид ленточный).
3. Работа с формой **Студенты**.
4. Создать простой запрос по выборке студентов указанного пола (запрос **Девушки**).
5. Создать простой запрос по выборке студентов указанной группы (запрос **ПИ 1501**).
6. Создать отчет на базе запроса (отчет **Список девушек**).
7. Создать отчет с итогами (**Отчет по среднему баллу**).
8. Редактирование отчета (**Отчет2 по среднему баллу**).

2.1 Конструирование формы **Список студентов**

Откройте пункт **Создание – Формы – Мастер форм**.

- В поле **Таблицы и запросы** выберите **Таблица: Студенты**, перенесите все доступные поля в окно **Выбранные поля** нажатием кнопки **>>**. Нажмите **Далее**. Выберите внешний вид формы **В один столбец**. Задайте имя формы **Список студентов**, выберите пункт **Изменить макет формы** и нажмите **Готово**.

- Измените (уменьшите) высоту рамки для надписи **Список студентов**. Поднимите вверх полосу **Область данных** к рамке **Список студентов** (рисунок 2.1).

Заголовок формы	
Список студентов	
Область данных	
№ личнДело	№ личнДело
Фамилия	Фамилия
Имя	Имя
Отчество	Отчество
Пол	Пол ▼
Дата рождения	Дата рождения
Обучение	Обучение ▼
Факультет	Факультет ▼
Группа	Группа
Средний балл	Средний балл
Примечание формы	

Рисунок 2.1 – Режим конструктора для таблицы **Студенты**

- Выделите правое поле **Пол**. Уменьшите ширину объекта с данными. То же самое сделайте с полем **Обучение**.
- Содержимое правых полей **Пол**, **Средний балл** установите в середине ячейки (Главная – Форматирование текста).
- Переключитесь в **Режим формы** (для этого правый щелчок на объект **Список студентов**), просмотрите и при необходимости отредактируйте, переключившись обратно в режим **Конструктор** (рисунок 2.2).

Список студентов

№ личнДело	2135
Фамилия	Кучерин
Имя	Владимир
Отчество	Львович
Пол	м ▾
Дата рождения	03.07.1998
Обучение	очное ▾
Факультет	энергетики ▾
Группа	ЭС 1502
Средний балл	3,75

Рисунок 2.2 – Режим формы для таблицы Студенты

2.2 Конструирование формы Список факультетов

Создайте форму **Список факультетов**, разместив в ней одно поле **Факультет** из таблицы **Факультеты**. Внешний вид – **Ленточный** (рисунок 2.3).

Рисунок 2.3 – Режим конструктора для таблицы **Факультеты**

- Откройте форму **Список факультетов** и добавьте новый факультет – **Экономический** (рисунок 2.4).

Список факультетов
агробизнеса
агрономический
агрохимический
ветеринарный
инж-архитектурный
инж-строительный
механизации
прикладной информатики
энергетики

Рисунок 2.4 – Режим формы для таблицы **Факультеты**

2.3 Работа с формой **Список студентов**

Откройте форму **Список студентов**.

- Просмотрите поочередно все записи.
- Перейдите сразу к первой записи, последней и пятой.
- Добавьте 3–5 записей по новым студентам с факультетов: агробизнеса, экономического и агрохимического (данные по прочим полям на ваше усмотрение).
 - В форме **Список студентов** с помощью команды **Поиск** найдите запись данных по студенту **Кучерин**. Измените ему средний бал на **4,57**.
 - Закройте форму.

2.4 Создание простого запроса по выбору пола

Создайте и просмотрите простой запрос **Девушки** на выборку всех данных из таблицы **Студенты** по критерию "ж" поля **Пол**. Сортировка по фамилии. Поле **Пол** не должно отображаться. Для этого:

- Откройте пункт **Создание – Запросы – Мастер запросов**. Выбрать **Простой запрос**.

- В поле **Таблицы и запросы** выберите **Таблица: Студенты**, перенесите все **Доступные поля** в окно **Выбранные поля** нажатием кнопки >>. Нажмите **Далее**.
- Выберите отчет **Подробный**. Нажмите **Далее**.
- Задайте имя запроса **Девушки**, установите **Изменить макет запроса** и нажмите **Готово**.
- Выключите вывод на экран поля **Пол** (рисунок 2.5).
- В параметр **Условие отбора** введите букву “ж”.
- Установите сортировку по возрастанию в поле **Фамилия**.
- Сохраните созданный запрос и откройте его в режиме **Таблица** (рисунок 2.6).

Поле:	№ личнДело	Фамилия	Имя	Отчество	Пол
Имя таблицы:	Студенты	Студенты	Студенты	Студенты	Студенты
Сортировка:		по возрастанию			
Вывод на экран:	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>
Условие отбора:					ж

Рисунок 2.5 – Параметры запроса **Девушки**

№ личн	Фамилия	Имя	Отчество	Дата рожд	Обуче
0016	Иваненко	Наталья	Семеновна	30.06.1996	очное
0018	Иванова	Мария	Фроловна	15.08.1997	очное
0423	Копыл	Анна	Александровна	07.02.1997	очное

Рисунок 2.6 – Реализация запроса **Девушки** в режиме таблицы

2.5 Создание простого запроса по выбору группы

Создайте и просмотрите простой запрос с именем "Запрос ПИ 1501" на выборку данных по полям **Фамилия**, **Имя**, **Отчество**, **Обучение**, **Факультет**, **Группа** из таблицы **Студенты** по критерию "**Прикладной информатики**" поля **Факультет** и "**ПИ 1501**" поля **Группа**. Поле **Факультет** и **Группа** не должны отображаться. Сортировка по полям **Фамилия**, **Имя** и **Отчество** (рисунок 2.7).

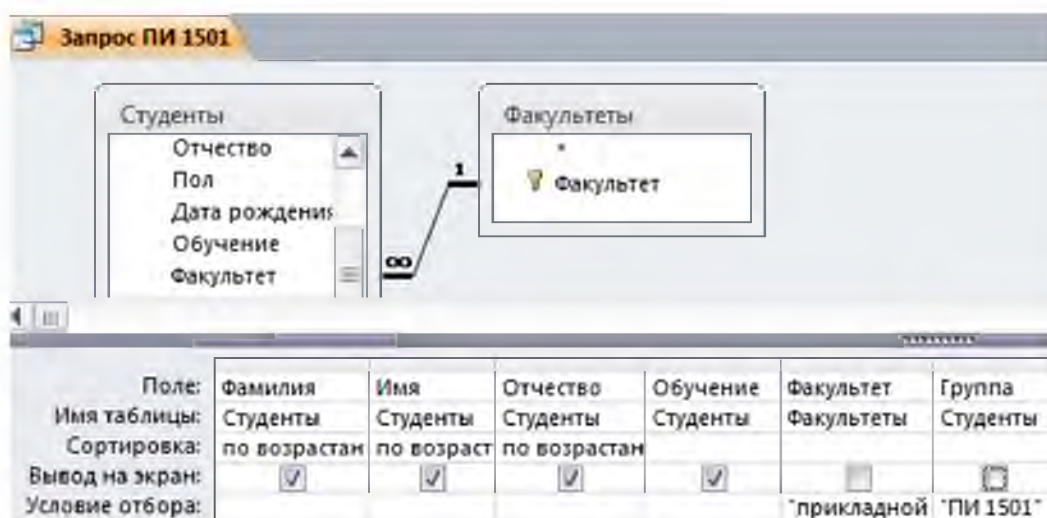


Рисунок 2.7 – Параметры запроса ПИ 1501

- Обратите внимание: в поле [**Факультет**] на второй строке указана таблица **Факультеты**.
- Результатом выполнения запроса является следующая таблица (рисунок 2.8):

Запрос ПИ 1501			
Фамилия	Имя	Отчество	Обучение
Гуревич	Григорий	Алексеевич	очное
Долгих	Павел	Петрович	очное

Рисунок 2.8 – Реализация запроса ПИ 1501 в режиме таблицы

- Если в запросе указать вывод на экран полей **Факультет** и **Группа**, то будет выведена следующая таблица (рисунок 2.9):

Запрос ПИ 1501					
Фамил ▾	Имя ▾	Отчество ▾	Обуч ▾	Факультет ▾	Группа ▾
Гуревич	Григорий	Алексеевич	очное	прикладной информатики	ПИ 1501
Долгих	Павел	Петрович	очное	прикладной информатики	ПИ 1501

Рисунок 2.9 – Реализация запроса **ПИ 1501** с показом всех полей

2.6 Создание простого отчета на основе запроса

Создайте отчет на базе запроса **Девушки**. Для этого:

- В главном окне выберите группу **Создание – Отчеты – Мастер отчетов**.
- В поле **Таблицы и запросы** выберите **Запрос: Девушки**, перенесите все **Доступные** поля в окно **Выбранные поля** нажатием кнопки **>>**. Нажмите **Далее**.
- Из уровней группировки добавьте **Факультет** и **Группа**. Нажмите **Далее** (рисунок 2.10).

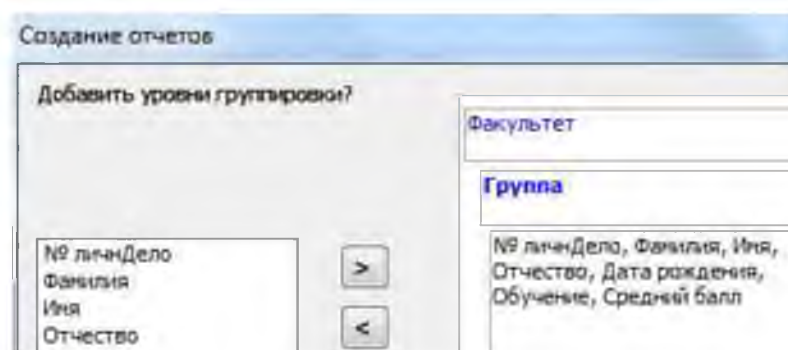


Рисунок 2.10 – Формирование группировок в отчете

- Задайте параметр сортировки для полей **Фамилия**, **Имя**, **Отчество**. Нажмите **Далее** (рисунок 2.11).

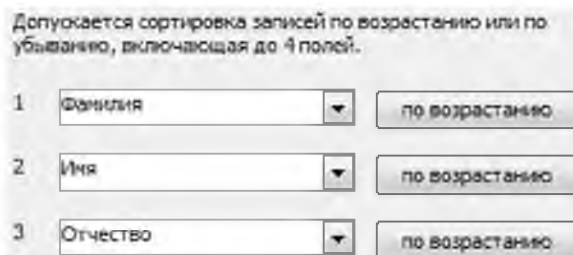


Рисунок 2.11 – Формирование сортировки в отчете

- Выберите макет – **блок**, ориентация – **Книжная**. Нажмите **Далее**.
- Введите название отчета **Список девушек**, установить **Просмотреть отчет**, нажать **Далее**.
- Для получения красивого представления отчета в **Конструкторе** внесите следующие изменения (рисунок 2.12):

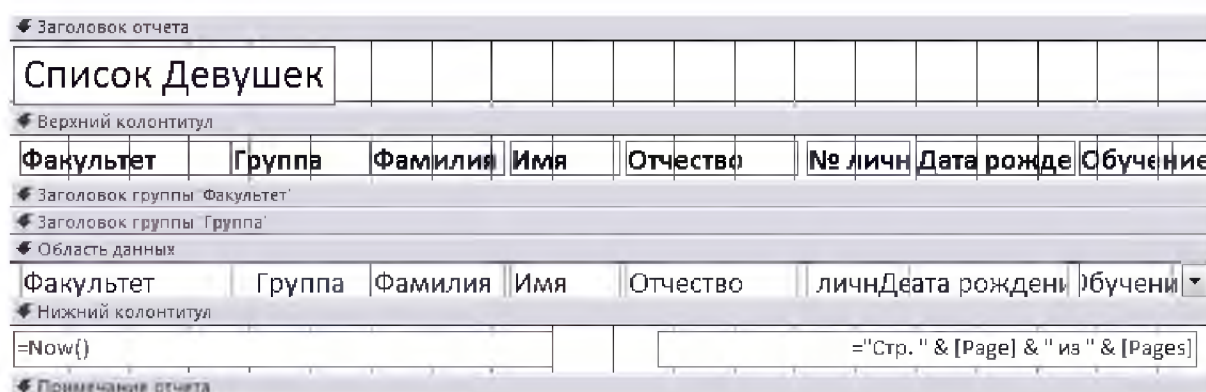


Рисунок 2.12 – Конструирование верхнего колонтитула и полей (область данных) в отчете

- Откройте отчет **Список девушек** (рисунок 2.13):

Список Девушек							
Факультет	Группа	Фамилия	Имя	Отчество	№ личн	Дата рожде	
агрохимический	АХ 1402	Иваненко	Наталья	Семеновна	0016	30.06.1996	
ветеринарный	ВЭ 1501	Иванова	Мария	Фроловна	0018	15.08.1997	
		Копыл	Анна	Александровна	0423	07.02.1997	

Рисунок 2.13 – Отчет

2.7 Создание отчета с итогами

Создайте **Отчет по среднему баллу** по всем студентам на основе таблицы **Студенты**. Не включать в отчет поля **№ личнДела**, **Пол** и **Дата рождения**. Группировка по полям **Факультет** и **Группа**. Сортировка по фамилии, имени и отчеству. Формировать итоги (вызывается на шаге сортировки) по полю **Средний балл** – **Avg**, **Min**, **Max** (рисунок 2.14):

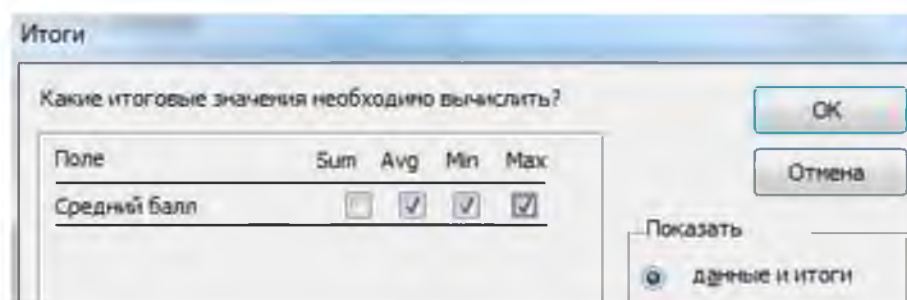


Рисунок 2.14 – Итоговые значения в отчете

- Отредактируйте отчет, используя **Конструктор**, следующим образом (рисунок 2.15).

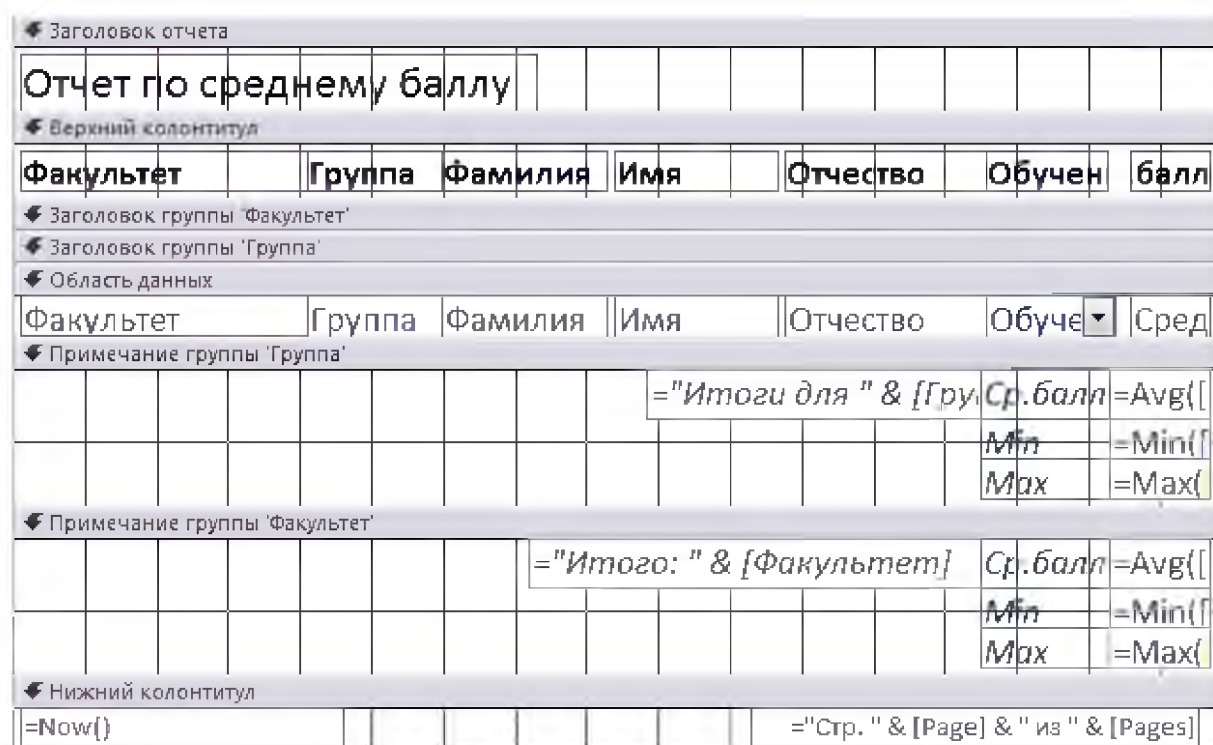


Рисунок 2.15 – Конструирование отчета

- Формируйте примечания, выделив поле и вызвав **Окно свойств**. На вкладке **Данные**, используйте значок с тремя точками – это **Построитель выражений** (рисунок 2.16).

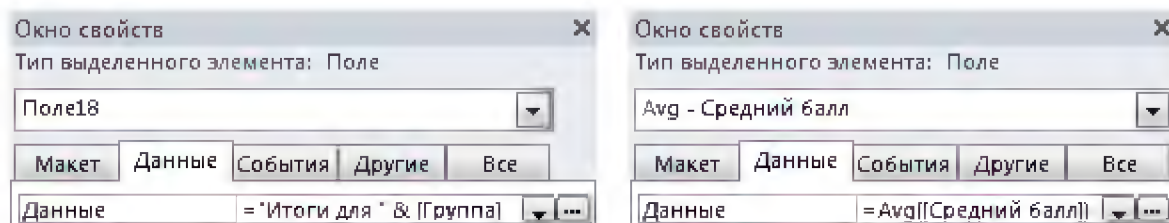


Рисунок 2.16 – Конструирование вычисляемых полей в Окне свойств

- В результате получается следующий фрагмент отчета (рисунок 2.17).

Отчет по среднему баллу						
Факультет	Группа	Фамилия	Имя	Отчество	Обучение	балл
агрохимический	АХ 1402	Иваненко	Наталья	Семеновна	очное	4,38
Итоги для АХ 1402					Ср.балл	4,38
					Min	4,38
					Max	4,38
Итого: агрохимический					Ср.балл	4,38
					Min	4,38
					Max	4,38

Рисунок 2.17 – Фрагмент отчета

2.8 Редактирование отчета с итогами

Скопировать отчет под именем **Отчет2 по среднему баллу**. Используя **Конструктор**, изменить **Примечания** следующим образом (рисунок 2.18):

Факультет	Группа	Фамилия	Имя	Отчество	Обуче	Сред
Примечание группы "Группа"						
		= "Итоги для "	[Гру	Ср.балл	редни	Min
				редни	Max	=Max(
Примечание группы "Факультет"						
		= "Итого: "	[Факультет]	Ср.балл	редни	Min
				редни	Max	=Max(

Рисунок 2.18 – Конструирование вычисляемых полей в отчете

При этом, содержание полей редактируются в **Окне свойств** во вкладке **Данные**. Кроме того, формируем рамки, используя **Элемент управления** «прямоугольник» и устанавливая **Положение** прямоугольника «На задний план». В результате получится **Отчет2 по среднему баллу** следующего вида (рисунок 2.19):

Отчет2 по среднему баллу							
Факультет	Группа	Фамилия	Имя	Отчество		Обучение	балл
агрохимический	АХ 1402	Иваненко	Наталья	Семеновна		очное	4,38
Итоги для АХ 1402		Ср.балл	4,38	Min	4,38	Max	4,38
Итого: агрохимический		Ср.балл	4,38	Min	4,38	Max	4,38
ветеринарный	ВЭ 1501	Иванова	Мария	Фроловна		очное	4,25
		Копыл	Анна	Александровна		очное	4,15
Итоги для ВЭ 1501		Ср.балл	4,20	Min	4,15	Max	4,25
Итого: ветеринарный		Ср.балл	4,20	Min	4,15	Max	4,25

Рисунок 2.19 – Фрагмент отчета с другим расположением вычисляемых полей

3 СОЗДАНИЕ СЛОЖНЫХ ФОРМ, ЗАПРОСОВ И ОТЧЕТОВ

Цель: научиться создавать сложные пользовательские формы, вводить, корректировать и осуществлять поиск данных в формах; научиться создавать сложные запросы на выборку данных и готовить списки данных в виде документов.

Конкретно:

I. Создать сложный запрос на отбор сведений о студентах, средний балл которых находится в заданном диапазоне. Запрос должен вызываться из формы, в которой вводятся значения верхнего и нижнего предела среднего балла для отбора записей. В форме необходимо сделать кнопку просмотра отчета, сформированного на основании созданного запроса.

II. Создать форму для выбора факультета, создать запрос для выбора студентов указанного факультета, создать отчет со списком выбранных студентов и открыть этот отчет.

Алгоритм действий в лабораторной работе № 3:

1. Конструировать форму с двумя полями для ввода данных с нижним и верхним значениями среднего балла (форма **Средний балл**).
2. Конструировать запрос с условием отбора среднего балла между указанными пределами (**Запрос на средний балл**).
3. Создать отчет **Список студентов по среднему баллу**, источник данных **Запрос средний балл**.
4. Вставить в форму **Средний балл** кнопку **Просмотр отчета**, чтобы при нажатии на кнопку открывался отчет **Список студентов по среднему баллу**.
5. Создать форму для выбора факультета (форма **Студенты факультета**).
6. Создать запрос для выбора студентов указанного факультета (**Запрос студенты факультета**).
7. Создать отчет со списком выбранных студентов (**Отчет студенты факультета**).
8. Через форму **Студенты факультета** открыть **Отчет студенты факультета**.

3.1 Создание формы Средний балл

Создадим форму **Средний балл** для ввода верхнего и нижнего предела баллов. Для этого:

- Откройте пункт **Создание формы** в режиме **Конструктора**.
- Создайте надпись **Введите средний балл**. Для этого на панели инструментов **Элементы управления** выберите кнопку **Надпись**, в верхней части формы проведите прямоугольник и введите текст надписи. Поместите ее по центру горизонтали формы. Затем пометьте текст и задайте шрифт – **Tahoma**, 16 пт.
- Аналогично, ниже надписи **Введите средний балл** создайте два поля для указания значения балла с именами **Нижнее значение** и **Верхнее значение**; формат поля – **основной**; число десятичных знаков **2**. Для назначения имён и параметров объектов откройте **Окно свойств**. Пока не надо создавать кнопку с надписью *Просмотр отчета*, это сделаем позже (рисунок 3.1).

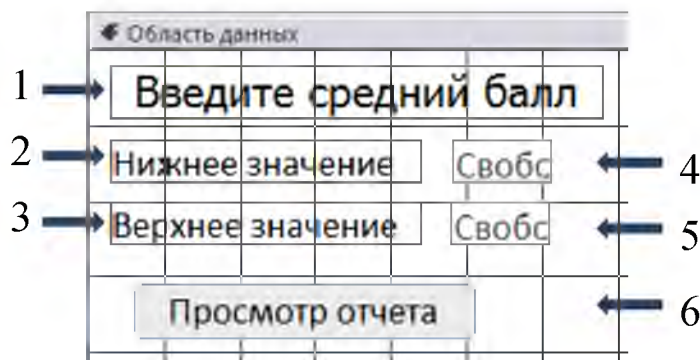


Рисунок 3.1 – Конструирование формы **Средний балл** с двумя полями и кнопкой

- На рисунке 3.1 введены обозначения:
 - 1 – Объект: Надпись;
Имя: Текст ВводСредБалл;
Подпись: Введите средний балл.
 - 2 – Объект: Надпись;
Имя: Текст НижЗнач;
Подпись: Нижнее значение.

- 3 – Объект: Надпись;
Имя: Текст ВерхЗнач;
Подпись: Верхнее значение.
- 4 – Объект: Поле;
Имя: Нижнее значение;
- 5 – Объект: Поле;
Имя: Верхнее значение;
- 6 – Объект: Кнопка;
Имя: ОткрОтчетСписСтуд;
Подпись: Просмотр отчета.

Здесь не надо путать понятия **Имя** и **Подпись**. **Имя** – это название (идентификатор) объекта, **Подпись** – это некая оболочка (одежда) объекта. Важным является именно **Имя** объекта. Сущность объекта передается через **Имя**. (Есть такая фраза: «Волк в овечьей шкуре»).

- Сохраните форму под именем Средний балл. Если открыть форму, то она примет следующий вид (рисунок 3.2):

The screenshot shows a web form with a title bar that says 'Средний балл'. Below the title bar is the text 'Введите средний балл'. There are two input fields: the first is labeled 'Нижнее значение' and the second is labeled 'Верхнее значение'. At the bottom of the form is a button labeled 'Просмотр отчета'.

Рисунок 3.2 – Вид формы Средний балл

3.2 Создание запроса со сложным условием отбора

Для организации отбора нужных записей откройте пункт **Создание – Запросы – Мастер запросов – Простой запрос**.

- В поле **Таблицы и запросы** выберите **Таблица: Студенты**, перенесите все **Доступные поля** в окно **Выбранные поля** нажатием кнопки [**>>**]. Нажмите **Далее** и выберите отчет **Подробный**. Нажмите **Далее**.
- Задайте имя запроса **Запрос на средний балл**, установите **Изменить макет запроса** и нажмите **Готово**.
- В появившемся окне в столбце **Средний балл** установите курсор в строку **Условие отбора**. На панели инструментов **Настройка запроса** выберите пункт **Построитель** (рисунок 3.3).

Поле:	Факультет	Группа	Средний балл
Имя таблицы:	Факультеты	Студенты	Студенты
Сортировка:			
Вывод на экран:	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>
Условие отбора:			> [Формы]![Средний балл]

Рисунок 3.3 – Фрагмент запроса для указания **Условия отбора**

- В появившемся окне необходимо построить выражение отбора записей, средний балл в которых больше нижнего значения и меньше верхнего.

Ввод знаков логических операций осуществляется нажатием соответствующих кнопок. Выбор полей осуществляется из списка объектов. Выберите в группе **Формы**, форму **Средний балл**, из полей формы выберите **Нижнее значение** и нажмите кнопку **Вставить** и т. д. (рисунок 3.4). Общий вид выражения должен быть следующий:

*>[Формы]![Средний балл]![Нижнее значение] And
< [Формы]![Средний балл]![Верхнее значение].*

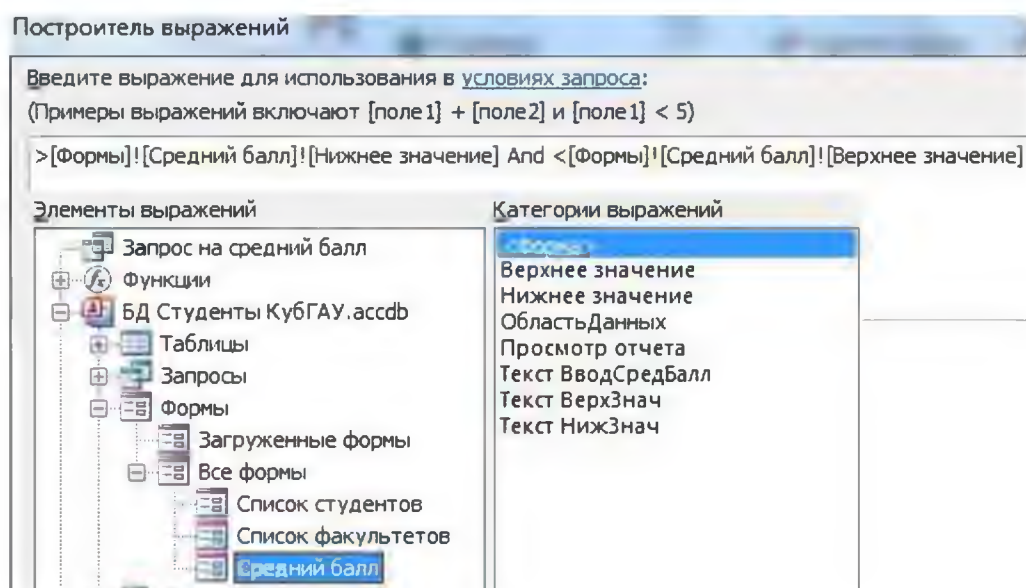


Рисунок 3.4 – Построитель выражений для отбора записей по запросу

- Реализация запроса при указании среднего балла между 3,5 и 5, имеет вид (рисунок 3.5).

Запрос на средний балл						
Фамилия	Имя	Отчество	П	Факультет	Группа	Сред
Иванов	Иван	Иванович	м	инж-архитектурный	АИ 1501	4,25
Иваненко	Наталья	Семеновна	ж	агрохимический	АХ 1402	4,38
Иванова	Мария	Фроловна	ж	ветеринарный	ВЭ 1501	4,25
Чабан	Антон	Пантелеевич	м	энергетики	ФЭ 1441	3,75

Рисунок 3.5 – Фрагмент запроса для отбора записей

3.3 Создание отчета на основе сложного запроса

Для отображения результатов запроса в виде отчета необходимо с помощью мастера создать отчет: название – **Список студентов по среднему баллу**, источник данных – **Запрос средний балл**. Группировка по факультетам, сортировка – по среднему баллу, итоги по полю **Средний балл сессии** – **Avg**. Для этого:

- Отредактируйте отчет, используя **Конструктор**, следующим образом (рисунок 3.6).

а

Список студентов по среднему баллу

Заголовок отчета

Список студентов по среднему баллу

Верхний колонтитул

Факультет	Сред. балл	Группа	Фамилия	Имя
-----------	------------	--------	---------	-----

Заголовок группы 'Факультет'

Область данных

Факультет	Сред. балл	Группа	Фамилия	Имя
-----------	------------	--------	---------	-----

Примечание группы 'Факультет'

= "Итого для факультета " & [Факультет] & " (" & Count(*)

Нижний колонтитул

=Now()

Примечание отчета

= "Стр. " & [P

б

Имя	Отчество	П	Дата рождения	Обучение
-----	----------	---	---------------	----------

Имя Отчество о дата рождения обучени

& Count(*) .Ср. балл= =Avg(

= "Стр. " & [Page] & " из " & [Pages]

Рисунок 3.6 – Конструирование отчета:
а – начало (левая часть); б – продолжение

- Проверим вид отчета, введя средний балл в поле нижнего значения **3,5** и верхнего **5** (рисунок 3.7):

Список студентов по среднему баллу					
Факультет	Сред. балл	Группа	Фамилия	Имя	Отчество
агрохимический	4,38	АХ 1402	Иваненко	Наталья	Семеновна
Итого для факультета агрохимический (1 зап)					Ср.балл= 4,38
ветеринарный	4,15	ВЭ 1501	Копыл	Анна	Александровна
	4,25	ВЭ 1501	Иванова	Мария	Фроловна
Итого для факультета ветеринарный (2 зап)					Ср.балл= 4,20

Рисунок 3.7 – Вид отчета (фрагмент)

3.4 Создание формы с кнопкой для просмотра отчета

Вставить в форму **Средний балл** кнопку **Просмотр отчета**, чтобы при нажатии на кнопку открывался отчет **Список студентов по среднему баллу**. Для этого:

- Откройте в режиме конструктора ранее созданную форму **Средний балл**. В нижней части создайте кнопку **Просмотр отчета**.
- В открывшемся окне **Создание кнопки** выберите в разделе **Категории** – **Работу с отчетом**, а в разделе **Действия** – **Открыть отчет** и нажать **Далее** (рисунок 3.8).

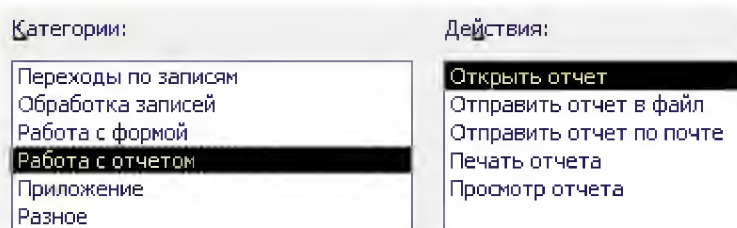


Рисунок 3.8 – Выбор действий при нажатии на кнопку

- Выбираем **Список студентов по среднему баллу** и нажать **Далее** (рисунок 3.7).

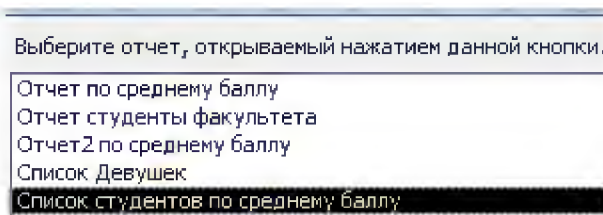


Рисунок 3.7 – Выбор названия отчета

- Набрать текст **Просмотр отчета**, размещаемый на кнопке, и нажать Далее (рисунок 3.8).

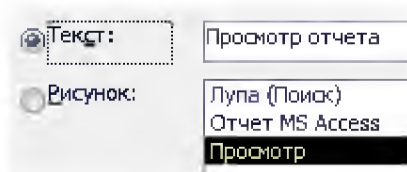


Рисунок 3.8 – Набор текста для размещения на кнопке

- Задать имя кнопки **ОткрОтчетСписСтуд** и нажать Готово.
- Прделанные операции по созданию кнопки (при возникновении ошибки открытия отчета) можно проверить в **Окне свойств** на вкладке **Макет** (рисунок 3.9), на вкладке **Другие** (рисунок 3.10) и на вкладке **События** (рисунок 3.11).

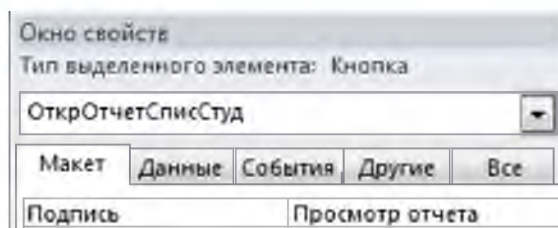


Рисунок 3.9 – Название подписи кнопки

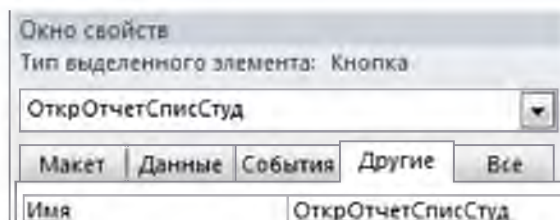


Рисунок 3.10 – Название имени кнопки

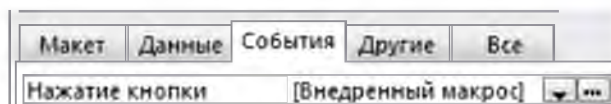


Рисунок 3.11 – Событие при нажатии кнопки

- Нажав на значок с тремя точками на вкладке **События**, можно увидеть содержание макроса (рисунок 3.12).

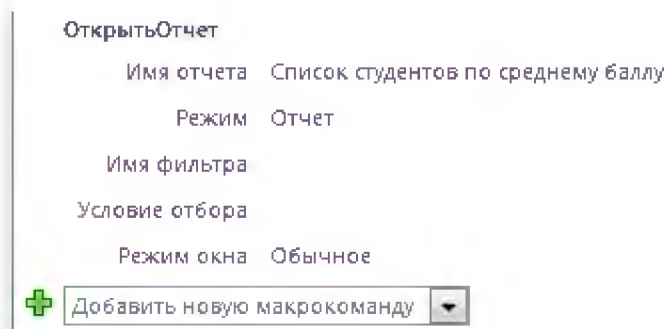


Рисунок 3.12 – Содержание макроса при нажатии кнопки

- Если **Имя отчета** представлено кодами букв (рисунок 3.13), то можно выполнить следующие операции:
 - ✓ нажать на стрелку **Добавить новую макрокоманду** (см. рисунок 3.12);
 - ✓ в появившемся списке выбрать **Открыть отчет**;
 - ✓ далее из списка отчетов выбрать имя **Список студентов по среднему баллу**;
 - ✓ прежний макрос с кодами имени отчета удалить (рисунок 3.14).



Рисунок 3.13 – Представление имени отчета кодами букв

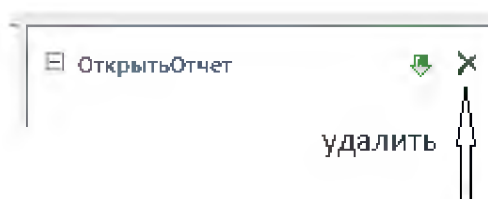


Рисунок 3.14 – Удаление ненужного макроса

- Сохраните конструктор формы **Средний балл**.
- Проверить работу формы **Средний балл**. Для этого откройте форму, введите в поле нижнего значения **3,5** и верхнего **5** (рисунок 3.15) и проверьте работоспособность кнопки **Просмотр отчета** (см. рисунок 3.5).

Рисунок 3.15 – Форма Средний балл

3.5 Создание формы, имеющей поле со списком

Создадим форму **Студенты факультета** для выбора факультета и последующего просмотра списка студентов выбранного факультета. Для этого:

- Откройте пункт **Создание формы** в режиме **Конструктора**.
- Создайте надпись **Студенты факультета**. Для этого на панели инструментов **Элементы управления** выберите кнопку **Надпись**, в верхней части формы проведите прямоугольник и введите текст надписи. Поместите ее по центру горизонтали формы. Затем пометьте текст и задайте шрифт – **Tahoma**, 16 пт. (рисунок 3.16):

Рисунок 3.16 – Конструктор формы с объектами: надпись, поле со списком, кнопка

- Ниже надписи **Студенты факультета** создайте поле со списком для выбора факультета из имеющегося списка. При создании **Поля со списком** автоматически появится рамка для надписи. Укажем надпись **Выбор факультета** (это подпись в макете окна свойств).
- Выделить рамку **Поля со списком** (Свободный) и в **Окне свойств** на вкладке **Другие** указать имя **ВыбранФак** (рисунок 3.17).

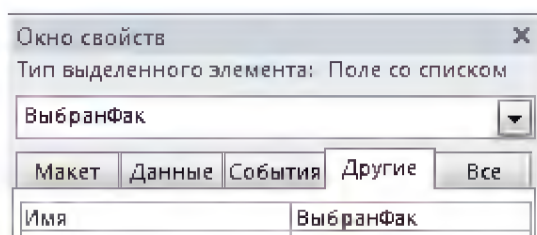


Рисунок 3.17 – Назначение свойств **Поля со списком** в форме

- Сохраните форму под именем **Студенты факультета**.

3.6 Создание запроса на основе поля со списком, включенного в форму

Для организации отбора нужных записей из таблицы **Студенты** (студенты только указанного факультета) откройте пункт **Создание – Запросы – Мастер запросов – Простой запрос**.

- В поле **Таблицы и запросы** выберите **Таблица: Студенты**, перенесите все **Доступные поля** в окно **Выбранные поля** нажатием кнопки [**>>**]. Нажмите **Далее** и выберите отчет **Подробный**. Нажмите **Далее**.

- Задайте имя запроса **Запрос студенты факультета**, установите **Изменить макет запроса** и нажмите **Готово**.

- В появившемся окне в столбце [**Факультет**] установите курсор в строку **Условие отбора** (рисунок 3.18) и на панели инструментов **Настройка запроса** выберите пункт **Построитель** и создайте выражение (рисунок 3.19).

[Формы] ! [Студенты факультета] ! [ВыбранФак]

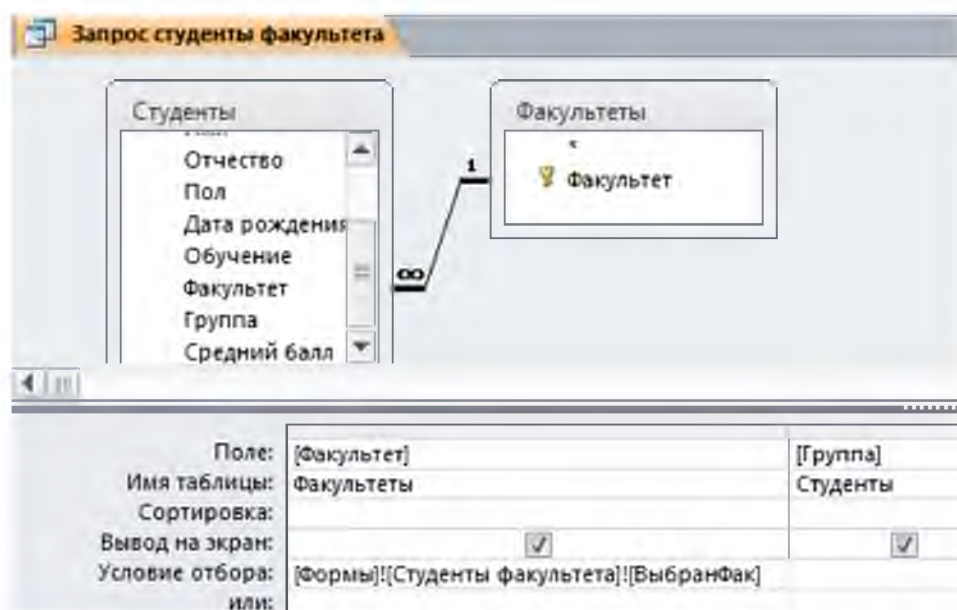


Рисунок 3.18 – Фрагмент запроса для указания **Условия отбора**

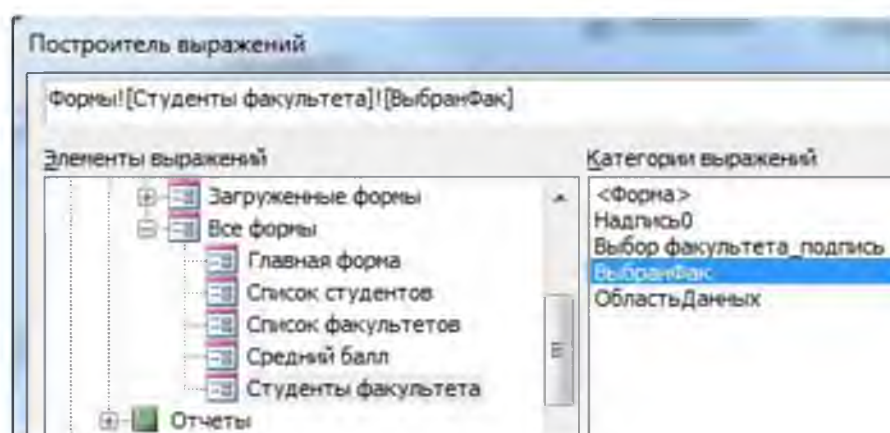


Рисунок 3.19 – Построитель выражения для выбора факультета

- Проверим выполнение запроса для факультета прикладной информатики (рисунок 3.20).

Запрос студенты факультета					
№	Фамилия	Имя	Факультет	Группа	Сред
1224	Долгих	Павел	прикладной информати	ПИ 1501	4,21
1508	Гуревич	Григорий	прикладной информати	ПИ 1501	4,35

Рисунок 3.20 – Вид запроса в режиме таблица (фрагмент)

3.7 Создание отчета со списком студентов

Для отображения результатов запроса необходимо с помощью мастера создать отчет: название – **Отчет студенты факультета**, источник данных – **Запрос студенты факультета**.

- Выбираем все доступные поля; Вид представления – **Студенты**; группировки нет; сортировки – **Фамилия**; макет – табличный; имя отчета – **Отчет студенты факультета**; изменить макет отчета – нажать Готово.

- В **Конструкторе** отредактировать отчет соответствующим образом (рисунок 3.21), при этом из области данных следует перенести в заголовок отчета поле **Факультет** и в **Окне свойств** на вкладке **Данные** установить **Факультет** (рисунок 3.22).

а

Отчет студенты факультета

1 1 2 1 3 1 4 1 5 1 6 1 7 1 8 1 9 1 10 1 11 1 12

Заголовок отчета

Отчет: студенты факультета Факультет

Верхний колонтитул

№ дела	Фамилия	Имя	Отчество	Группа	Пол

Область данных

личнДе	Фамилия	Имя	Отчество	Группа	Пол
--------	---------	-----	----------	--------	-----

Нижний колонтитул

=Now()

= "Стр. " & [P

Примечание отчета

б

Пол	Дата рожд	Обучение	Сред балл

Пол	Дата рожден	Обучени	дний
-----	-------------	---------	------

, " & [Page] & " из " & [Pages]

Рисунок 3.21 – Редактирование отчета в конструкторе:
а – начало; б – продолжение

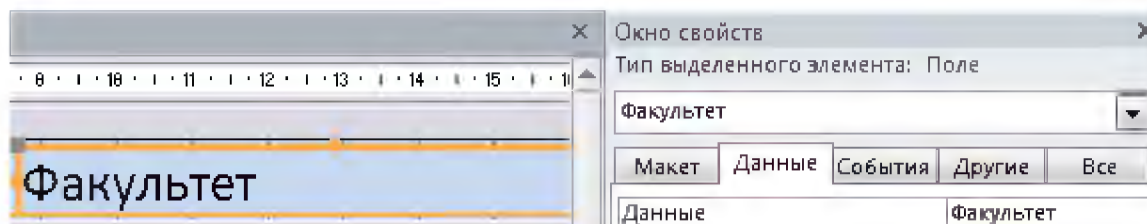


Рисунок 3.22 – Установка свойств поля **Факультет**

- Проверим отчет для факультета прикладной информатики (рисунок 3.23).

Отчет: студенты факультета прикладной информатики							
№ дела	Фамилия	Имя	Отчество	Группа		Обучение	Сред балл
1508	Гуревич	Григорий	Алексеевич	ПИ 1501		очное	4,35
1224	Долгих	Павел	Петрович	ПИ 1501		очное	4,21

Рисунок 3.23 – Список студентов выбранного факультета (фрагмент)

3.8 Просмотр отчета через кнопку формы

Откройте в режиме конструктора ранее созданную форму **Студенты факультета**. В нижней части создайте кнопку **Просмотр студентов факультета** (см. рисунок 3.16), выполнив следующие действия:

- В открывшемся окне **Создание кнопки** выберите в разделе **Категории** – **Работу с отчетом**, а в разделе **Действия** – **Открыть отчет** и нажать **Далее** (рисунок 3.24).



Рисунок 3.24 – Выбор действий при нажатии на кнопку

- Выбираем **Отчет студенты факультета** и нажать Далее (рисунок 3.25).

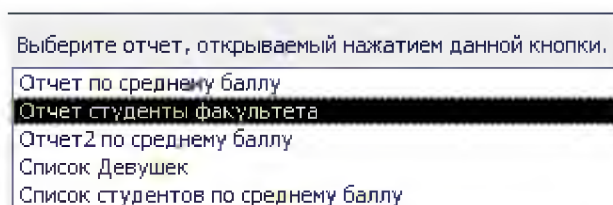


Рисунок 3.25 – Выбор названия отчета

- Набрать текст **Просмотр студентов факультета**, размещаемый на кнопке, и нажать Далее (рисунок 3.26).

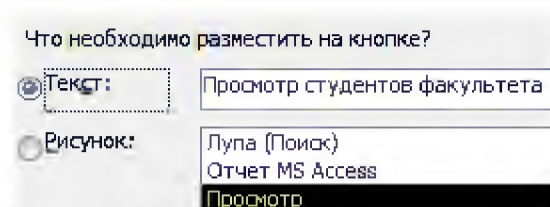


Рисунок 3.26 – Набор текста для размещения на кнопке

- Задать имя кнопки **ОткрСтудФак** и нажать Готово (рисунок 3.27).

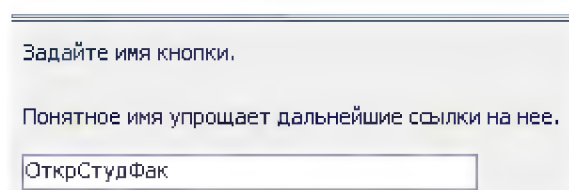


Рисунок 3.27 – Набор имени кнопки

- Прделанные операции по созданию кнопки (при возникновении ошибки открытия отчета) можно проверить в **Окне свойств** на вкладках:

Макет – Подпись – Просмотр студентов факультета;

Другие – Имя – ОткрСтудФак;

События – Нажатие кнопки – Внедренный макрос (рисунок 3.28).

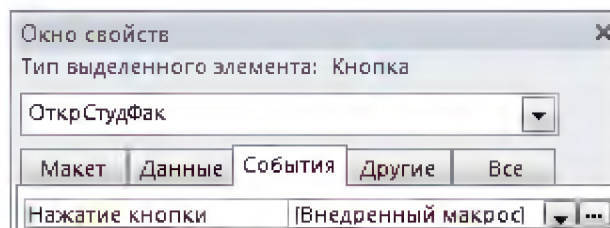


Рисунок 3.28 – Событие при нажатии кнопки

- Нажав на значок с тремя точками на вкладке **События**, можно увидеть содержание макроса (рисунок 3.29).

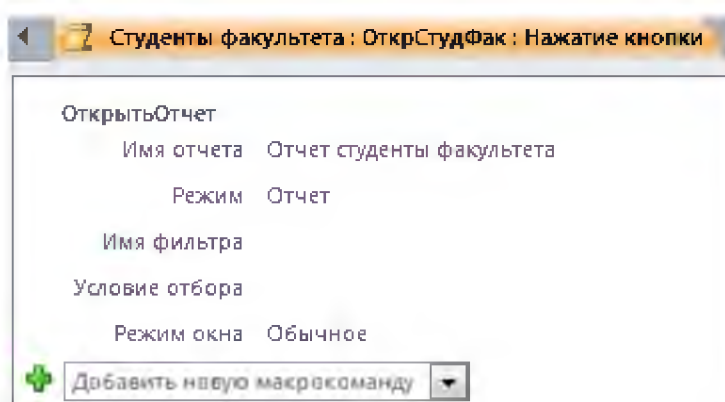


Рисунок 3.29 – Содержание макроса при нажатии кнопки

- Закройте с сохранением конструктор формы **Студенты факультета**.
- Проверим работу формы **Студенты факультета**. Для этого откройте форму.
- Выберите факультет, например, **прикладной информатики** (рисунок 3.30).

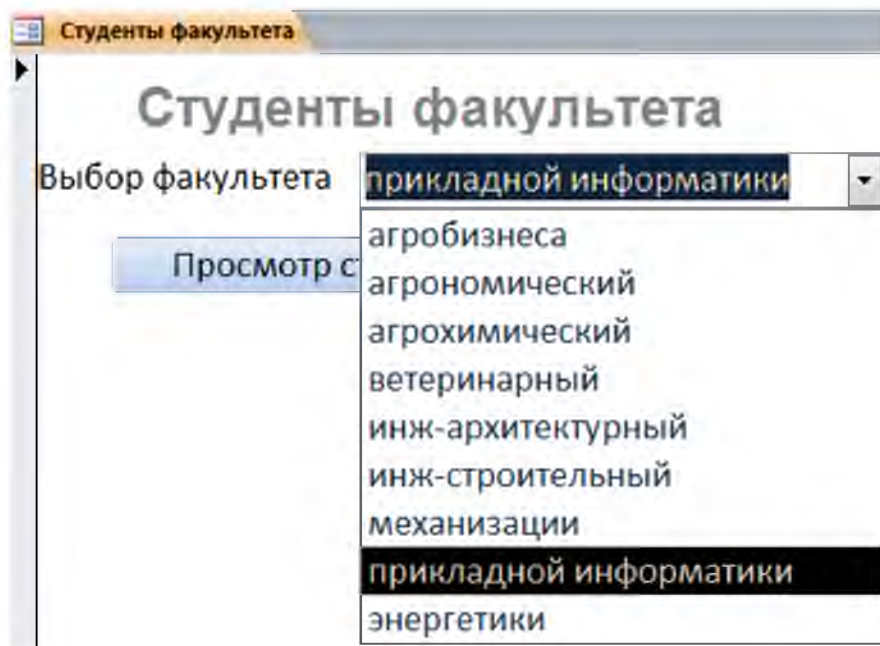


Рисунок 3.30 – Выбор факультета в форме

- Нажатие на кнопку **Просмотр студентов факультета** открывает отчет из списка студентов выбранного факультета (см. рисунок 3.23).

4 СОЗДАНИЕ ГЛАВНОЙ ФОРМЫ БД. СОЗДАНИЕ НОВОЙ ТАБЛИЦЫ И ВЫВОД ФОТОГРАФИЙ

Цель: научиться создавать сложную главную форму, дополнить базу данных новой таблицей, выводить на экран фотографии студентов.

Конкретно:

- I. Создать главную форму с кнопками для вывода на экран ранее созданных объектов, вставить в форму рисунки.
- II. Дополнить базу данных новой таблицей **Анкеты**, в которой выделить поле для указания имени файла фотографии студента для вывода на экран.

Алгоритм действий в лабораторной работе № 4:

1. Конструировать главную форму с кнопками для вывода на экран ранее созданных объектов (**Главная форма**).
2. Вставить в **Главную форму** два рисунка, относящихся к университету (*рисКубГАУ.jpg* , *рисКриница.jpg*).
3. Создать новую таблицу **Анкеты** с полем **фото** для текста имени файла фотографии студента.
4. Создать форму для вывода анкетных данных с фотографиями студентов.

4.1 Создание главной формы с кнопками

Откройте пункт **Создание – Формы – Конструктор форм**.

- Создайте надпись **База студентов КубГАУ («Университет»)**. Для этого на появившейся панели инструментов выберите кнопку **Надпись**, в верхней части формы проведите прямоугольник и введите текст надписи. Разместите ее по середине формы. Затем пометьте его и задайте шрифт – **Tahoma**, 16 пт. Растяните поле области данных до желаемого размера (рисунок 4.1).

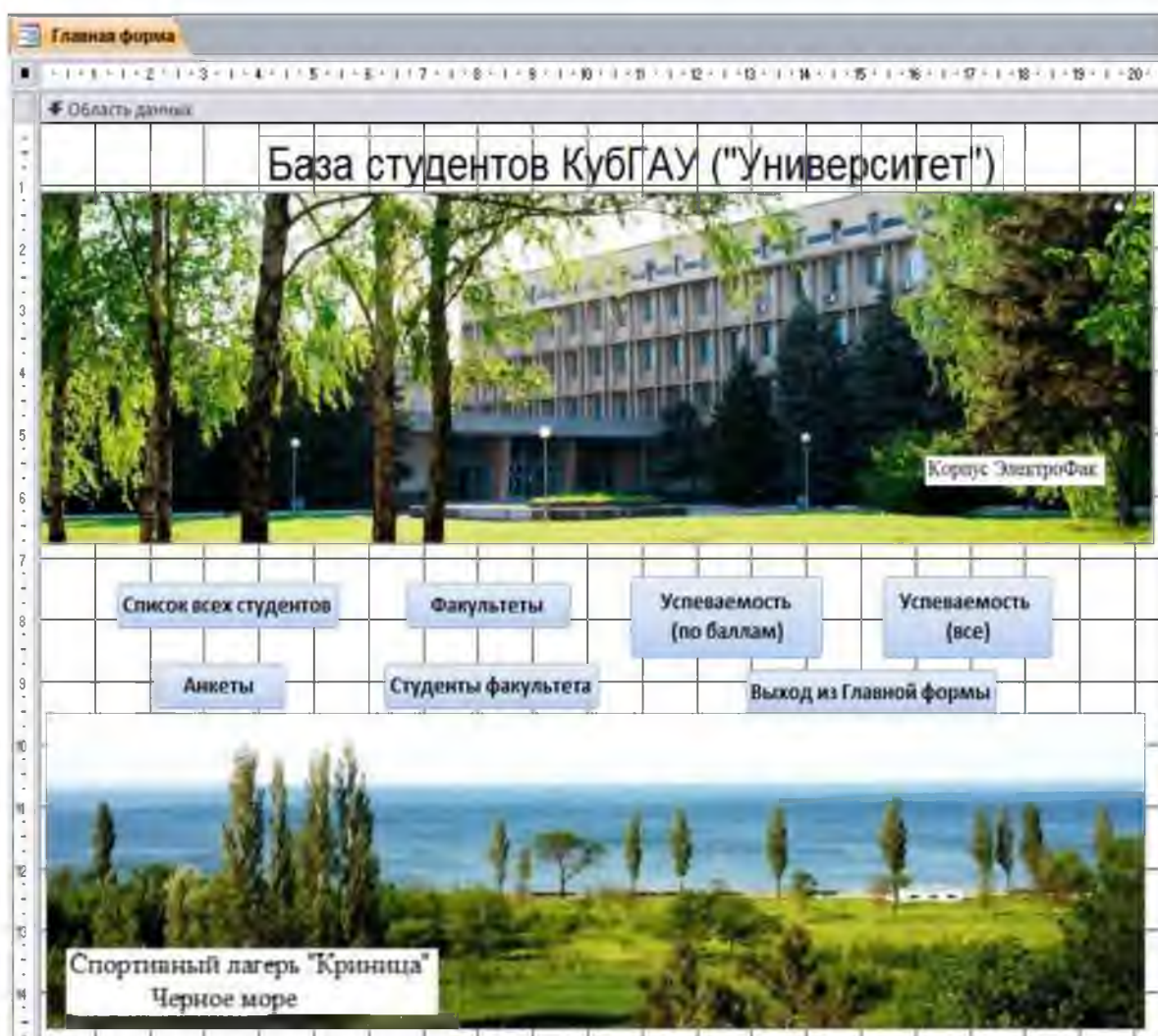


Рисунок 4.1 – Конструктор Главной формы

- Создайте кнопку для вызова на просмотр и редактирование **формы Список студентов**. На панели инструментов выберите **Кнопка**, в левой части формы проведите прямоугольник и проведите следующий диалог с мастером создания кнопки.
- В параметре **Категории** выберите **Работа с формой**, в параметре **Действия** – **Открыть форму**, и нажмите Далее (рисунок 4.2).

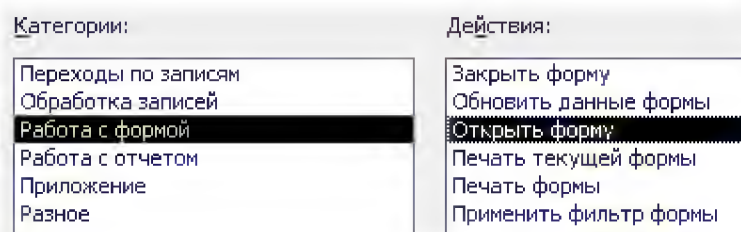


Рисунок 4.2 – Начало диалога с мастером создания кнопки

- Выберите форму **Список студентов** и нажмите Далее. Выберите **Открыть форму** и **Показать все записи** и нажмите Далее. Укажите текст **Открыть форму** и нажмите Готово. На кнопке разместите текст **Список студентов**, задайте имя кнопки – **ОткрСтудентов**, и нажмите Готово. Подпись на кнопке **Список студентов** и имя кнопки – **ОткрСтудентов** можно указать в **Окне свойств** на вкладках **Макет** и **Другие**, предварительно выделив кнопку.

На вкладке **События**, нажав на значок с тремя точками (рисунок 4.3), можно посмотреть **Внедренный макрос** (рисунок 4.4). Макрос – это заранее заготовленная макрокоманда.

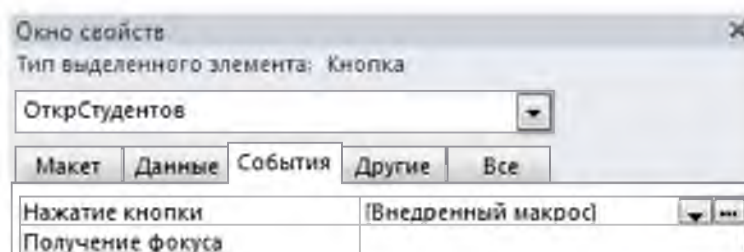


Рисунок 4.3 – Вкладка События на кнопке

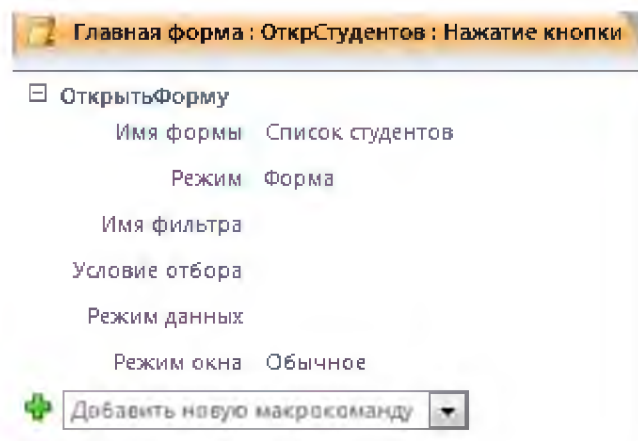


Рисунок 4.4 – Макрос Открыть форму

- Аналогично создайте кнопку **Факультеты**, открывающий форму **Список факультетов**. Имя кнопки – **ОткрФакультеты**.

- Аналогично создайте кнопку **Успеваемость (по баллам)**, открывающий форму **Средний балл**. Имя кнопки – **ОткрСредБалл**.

- Аналогично создайте кнопку **Успеваемость (все)**, открывающий отчет **Отчет2 по среднему баллу**. Имя кнопки – **ОткрОтчет2**.
- Создайте в нижней части формы кнопку выхода. На панели инструментов выберите кнопку **Кнопка**, в нижней части формы создайте прямоугольник и проведите диалог с мастером создания кнопки. В параметре **Категории** выберите **Приложение**, в параметре **Действия** – **Выйти из приложения** и нажмите Далее. На кнопке разместите текст **Выход из главной формы**. Задайте имя кнопки – **ВыхГлавФорм**.
- В окне свойств кнопки **ВыхГлавФорм** на вкладке **События** можно посмотреть макрос при нажатии кнопки (рисунок 4.5 и 4.6).

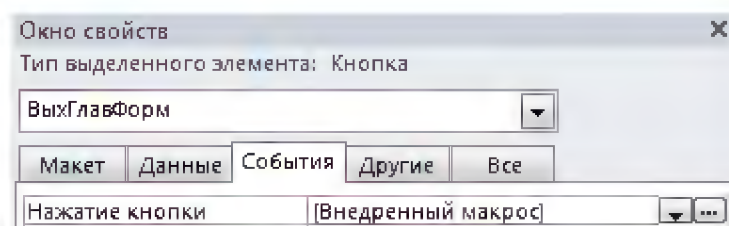


Рисунок 4.5 – Вкладка **Событие** в окне свойств

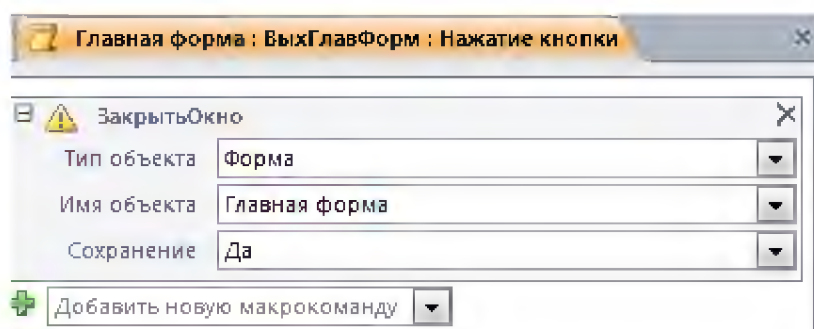


Рисунок 4.6 – Макрос для выхода из **Главной формы**

4.2 Вставка рисунка в Главную форму

Для вставки рисунка в Главную форму выполняем следующие действия:

- В разделе **Конструктор** выделяем элемент **Рисунок** и прямоугольником отмечаем место для рисунка (см. рисунок 4.1).
- Не убирая выделение прямоугольника, активируем элемент **Вставить изображение** и через обзор указываем рисунок **рисКубГАУ.jpg**, расположенный в папке нашей БД. Заметим, что данный рисунок взят с сайта www.kubsau.ru.
- В окне свойств устанавливаем следующие параметры на вкладке **Макет** (рисунок 4.7):
 - Вывод на экран – Да;
 - Тип рисунка – Внедренное;
 - Рисунок – рисКубГАУ.jpg;
 - Установка размеров – Вписать в рамку;
 - Выравнивание рисунка – По центру.

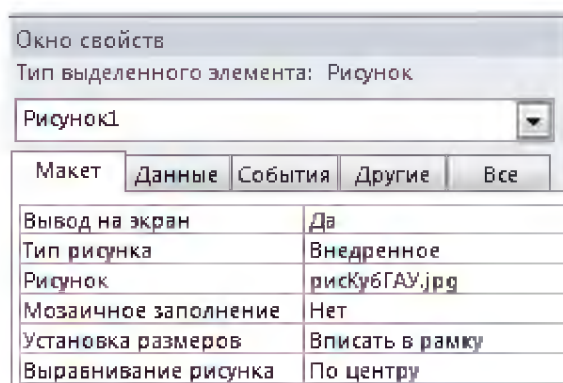


Рисунок 4.7 – Параметры для рисунка на вкладке **Макет**

- Аналогично вставляем в Главную форму рисунок **рисКриница.jpg**.
- Закройте окно конструктора создания формы с сохранением ее под именем **Главная форма**.
- Проверьте работоспособность **Главной формы**.

4.3 Создание новой таблицы

Создать новую таблицу **Анкеты** с полем **фото** для текста имени файла фотографии студента (таблица 2).

Таблица 2. Структура таблицы **Анкеты**

Имя поля	Тип данных	Свойства поля
№ личнДело	Текстовый	Ключевое поле; размер поля – 5 ; обязательное поле – да ; пустые строки – нет ; индексированное поле – Да (Совпадения не допускаются)
Адрес	Текстовый	Размер поля – 25 ; обязательное поле – да ; пустые строки – нет
фото	Текстовый	Размер поля – 15 ; обязательное поле – нет ; пустые строки – да

- Создать связь 1 – 1 между таблицами **Студенты** и **Анкеты** через поле **№ личнДело** (рисунок 4.8).

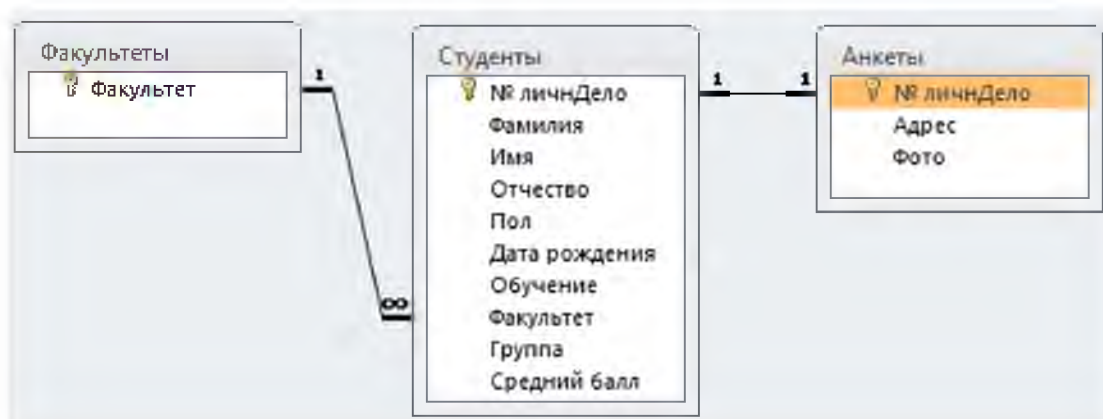


Рисунок 4.8 – Схема данных с тремя таблицами

- Заполнить таблицу **Анкеты** данными, при этом значение в поле **№ личнДело** должно совпадать с таблицей **Студенты** (рисунок 4.9).

Анкеты		
№ лич	Адрес	Фото
0015	Общежитие 4	Иванов.jpg
0016	Общежитие 1	Иваненко.jpg
0018	Общежитие 5	Иванова.jpg
0245	г. Горячий ключ	
0423	Общежитие 5	Копыл.jpg
1013	ст-ца Ивановская	Онищенко.jpg
1224	Общежитие 20	Долгих.jpg
1508	Общежитие 20	Гуревич.jpg
2124	Общежитие 3	
2135	Общежитие 3	Кучерин.jpg

Рисунок 4.9 – Заполнение таблицы **Анкеты**

- В поле **Фото** указаны имена графических файлов, которые расположены в папке данной базы данных. Обратите внимание, что у некоторых студентов нет фото.
- Создадим **Запрос Анкеты**, чтобы включить в виртуальную таблицу поле **Фамилия**, ссылающееся на таблицу **Студенты** (рисунок 4.10).

Запрос Анкеты

Студенты

- № личнДело
- Фамилия
- Имя
- Отчество
- Пол

Анкеты

- № личнДело
- Адрес
- Фото

Поле:	№ личнДело	Фамилия	Адрес	Фото
Имя таблицы:	Анкеты	Студенты	Анкеты	Анкеты
Сортировка:				
Вывод на экран:	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>

Рисунок 4.10 – Конструктор запроса **Запрос Анкеты**

- Виртуальная таблица **Запрос Анкеты** имеет следующий вид (рисунок 4.11).

Запрос Анкеты			
№ лич	Фамилия	Адрес	Фото
0015	Иванов	Общежитие 4	Иванов.jpg
0016	Иваненко	Общежитие 1	Иваненко.jpg
0018	Иванова	Общежитие 5	Иванова.jpg
0245	Чабан	г. Горячий ключ	
0423	Копыл	Общежитие 5	Копыл.jpg
1013	Онищенко	ст-ца Ивановская	Онищенко.jpg
1224	Долгих	Общежитие 20	Долгих.jpg
1508	Гуревич	Общежитие 20	Гуревич.jpg
2124	Курочкин	Общежитие 3	
2135	Кучерин	Общежитие 3	Кучерин.jpg

Рисунок 4.11 – Режим таблицы для запроса **Запрос Анкеты**

4.4 Создание формы для вывода фотографий

Создать форму для вывода анкетных данных с фотографией студентов. Для этого:

- Откройте пункт **Создание формы** в режиме **Мастера форм** на основе запроса **Запрос Анкеты**. Сохраните форму под именем **Анкеты**.
- Отредактируйте **Конструктором** форму **Анкеты** следующим образом (рисунок 4.12).

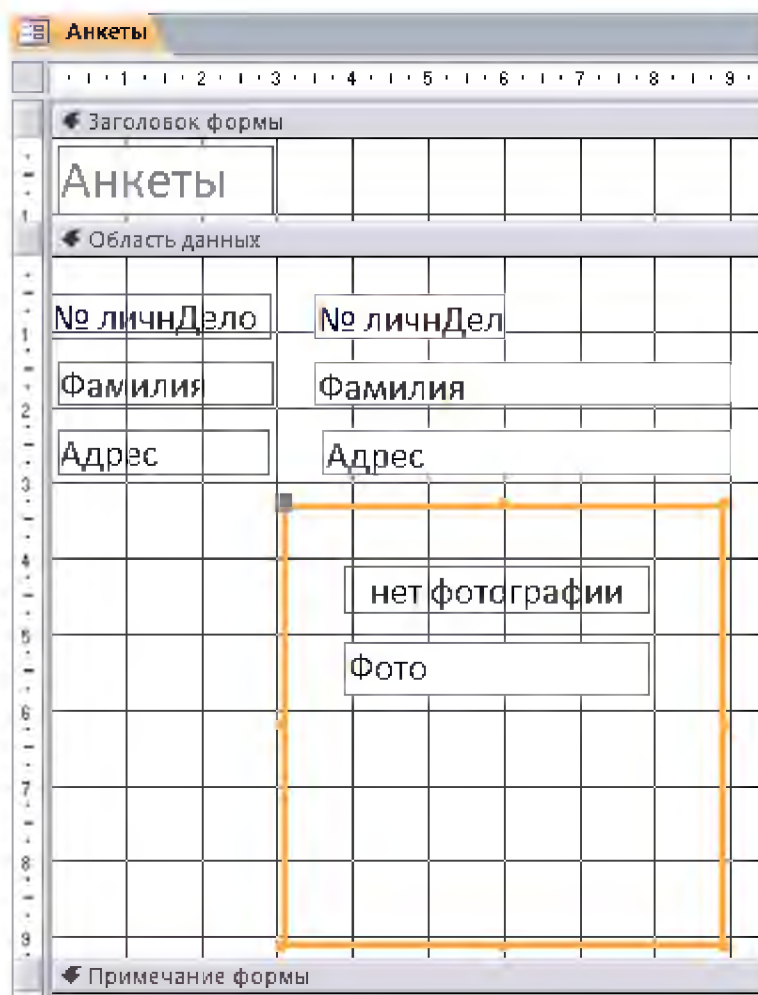


Рисунок 4.12 – Конструирование формы **Анкеты**

• На рисунке 4.12 надо обратить внимание на три рамки в нижней части:

- большая рамка,
- рамка с надписью «Фото»,
- рамка «нет фотографии».

Большая рамка (это объект-рисунок с именем *ImageFrame*) предназначена для размещения на выделенной площади фотографии. Рамка «Фото» (это объект-поле с именем *ImagePath*) предназначена для указания имени графического файла. Рамка «нет фотографии» (это объект-надпись с именем *ErrorMsg*) предназначена для вывода сообщения при отсутствии файла с фотографией.

Для объектов-рамок надо установить необходимые свойства: *Конструктор* – *Сервис* – *Страница свойств*.

Отметим самые важные свойства:

Рисунок *ImageFrame*, вкладки:

- Макет – Вывод на экран – **Да**
 Тип рисунка – **Связанное**
 Установка размеров – **Вписать в рамку**
 Выравнивание рисунка – **По центру**
- Другие – Имя – **ImageFrame**

Поле *ImagePath*, вкладки:

- Макет – Вывод на экран – **Нет**
- Данные – Данные – **фото**
- События – После обновления – **Процедура обработки событий**
- Другие – Имя – **ImagePath**

Надпись *ErrorMsg*, вкладки:

- Макет – Подпись – надпись – фото отсутствует
 Вывод на экран – **Да**
- Другие – Имя – **ErrorMsg**

Свойства всей формы (нажать на значок с прямоугольной точкой на пересечении линеек формы):

Тип выделенного элемента – *Форма* – вкладки:

- Данные – Источник записей – запрос **Запрос Анкеты**
- События – Текущая запись – **Процедура обработки событий**
 После обновления – **Процедура обработки событий**

• Текст кода процедур обработки событий на языке VBA надо смотреть в меню **Конструктор – Сервис – Посмотреть код**. Код следующий:

'Листинг программы вывода фотографий при смене записей.

'Имена файлов фотографий указываются в таблице Анкеты

'в поле "Фото"

Option Compare Database

Option Explicit 'Имена переменных должны объявляться

```

Private Sub Form_AfterUpdate()
    showImageFrame 'отображение фото(событие: смена записи)
End Sub

```

```

Private Sub Form_RecordExit(Cancel As Integer)
    'При переходе между записями надпись ErrorMessage скрывается,
    'чтобы избежать ненужного мелькания.
    ErrorMessage.Visible = False
End Sub

```

```

Private Sub ImagePath_AfterUpdate()
    'Отображает фото (событие: смена имени файла фото)
    showImageFrame 'отображение фото
End Sub

```

```

Private Sub Form_Current()
    'Если имеется фотография, она отображается в форме.
    'Если файл-фото не существует, либо если поле пусто,
    'надпись ErrorMessage выводит сообщение "Фото отсутствует".
    On Error Resume Next
    ErrorMessage.Visible = False
    If Not IsNull(Me![ImagePath]) Then 'если есть имя файла фото
        showImageFrame 'вызов процедуры отображения фото
    Else
        hideImageFrame 'сокрытие рамки для фото
        ErrorMessage.Caption = "Фото отсутствует" 'сообщение
        ErrorMessage.Visible = True 'показать сообщение
    End If
End Sub

```

```

Sub showErrorMessage()
    'Выводит сообщение ErrorMessage, если файл фото недоступен
    If Not IsNull(Me![ImagePath]) Then
        ErrorMessage.Visible = False
    Else
        ErrorMessage.Visible = True
    End If
End Sub

```



```

Sub hideImageFrame()
    Me![ImageFrame].Visible = False 'Скрывает рамку для фото
End Sub

```

```

Sub showImageFrame()
    'Выводит фото
    Me![ImageFrame].Picture = fName 'вставляет в рамку фото
    Me![ImageFrame].Visible = True 'показывает фото
End Sub

```

```

Private Function fName() As String
    'Анализ полного пути файла фотографии.
    'Файл фотографии размещается в текущей папке БД, либо
    'в любой другой папке, в этом случае указывается полный путь.
    'Имя файла записано в поле Фото таблицы Сотрудники,
    'а в форме это поле обозначено ImagePath
    'Если файл расположен в текущей папке, то к имени
    'добавляется полный путь
    Dim path As String
    path = CurrentProject.path 'путь текущей папки БазыДанных
    fName = Me![ImagePath] 'путь, записанный в поле фото
    If (InStr(1, fName, ":") = 0) And (InStr(1, fName, "\\") = 0) Then
        fName = path & "\" & fName
    End If
End Function

```

- Сконструированная форма, которой присвоено имя **Анкеты** будет выглядеть следующим образом (рисунок 4.13):

Анкеты

№ личнДело	0018
Фамилия	Иванова
Адрес	Общежитие 5




Рисунок 4.13 – Форма Анкеты

СПИСОК ЛИТЕРАТУРЫ

1. Когаловский М. Р. Перспективные технологии информационных систем / М. Р. Когаловский. – М. : ДМК Пресс; М: Компания АйТи, 2003. – 288 с.
2. Стандарт ISO/IEC 2382-1. Термины и определения стандарта ISO/IEC 2382-1 [Электронный ресурс]. – Режим доступа : <http://www.standards.ru/document/3628811.aspx>
3. Маглинец Ю. А. Анализ требований к автоматизированным информационным системам / Ю. А. Маглинец. – Бином, 2008.
4. Гурвиц Г. А. Microsoft Access 2010. Разработка приложений на реальном примере / Г. А. Гурвиц. – СПб. : БХВ-Петербург, 2010. – 496 с.
5. Грошев А. С. Информатика : учебник для вузов / А. С. Грошев. – Архангельск : АГТУ, 2010. – 470 с.
6. Грошев А. С. Базы данных: учеб.пособие / А. С. Грошев. – Архангельск : АГТУ, 2005. – 111 с.

Учебное издание

Галиев Карим Сулейманович
Печурина Елена Каримовна

Базы данных и СУБД

Учебно-методическое пособие

Подписано в печать _____.____.2016. Формат $60 \times 84^{1/16}$.

Усл. печ. л. – 4,75. Уч.-изд. л. – 3,45.

Тираж 100 экз. Заказ № _____

Типография Кубанского государственного
аграрного университета.

350044, г. Краснодар, ул. Калинина, 13