

СОДЕРЖАНИЕ

<i>Задание 1. Создание таблиц в режиме ввода данных</i>	<i>3</i>
<i>Задание 2. Создание таблиц в режиме Конструктора</i>	<i>5</i>
<i>Задание 3. Использование условий на значения</i>	<i>6</i>
<i>Задание 4. Использование масок ввода</i>	<i>8</i>
<i>Задание 5. Создание форм</i>	<i>11</i>
<i>Задание 6. Создание диаграмм и кнопочных форм</i>	<i>15</i>
<i>Задание 7. Создание запросов на выборку</i>	<i>16</i>
<i>Задание 8. Вычисления в запросах</i>	<i>18</i>
<i>Задание 9. Создание отчетов</i>	<i>21</i>
<i>Задание 10. Связывание таблиц</i>	<i>23</i>
<i>Задание 11. Создание составных форм, запросов, отчетов</i>	<i>25</i>
<i>Задание 12. Создание макросов</i>	<i>28</i>

ЗАДАНИЯ К ЛАБОРАТОРНЫМ РАБОТАМ

Правила и порядок выполнения лабораторных работ

Для выполнения лабораторных работ студент должен:


- *иметь представление* о принципах построения реляционных баз данных, этапах создания и возможностях их эксплуатации;
- *знать* основные возможности системы управления базами данных MS Access;
- *уметь* использовать реляционные базы данных в локальных и глобальных сетях.

Лабораторные работы выполняются в течение семестра в соответствии с методическими рекомендациями и представляются преподавателю для защиты в электронном виде в соответствии с графиком работ.

Файлы, созданные в процессе выполнения лабораторных работ и не защищенные на текущем занятии, сохраняются в электронном виде на электронном носителе студента.

Задание 1. Создание таблиц в режиме ввода данных

Цель: первоначальное знакомство с СУБД MS Access; изучение способа создания таблиц в режиме ввода данных (режиме таблицы); ввод и редактирование данных в таблицах.

- 1. Создать** на доступном диске личную папку.
- 2. Запустить СУБД MS Access.**
- 3. Создать файл базы данных.**
 - 3.1. Перейти в режим *Создания новой базы данных*.
 - 3.2. Сохранить файл, определив в качестве имени файла **База№1**, а в качестве ее местонахождения задать личную папку.
- 4. Выбрать вкладку *Таблицы* и **создать новую таблицу** в режиме ввода данных.**
- 4.1. Задать названия полей (*Имя, Рост, Вес*), как показано на рис. 1.1.
- 4.2. Закончить создание таблицы и закрыть ее (). Сохранить таблицу в активной базе данных под именем **Личности**.
- 5. Открыть таблицу *Личности*** для работы с ней.
 - 5.1. Ввести в таблицу не менее шести осмысленных записей (рис. 1.2).
 - 5.2. Научиться перемещаться по пространству таблицы (по строкам-записям и по столбцам – полям), пользуясь органами управления окна таблицы.
 - 5.3. Сохранить введенные данные и закрыть таблицу.

Имя	Рост	Вес	Поле4	Поле5
	2			

Рис. 1.1

Имя	Рост	Вес
Нина	160	70
Вера	185	80
Лена	185	55
Зина	150	65
Нюра	177	85
Лена	180	75

Рис. 1.2

6. Изменить структуру таблицы *Личности*, добавив новое поле и переместив имеющиеся поля *Рост* и *Вес*. В результате выполнения этого пункта таблица должна выглядеть так, как представлено на рис. 1.3.

6.1. Открыть базу данных *База№1*.

6.2. Открыть таблицу *Личности*.

6.3. Добавить перед полем *Имя* новое поле *ЛичнКод*.

6.4. Заполнить поле *ЛичнКод* текстовыми данными, как показано на рис. 1.3.

6.5. Поменять местами поля *Вес* и *Рост*.

6.6. Сохранить введенные данные и закрыть таблицу.

ЛичнКод	Имя	Вес	Рост
Л-001	Нина	70	160
Л-002	Вера	80	185
Л-003	Лена	55	165
Л-004	Зина	65	150
Л-005	Нюра	85	177
Л-006	Лена	75	180
			0

Рис. 1.3

ЛичнКод	Имя	Вес	Рост
Л-001	Нина	70	160
Л-002	Вера	80	185
Л-003	Лена	55	165
Л-004	Зина	65	150
Л-005	Нюра	85	177
Л-006	Лена	75	180
		0	0

Рис. 1.4

7. Изменить внешний вид таблицы. Пример выполнения представлен на рис. 1.4.

7.1. Уменьшить ширину полей *ЛичнКод*, *Имя*, *Вес*, *Рост* следующим образом:

- поле *Имя* – по ширине данных;
- поля *Вес*, *Рост* – до ширины 4,5 символа;
- поле *ЛичнКод* – при помощи мыши («на глазок»), как представлено на рис. 1.4.

7.2. Установить для таблицы новый тип и размер шрифта. При необходимости изменить высоту строк, содержащих записи таблицы.

7.3. Скрыть (сделать неотображаемыми в таблице) поля *ЛичнКод* и *Вес*.

7.4. Восстановить (вновь сделать видимыми) поля *ЛичнКод* и *Вес*.

8. Отредактировать содержимое таблицы *Личности*.

8.1. Используя кнопку «Найти», перейти к полю *Рост*, содержащему число «185».

- 8.2. Выделить последние две цифры и скопировать их в буфер обмена.
- 8.3. Перейти на поле *Вес* той же записи и вставить их, заменив предыдущее значение.
- 8.4. Выделить содержимое полей *Рост* и *Вес* третьей записи.
- 8.5. Скопировать содержимое этих полей в буфер обмена и вставить вместо соответствующих данных пятой записи.
- 8.6. Выделить пятую запись и поместить (вырезать) ее в буфер обмена.
- 8.7. Вставить содержимое буфера обмена в конец таблицы, как новую запись.
- 8.8. Закрыть файл базы данных.
- 9. Открыть базу данных *База№1***, найдя ее в соответствующей папке.
- 9.1. Открыть окно «Файл» - «Сведения» - «Свойства базы данных».
- 9.2. На вкладке *Документ* заполнить поля *Тема* и *Автор*, введя текст «БД для 1-го занятия» и вашу фамилию с инициалами.
- 10. Сохранить базу данных *База№1*** на диске для дальнейшего использования.

Задание 2. Создание таблиц в режиме Конструктора

Цель: изучение способов формирования структуры таблиц в режиме «Конструктора таблиц».

1. Запустить СУБД MS Access. Открыть базу данных *База№1*.

1.1. Просмотреть содержимое вкладки *Таблицы*. Убедиться, что в базе данных *База№1* содержится лишь одна таблица *Личности*.

2. Создать новую таблицу *Адреса*, пользуясь *Конструктором таблиц*.

2.1. Прототип этой таблицы, в котором представлены имена и типы полей, приведен на рис. 2.1.

Имя поля	Тип данных	Описание
ЛичнКод	Текстовый	Личный код гражданина (Текст 5)
ПочтКод	Текстовый	Почтовый индекс (Текст 6)
Город	Текстовый	Название города (Текст 20)
Улица	Текстовый	Название улицы (Текст 20)
Дом	Текстовый	Номер дома (Текст 8)
Корпус	Текстовый	Корпус (Текст 3)
Квартира	Текстовый	Номер квартиры (Текст 5)

Свойства поля

Рис. 2.1

2.2.1. При создании таблицы *Адреса* обязательно заполнять раздел *Описание*.

2.2.2. В разделе *Свойства поля* вводить только значения параметра «Размер поля» в соответствии с размерами полей, указанными в описании

поля в скобках, например, для поля *Улица* тип поля – *текстовый*, а размер поля – 20 символов.

2.3. Сохранить созданную таблицу в составе базы данных *База№1* под именем *Адреса*, не задавая ключевых полей.

2.4. Заполнить таблицу *Адреса*. Поле *ЛичнКод* заполнять в соответствии со значениями поля *ЛичнКод* таблицы *Личности*.

3. Создать новую таблицу *Предметы*, пользуясь *Конструктором таблиц*.

3.1. Таблица *Предметы* должна иметь структуру согласно табл. 2.1.

Таблица 2.1

Имя поля	Тип поля	Размер поля	Описание поля
<i>КодПред</i>	Текст	5	Код предмета
<i>Наименов</i>	Текст	50	Наименование предмета
<i>Часы</i>	Числа	байт	Объем (в часах)
<i>Контроль</i>	Текст	3	Вид контроля: Экз или Зач

3.2. Заполнить таблицу *Предметы*, введя информацию о пяти дисциплинах. В качестве Кода предмета вводить значения П-001, П-002 и т. д.

4. Открыть при помощи *Конструктора таблиц* таблицу *Личности*.

4.1. Просмотреть *типы* и *размеры* всех полей таблицы *Личности*.

4.2. Добавить в таблицу *Личности* новое поле *Рожд*, имеющее тип *Дата/время* с кратким форматом даты (10.12.75)

4.3. Изменить типы и размеры для некоторых полей в таблице *Личности* (табл. 2.2).

Таблица 2.2

Имя поля	Тип поля	Размер поля
<i>ЛичнКод</i>	Текстовое	5 символов
<i>Имя</i>	Текстовое	25 символов
<i>Рост</i>	Числовое	Байт
<i>Вес</i>	Числовое	Байт

4.4. Заполнить поле *Рожд* и обратить внимание на отображения дат при вводе месяца текстом (например: 5 мая 1980).

4.5. Сохранить и закрыть таблицы *Личности* и *Адреса*.

Задание 3. Использование условий на значения

Цель: проверка вводимых данных с использованием *Условий на значения*.

1. Открыть базу данных *База№1*.

1.1. Просмотреть содержимое вкладки *Таблицы*. Убедиться, что в базе данных *База№1* содержатся таблицы *Личности*, *Адреса*, *Предметы*.

2. Создать таблицу *Экзамены*, пользуясь *Конструктором таблиц*.

2.1. Создать поля: Код Предмета, Код Студента, Балл.

2.2. Для поля *Код Предмета*:

- установить тип поля *Текстовый*;
- установить размер поля в 5 символов;

2.3. Для поля *Код Студента*:

- установить тип поля *Текстовый*;
- установить размер поля в 5 символов.

2.4. Для поля *Балл*:

- установить тип поля *Числовой*;
- установить размер поля *Байт*;
- изменить подпись поля на «*Оценка*»;
- установить значение по умолчанию равным 0;
- установить такое условие на значение поля *Балл*, чтобы возможными значениями этого поля были: 0, 2, 3, 4, 5 (оценка «0» соответствует случаю, когда студент не сдавал экзамен, например, не был допущен к нему). Условие записать в виде логического выражения, связывающего допустимые значения оценок;

- задать в качестве сообщения об ошибке следующий текст: «Недопустимое Значение в поле "Балл"!!!»

2.5. Сохранить произведенные изменения в структуре таблицы *Экзамены* и закрыть ее.

2.6. В таблицу *Экзамены* ввести данные об экзаменах и зачетах по 5 дисциплинам за 3 семестра (некоторые дисциплины будут повторяться). При заполнении вводить в поле *Код студента* данные из *Личного кода* таблицы *Личности*, а в поле *Код предмета* данные из поля *Код предмета* таблицы *Предметы*.

3. Просмотреть все созданные ранее таблицы, пользуясь Конструктором таблиц, и проверить выполнение для этих таблиц следующих требований.

3.1. Все поля, несущие кодовые данные (например, поле *ЛичнКод*, *Код предмета* и т. п.), должны иметь тип *Текстовый* и размер 5 символов.

3.2. Для полей *Рост* и *Вес* таблицы *Личности* должны выполняться следующие требования:

- а) поля должны иметь тип *Числовой*, размер *байт*;
- б) поля должны иметь *Значения по умолчанию* **233** (для поля *Рост*) и **244** (для поля *Вес*);

- в) *Условия на значения* поля *Рост* должны допускать ввод значений роста в сантиметрах, лежащих в диапазоне от **100** до **210** и включающих значение по умолчанию **233**;

- г) *Условия на значения* поля *Вес* должны допускать ввод значений веса в килограммах, лежащих в диапазоне от **40** до **220** и включающих значение по умолчанию **244**.

3.3. Для поля *Рожд* сформировать и записать в соответствующих позициях раздела *Свойства поля* *Значение по умолчанию* и *Условие на значение*, которые позволят выполнить следующие требования:

- поле *Рожд* должно допускать ввод дат рождения лиц (студентов). При этом считается, что студентами дневной формы обучения могут быть лица не моложе 10 и не старше 80 лет;

- значением по умолчанию, свидетельствующим об отсутствии данных, должно быть значение «1 января 1111 года» (впоследствии обратите внимание на отображение данных, относящихся к XII в.).

Пример заполнения свойств поля для поля *Рожд* (без Условия на значение) представлен на рис. 3.1.

3.4. Все поля таблицы *Адреса* за исключением поля *Корпус* должны быть *Обязательными полями*.

4. Сохранить созданные таблицы в базе данных *База№1* и закрыть базу данных.

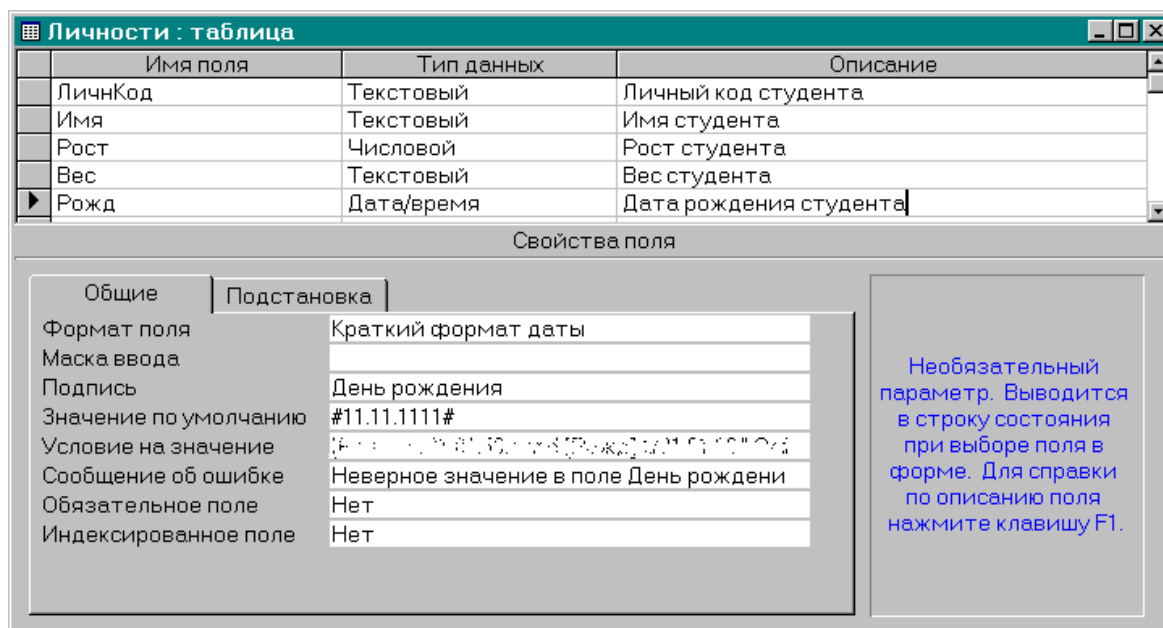


Рис. 3.1

Задание 4. Использование масок ввода

Цель: знакомство с использованием Масок ввода для повышения достоверности вводимых данных.

1. Открыть базу данных *База№1*.

1.1. Проверить наличие и убедиться в сохранности и работоспособности всех созданных ранее таблиц данной базы.

1.2. Продолжить модификацию таблиц базы данных, начатую в предыдущем задании, обеспечив выполнение для перечисленных ниже полей следующих требований.

Таблица *Адреса*

1.2.1. Для поля *ЛичнКод* должна быть определена маска ввода, обеспечивающая выполнение следующих требований:

- количество символов – 5;

- первый символ – **буква**;
- второй символ – **тире**;
- три последних символа – **цифры**;
- при вводе все символы должны преобразовываться в **прописные**;
- маска должна содержать вторую секцию «;**0**» ;
- маска должна содержать третью секцию «*».

1.2.2. Для поля *ПочтКод* должна быть установлена маска ввода, требующая обязательного ввода шести цифр без знака, пробелы не допускаются (поле *ПочтКод* является текстовым, но вводится в него должны только **цифровые символы**).

1.2.3. Для поля *Город* должна быть задана такая маска, чтобы при вводе символы автоматически преобразовывались в **прописные**.

1.2.4. Повторить п. 1.2.3 для поля *Улица*, используя не маску ввода, а **формат поля**.

1.2.5. Создать новое поле *Телефон*, в котором будет храниться семизначный телефонный номер (с трехзначным междугородным кодом) студента, по образцу **(812) 234-5678**.

1.2.6. Для поля *Телефон* после задания маски ввода определить и записать в соответствующей позиции *Свойств поля* минимально необходимый размер текстового поля.

Таблица Личности

1.2.7. Создать новое поле *Прим*, предназначенное для хранения произвольного текста примечания (например, автобиографии произвольного размера), выбрав для него подходящий тип поля. Для поля *Прим* должны быть заданы *Подпись* и *Описание поля*.

1.2.8. Создать новое поле *Фамилия*, удовлетворяющее следующим требованиям:

- тип – текстовый;
- размер – 25 символов;
- маска – автоматически вводит прописные буквы.

1.2.9. Создать новое поле *Пол*, удовлетворяющее следующим требованиям:

- тип – текстовый;
- размер – 1 символ;
- маска – автоматически вводит прописные буквы;
- условие на значение – допускает только **М** и **Ж** (русские) и **M** и **F** (английские).

1.2.10. Создать новое поле *Паспорт*, удовлетворяющее следующим требованиям:

- тип – текстовый;
- размер – 15 символов;
- маска – позволяет ввести запись, содержащую серию и номер паспорта, разделенные пробелами, по образцу **40 01 123456**.

1.2.11. Все созданные поля должны иметь заполненные Описания и Подписи.

Таблица *Предметы*

1.2.12. Для поля *КодПредм* должна быть определена маска ввода, аналогичная маскам для других «кодовых» полей.

1.2.13. Для поля *Часы*:

- должно быть определено Условие на значение, которое позволит вводить только целые числа, входящие в диапазон от **0** до **160 час**;
- в случае ввода неверных данных должно появляться сообщение «**Неверные данные в поле "Часы"**».

1.2.14. Для поля *Контроль* должны быть определены:


- маска ввода, позволяющая ввести только три буквы и преобразовать их к регистру прописных символов;
- условие на значение, которое позволит вводить только буквосочетания **ЭКЗ** или **ЗАЧ**.

1.3. Сохранить измененные таблицы *Личности*, *Адреса* и *Предметы* под старыми именами в составе базы данных *База№1*.

1.4. Осуществить ввод в каждую из таблиц по одной новой записи и проверить работоспособность всех заданных инструментов. Проверить корректность введенных данных и в случае необходимости отредактировать их. Убедиться в том, что все используемые маски ввода используются правильно, для чего попытаться ввести в поля заведомо некорректные данные.

1.5. Закрыть все таблицы.

2. Открыть окно «Схема данных» (Меню «Сервис» – «Схема данных»).

2.1. Активизировать режим *Добавление таблицы* () и добавить в схему данных вновь созданные таблицы. Схема данных должна выглядеть так, как показано на рис. 4.1.

2.2. Просмотреть схему данных, состоящую из четырех несвязанных таблиц *Личности*, *Адреса*, *Предметы* и *Экзамены*, которые входят в состав базы данных *База№1*. Установить размеры индивидуальных окон для каждой таблицы и для всей схемы данных так, как показано на рис. 4.1.

2.3. Сохранить созданную структуру и закрыть окно *Схема данных*.

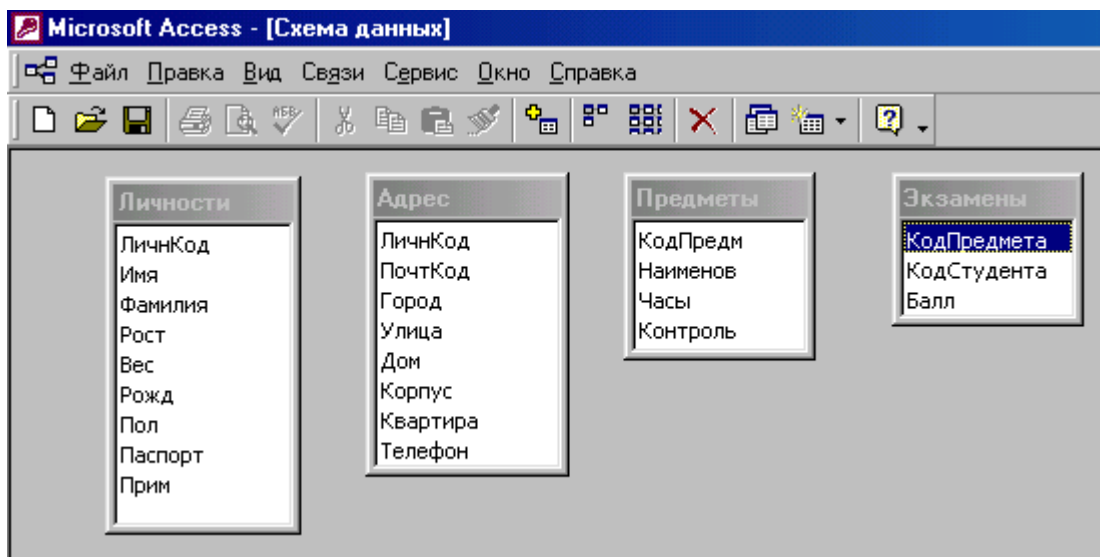


Рис. 4.1

3. Закрывать все таблицы базы данных *База№1*.

4. Продемонстрировать преподавателю схему данных и корректно заполненные таблицы.

5. Сохранить базу данных на диске для дальнейшего использования.

Задание 5. Создание форм

Цель: изучение технологий создания и изменения простых форм с использованием «Мастера форм», «Конструктора форм» и «Автоформ».

1. Открыть базу данных *База№1*.

1.1. Проверить наличие и убедиться в сохранности и работоспособности всех созданных ранее таблиц данной базы.

2. Создать форму *ФормаЛичности*.

2.1. На вкладке «*Формы*» выбрать *Мастер форм* и определить таблицу *Личности* в качестве источника.

2.2. Выбрать все поля из таблицы *Личности* для представления в создаваемой форме.

2.3. Выбрать внешний вид формы «*В один столбец*».

2.5. Задать в качестве имени формы «*ФормаЛичности*», а для дальнейшей работы определить режим *Изменение макета формы*.

2.6. Перейти в режим *Конструктора форм* и выполнить следующие действия.

2.6.1. Открыть области для заголовка формы и примечания формы. Разместить в них текст, как показано на рис. 5.1.

2.6.2. Поместить в области примечания текущую дату и время. Для этого воспользоваться меню «*Колонтитулы*» – «*Дата и время*».

2.6.3. Создать в заголовке формы вычисляемое поле, отображающее имя и фамилию:

а) пользуясь *панелью элементов управления*, разместить в области заголовка новое поле;

б) выбрать команду *Свойства* в контекстном меню и записать в поле *Данные* соответствующую формулу, пользуясь построителем выражений (...).

Обратите внимание, что имена полей всегда берутся в квадратные скобки, а текстовые данные в кавычки. Для соединения полей между собой можно использовать оператор «&» или «+».

Заголовок формы Личности		АННА ИВАНОВА	
Личкод	K_001	Прим	Анна Иванова-отличница
Имя	АННА		
Фамилия	ИВАНОВА		
Родд	01.01.78		
Рост	165		
Вес	65		
Пол	Ж		
Паспорт	40 08 787878		

Рис. 5.1

2.6.4. Создать поле со списком *Пол*:

а) удалить из формы поле *Пол*;

б) на **панели элементов управления** выбрать *Поле со списком* и разместить его в области данных;

в) в окне мастера *Создание полей со списком* необходимо проделать следующие действия:

- активизировать переключатель «*Будет введен фиксированный набор значений*»;

- заполнить **Столбец1** значениями, которые может принимать поле *Пол*: М, Ж, М, F;

- сохранить значение, которое содержит поле со списком в поле *Пол*;

- задать подпись поля «Пол».

2.6.5. Поместить в форму рисунок (рис. 5.1):

а) при помощи любого доступного графического редактора (например, **MS Paint**) создать растровый рисунок, сохранив его в рабочей папке;

б) на панели инструментов **Панель элементов** выбрать элемент управления *Рисунок*;

в) в открывшемся диалоговом окне выбрать созданный ранее файл рисунка и корректно разместить его в рамке, выбрав пункт *Свойства* в контекстном меню и определив вариант размещения (например, *По размеру рамки*).

3. Сохранить форму. Перейти в *режим работы с формой*.

4. Ввести в таблицу *Личности* две строки реальных данных, пользуясь созданной формой. Проверить, как работают созданные для таблиц *Условия на значения* и *Маски ввода*.

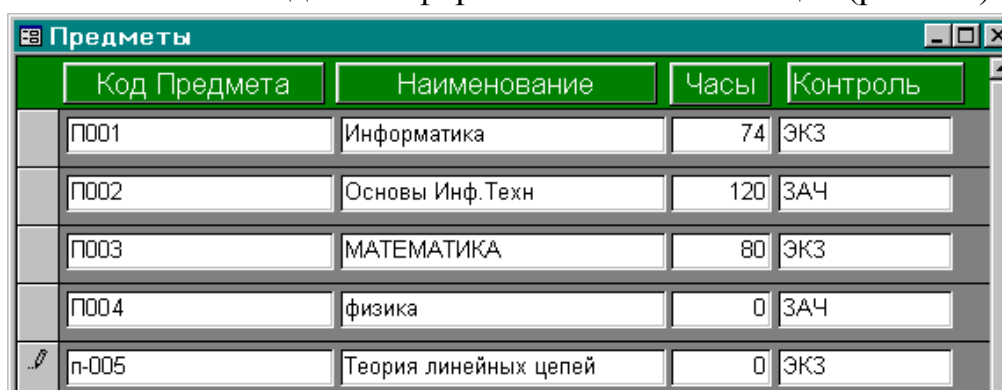
5. Создать форму *ФормаПредметы*.

5.1. В окне *Новая форма* указать использование режима *Мастера форм* и определить таблицу *Предметы* в качестве источника.

5.2. Выбрать из доступных полей таблицы *Предметы* все поля и в следующем диалоге выбрать *Ленточный* вариант размещения полей.

5.3. Задать для создаваемой формы имя ***ФормаПредметы*** и указать последующий переход в режим ввода данных.

5.4. Изменить в созданной форме заголовки столбцов (рис. 5.2).



Код Предмета	Наименование	Часы	Контроль
П001	Информатика	74	ЭКЗ
П002	Основы Инф.Техн	120	ЗАЧ
П003	МАТЕМАТИКА	80	ЭКЗ
П004	физика	0	ЗАЧ
п-005	Теория линейных цепей	0	ЭКЗ

Рис. 5.2

5.5. Закончить создание формы, проверить ее внешний вид, перейдя в режим представления *Формы*, сохранить ее как один из элементов базы данных ***База№1***.

6. Воспользоваться созданной формой и ввести в таблицу *Предметы* три новые записи, соответствующие трем реальным учебным дисциплинам. При вводе проверить, как работают *Условия на значения* и *Маски ввода*.

7. Создать форму *ФормаАдреса*.

7.1. Начать создание новой формы в режиме *Конструктора форм*.

7.2. Нажать кнопку «Добавить поля», выбрать таблицу *Адреса*, и пользуясь технологией «Drag and Drop» перенести в область данных формы все поля таблицы *Адреса*.

7.3. Сделать доступными в форме области для создания заголовка и примечания. Увеличить их размер и разместить в этих областях тексты заголовка и примечания, как это сделано на рис. 5.3.

Рис. 5.3

7.4. Пользуясь элементами управления на вкладках «Формат» и «Упорядочить» оформить поля таблицы и подписи, как показано на рис. 5.3.

7.5. Завершить создание формы и дать ей имя **ФормаАдреса**.

8. Ввести в таблицу Адреса две строки реальных данных, пользуясь созданной формой. Проверить, как работают созданные для таблиц *Условия на значения* и *Маски ввода*.

9. Создать форму ФормаЭкзамены по образцу рис. 5.4.

Код Предмета	Код студента	Оценка	Дата Экзамена:
		2	
П-001	Л-001	2	12.12.96
П-002	Л-001	3	
П-003	Л-001	4	

Рис. 5.4

9.1. Выделить мышкой имя таблицы **Экзамены** на вкладке «Таблицы».

9.2. В меню «Формы» выбрать тип формы «*Несколько элементов*».

9.3. После автоматического создания формы все необходимые корректировки (оформление заголовка и примечаний, изменение цветов и т.п.) выполнить при помощи *Конструктора форм*.

9.4. Для создания поля с текущей датой в соответствии с рис. 5.4 необходимо применить к полю пользовательский формат, используя символы «d, m, y».

9.5. Сохранить созданную форму под именем **ФормаЭкзамены** и проверить ее работоспособность, введя в таблицу **Экзамены** 5 записей об экзаменах конкретных студентов по конкретным предметам. Проверить, как работают созданные для таблиц *Условия на значения* и *Маски ввода*.

9.6. Внести коррективы в структуру таблицы *Экзамены*, поместив в нее новое поле с данными о дате сдачи экзамена или зачета – *ДатаЭкз.*

9.7. Откорректировать форму *ФормаЭкзамены* так, чтобы она содержала соответствующее новое поле, и выглядела так, как показано на рис. 5.4.

9.8. Для изменения формы воспользоваться *Конструктором форм*.

9.9. Проверить правильность работы нового варианта формы, введя в таблицу *Экзамены* три новые записи.

10. Сохранить базу данных *База№1* со всеми ее таблицами и формами.

Задание 6. Создание диаграмм и кнопочных форм

Цель: изучение технологии создания диаграмм в формах и кнопочных (управляющих) форм.

1. Открыть базу данных *База№1*.

1.1. Проверить работоспособность всех созданных ранее таблиц.

2. Создать круговую диаграмму, показывающую распределение предметов по часам (рис. 6.1).

2.1. Открыть вкладку форм, выбрать тип формы – *Сводная диаграмма* и в качестве источника указать таблицу *Предметы*.

2.2. Выбрать тип диаграммы *Круговая*.

2.3. Разместить в диаграмме поля *Наименование* и *Часы*.

2.4. Сохранить созданную форму под именем *График*.

2.5. Откорректировать диаграмму, добавив подписи рядов и название диаграммы, используя *Страницу свойств* (рис. 6.1).

Распределение предметов по часам



Рис. 6.1

3. Создать кнопочную форму по образцу, приведенному на рис. 6.2.

3.1. На вкладке *Работа с базами данных* открыть диалоговое окно *Диспетчера кнопочных форм*.

3.2. Нажав кнопку *Изменить*, ввести имя «**Кнопочная форма**» в поле *Название кнопочной формы* и перейти в режим *Создания*.

3.3. Ввести в поле *Текст* строку «Личности», выбрать в поле *Команда Открытие формы для изменения* и в качестве источника определить форму **Форма Личности**.

3.4. Повторить аналогичную последовательность действий для форм **Форма Адреса**, **Форма Предметы** и **Форма Экзамены**.

3.5. Добавить новую кнопку *Выход* для закрытия файла базы данных.

3.6. Откорректировать созданную кнопочную форму в режиме Конструктора форм в соответствии с рис. 6.2.

3.7. Сделать кнопочную форму стартовой, выбрав меню «Файл» - «Параметры» - Текущая база данных - Форма просмотра.

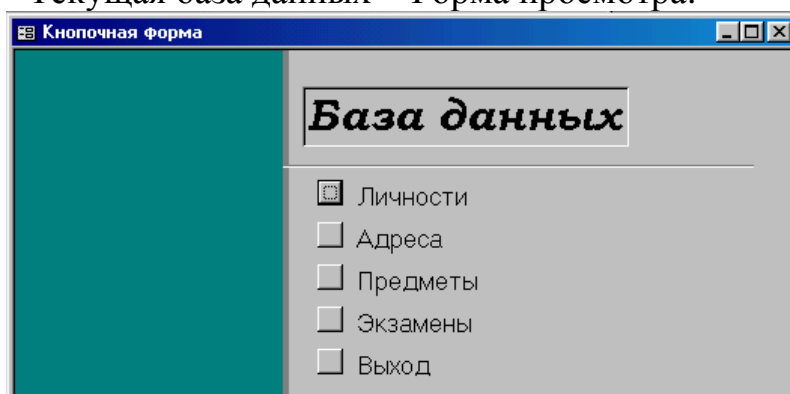


Рис. 6.2

4. Открыть таблицу *Switchboard Items* и проанализировать ее содержимое.

5. Сохранить базу данных *База№1* со всеми ее таблицами и формами.

Задание 7. Создание запросов на выборку

Цель: изучение техники построения запросов на выборку к одиночной таблице и запросов на создание таблицы.

1. Открыть базу данных *База№1*.

1.1. Проверить наличие и убедиться в сохранности и работоспособности всех созданных ранее таблиц.

2. Задать критерии выборки в запросе.

2.1. Открыть окно «Мастер запросов» и выбрать *Простой запрос*.

2.2. В качестве источника запроса указать таблицу *Личности*.

2.3. Выбрать поля *Фамилия*, *Пол*, *Рост*, *Вес* и указать *Подробный отчет*.

2.4. Перейти в режим изменения макета запроса.

2.5. Выполнить запрос и просмотреть его результат.

2.6. Пользуясь кнопкой *Режим*, расположенной на панели «Результаты», просмотреть созданный запрос в трех возможных формах его представления:

- в окне Конструктора запросов (*QBE - Query By Example*);
- в режиме таблицы;
- в режиме SQL (*Structured Query Language*).

2.7. В строке «Условие отбора» записать критерии выборки и последовательно выполнить следующие запросы о лицах с личным кодом:

- Л-003, Л-005, Л-007 (*Запрос1*);
- «большим», чем Л-007 (*Запрос2*);
- в интервале от Л-003 до Л-007 (*Запрос3*).

Каждый запрос должен быть сохранен отдельно под своим именем.

Примечание. Для выполнения запросов необходимо использовать логические операторы: >, <, >=, <=, <>, Between, AND, OR, NOT.

2.8. Сформулировать критерии выборки, создать и выполнить следующие запросы о лицах:

- с личным кодом Л-003, Л-005, Л-007 и весом, более 80 кг (*Запрос4*);
- с ростом более 170 см и личным кодом, «большим» чем Л-007 (*Запрос5*);
- женского пола (Ж и Ф) с личным кодом в интервале от Л-003 до Л-007 (*Запрос6*).

3. Осуществить выборку с последующей обработкой результатов.

3.1. Выполнить выборку с последующей сортировкой результатов.

3.1.1. Сформулировать критерии выборки, определить варианты сортировки результатов выборки, создать и выполнить следующие запросы:

- лиц с личным кодом в интервале от Л-003 до Л-007, отсортированными по убыванию (*Запрос_31_а*);
- лиц мужского пола (М-русск. и М-латин.) с личным кодом в интервале от Л-003 до Л-013, ростом более 180 см., весом в интервале от 70 до 120 кг. Результаты выборки должны быть отсортированы сначала в порядке возрастания значений роста, затем в порядке убывания значений веса (*Запрос_31_б*).

3.2. Выполнить выборку с сохранением результатов в новой таблице.

3.2.1. Выбрать режим *Создание таблицы* для размещения результатов запроса в новой таблице. Выполнить последовательно запросы п. 3.1.1 и сохранить их результат в новых таблицах *РезЗан311_а* и *РезЗан311_б*, которые будут размещены на вкладке *Таблицы* базы данных *База№1*.

3.3. Выполнить выборку с последующей группировкой результатов.

3.3.1. Осуществить выборку данных о лицах женского (Ж) и мужского (М) пола, включив в результат следующие поля *Пол, Фамилия, Рост, Вес*. Выполнить выборку, просмотреть и проверить результат.

3.3.2. Выполнить выборку по критериям п. 3.3.1 с группировкой результатов по полу. Над данными других полей необходимо выполнить следующие виды обработки:

- для поля *Фамилия* подсчитать **количество** лиц мужского и женского пола;
- для поля *Рост* определить **средний рост** мужчин и женщин;
- для поля *Вес* определить **наибольший и наименьший** вес мужчин и женщин.

Примечание. Для выполнения запросов п. 3.3.2 необходимо указать на необходимость выполнения операций группировки результатов (Σ) и определить групповые операции (определение количества **Count**, вычисление среднего **Avg**, нахождения наибольшего **Max** или наименьшего **Min** значений).

3.3.3. Сохранить запрос под именем *Запрос_333*. Сохранить результат запроса в таблице под именем *РезЗап333*. Результат выполнения запроса представлен на рис. 7.2.

Пол	Count_Фамил	Avg_Рост	Min_Вес	Max_Вес
Ж	5	185.6	56	90
М	3	182.666	55	65

Рис. 7.2

3.3.4. Выполнить п. 3.3.2 с группировкой по полу лиц, чей рост превышает **170** см (*Запрос_334*). Сравнить результаты, полученные в пп. 3.3.4 и 3.3.2.

4. Сохранить базу данных *База№1*.

Задание 8. Вычисления в запросах

Цель: знакомство с возможностями создания и использования вычисляемых полей в запросах.

1. Открыть базу данных *База№1*.

1.1. Проверить наличие и убедиться в сохранности и работоспособности всех созданных ранее таблиц данной базы.

2. Создать запрос *Measure*, позволяющий вывести фамилии людей, отсортированных по признаку пола, для которых в результирующей таблице будут представлены два значения роста – в сантиметрах и в футах, а также два значения веса – в килограммах и фунтах.

2.1. Создать в запросной *QBE*-форме два новых поля – для роста, выраженного в футах (ft), и веса – в фунтах (lb). В этих полях следует записать выражения, связывающие традиционные английские меры длины и веса с метрическими единицами (1 м = 3,28 ft; 1 кг = 2,2 lb).

2.2. Повторить выполнение запроса из п. 2.1, предварительно обеспечив выполнение следующих условий:

- вычисляемые поля с английскими единицами измерения должны располагаться рядом (справа) со своими метрическими аналогами;


- поля, представленные в результате выборки, должны иметь соответствующие подписи;
- количество десятичных знаков после запятой в числовых полях должно быть равно 2.

Для выполнения этих условий необходимо, находясь в запросной *QBE*-форме, активизировать диалоговое окно *Свойства* для конкретного поля и задать необходимые значения параметров поля (в том числе и подписи). Образец результата запроса представлен на рис. 8.1.

	Пол	Фамилия	Рост (См)	Рост (Ft)	Вес (кг)	Вес (Lb)
	Ж	Чурикова	165	5,41	56	123,20
	Ж	Иванова	180	5,90	90	198,00
	Ж	Петрова	170	5,58	85	187,00
	Ж	Мормонштейн	180	5,90	90	198,00
	Ж	Судакова	233	7,64	65	143,00
	М	Чумаков	123	4,03	35	77,00
	М	Иванов	190	6,23	120	264,00
	М	Гогоберидзе	170	5,58	120	264,00
			0		0	

Рис. 8.1

3. Создать запрос, в котором будут представлены сведения о дате рождения и возрасте студента, как показано на рис. 8.2.

3.1. Для выполнения задания необходимо записать выражение, вычитающее год даты рождения из года текущей даты. Для этого, в свою очередь, следует воспользоваться стандартными функциями **Date()** и **Year()**. Созданное выражение должно быть записано в соответствующем поле запросной *QBE*-формы при помощи *Построителя выражений* .

	Пол	Фамилия	Рост	Вес	Возраст	День рождения
	Ж	Чурикова	165	56	886	11.11.11
	Ж	Иванова	180	90	17	10.10.80
	Ж	Петрова	170	85	27	01.01.70
	Ж	Мормонштейн	180	90	20	10.10.77
	Ж	Судакова	233	65	32	20.12.65
	М	Чумаков	123	35	886	11.11.11
	М	Иванов	190	120	20	11.11.77
	М	Гогоберидзе	170	120	886	11.11.11
	М	Чудаков	170	80	32	11.11.65
	*		233	0		11.11.11

Рис. 8.2

3.2. Выполнить запрос, удостовериться в правильности его работы и сохранить запрос под именем **Возраст**.

4. Создать запрос, где при помощи *Построителя выражений* будет определено вычисляемое поле, в котором для каждого студента будут представлены обобщенные антропометрические данные.

4.1. Создать поле *Антроп_Коэфф*, содержащее обобщенный антропометрический коэффициент, вычисленный, как поделенная на 10 разность между ростом в сантиметрах и весом в килограммах.

4.2. Выполнить запрос, проверить правильность его работы и сохранить под именем *Антропо*.

5. Создать запрос *Антропо2*, в котором непосредственно перед его выполнением будут запрашиваться не определенные в *QBE*-форме параметры.

5.1. На основе сохраненного запроса *Антропо* создать запрос, позволяющий получить сведения о фамилии, имени, росте, весе и вычисляемом антропометрическом показателе для конкретного лица, имя которого будет запрошено как значение параметра перед выполнением запроса. Для этого вместо конкретного имени в строке *Условие отбора* в поле *Имя* введите текст: [**Введите имя студента:**].

5.2. Выполните запрос, введя в появившееся окно одно из имен.

5.3. Модифицируйте запрос, написав оператор **Like** перед параметром. Теперь запрос должен позволять вводить вместо букв имени символы шаблона * и ?.

5.4. Выполните запрос несколько раз, задавая следующие условия: все имена, в которых есть буква «а»; все имена, состоящие из 4 букв; все имена, в которых вторая буква «о».

6. Создать запрос на обновление таблицы.

6.1. Открыть новую запросную форму и определить в качестве источника таблицу *Предметы*.

6.2. Определить тип запроса *Запрос на обновление*.

6.3. В поле *Обновление* сформулировать выражение, удваивающее количество часов для всех предметов.

6.4. В поле *Условие отбора* ограничить выполнение этого запроса только теми предметами, видом контроля для которых является Зачет.

6.5. Сохранить запрос под именем *Часы*, выполнить и проверить правильность его работы.

7. Создать перекрестный запрос на основе таблицы *Личности*, иллюстрирующий распределение мужчин и женщин по годам рождения.

7.1. В *Мастере запросов* выбрать вариант *Перекрестный запрос*.

7.2. Выбрать в качестве заголовка строк поле *Пол*, в качестве заголовка столбцов поле *Рожд*, а в качестве значения поле *Фамилия* с функцией **Count** (подсчет значений).

7.3. Сохранить запрос под именем *Перекрестный запрос*.

7.4. Выполнить запрос и проверить правильность его работы.

7.5. В режиме *Конструктора* изменить этот запрос так, чтобы в заголовках столбцов выводились не даты, а года рождения.

8. Сохранить базу данных *База№1* со всеми ее таблицами, формами и запросами.

Задание 9. Создание отчетов

Цель: знакомство с созданием простых отчетов к одиночным таблицам.

1. Открыть базу данных *База№1*.

1.1. Проверить наличие и убедиться в сохранности и работоспособности всех созданных ранее таблиц данной базы.

1.2. Создать и выполнить запрос, позволяющий извлечь в отдельную таблицу *Девушки* (рис. 9.1) лиц женского пола в возрасте от 16 до 20 лет.

1.3. Создать и выполнить запрос, позволяющий извлечь в отдельную таблицу *Дедушки* лиц мужского пола в возрасте от 50 до 80 лет. Таблица должна содержать поля *Фамилия, Имя, Пол, Рожд.*

1.4. При отсутствии в таблице *Личности* подходящих данных, предварительно ввести 5 записей, удовлетворяющих критерию выборки.



	Фамилия	Имя	Пол	Возраст	Рост	Вес
▶	Сидорова	Лена	F	17	165	55
	Пятакова	Ираида	F	20	233	55
	Мормонштейн	Ира	Ж	20	180	90
	Иванова	Нюра	Ж	17	180	90
*						

Рис. 9.1

2. Создать отчет *Отчет Девушки*.

2.1. Активизировать *Конструктор отчетов* для создания отчета на основе таблицы *Девушки*, созданной в п. 1.3.

2.2. Пользуясь *Списком полей*, выбрать все необходимые поля.

2.3. Разместить выбранные поля и подписи к ним, а также заголовок и колонтитул отчета на пространстве листа так, как показано на рис. 9.2

2.4. Перейдя в режим просмотра отчета, убедиться в его сходстве с образцом.



Как много девочек хороших ...
(Отчет созданный на основе таблицы "Девушки")

Фамилия:	Имя:	Пол:	Возраст
Сидорова	Лена	F	17
Пятакова	Ираида	F	20
Мормонштейн	Ира	Ж	20
Иванова	Нюра	Ж	17

28-январь-97

Рис. 9.2

2.5. Поместить в примечании отчета итоговое поле, в котором рассчитывается средний возраст лиц, данные о которых приведены в отчете.

2.6. Сохранить отчет под именем *Отчет_Девушки*.

3. Создать отчет *Отчет_Дедушки*.

3.1. Создать отчет, представленный на рис. 9.3, в режиме *Мастера отчетов* и выполнить следующее:

- определить имя таблицы-источника *Дедушки*;
- выбрать все поля таблицы (*Фамилия, Имя, Пол, Рожд*);
- выбрать *режим группировки* на основе данных поля *Рожд*;
- определить интервал группировки по *годам рождения*;
- задать *сортировку* записей по алфавиту фамилий в пределах группы;
- выбрать в качестве варианта вида макета *Табличный*;
- задать имя отчета *Отчет_Дедушки*;
- определить для дальнейшей работы с отчетом режим *Изменения структуры отчета*.

Отчет "Дедушки"

<i>Рожд по годам</i>	<i>Фамилия</i>	<i>Имя</i>	<i>Пол</i>	<i>Рожд</i>
<u>Заголовок группы, организованной по</u> 1925 <u>году рождения</u>				
1925	Чумаков	Алан	М	12.12.25
<u>Заголовок группы, организованной по</u> 1930 <u>году рождения</u>				
1930	Гогоберидзе	Рустам	М	11.11.30
	Вунюков	Лавр	М	10.10.30
	Пятакович	Ариан	М	05.05.30
<u>Заголовок группы, организованной по</u> 1935 <u>году рождения</u>				
1935	Кузяков	Иероним	М	11.11.35
<u>Заголовок группы, организованной по</u> 1936 <u>году рождения</u>				
1936	Дураков	Матвей	М	11.11.36
	Залкинд	Феофан	М	19.09.36

Рис. 9.3

3.2. Для окончательного оформления отчета необходимо вернуться к *Конструктору отчетов* и выполнить следующие действия:

- создать заголовки отчета;
- создать подзаголовки групп;
- изменить названия столбцов отчета;
- изменить размеры полей;
- создать итоговое поле в примечании группы, подсчитывающее количество лиц в каждой группе.

Для того чтобы включить в подзаголовок цифру года рождения, необходимо воспользоваться *Построителем выражений*. На основе значения поля *Рожд* и функций преобразования дат создать выражение, выделяющее год из поля даты. Полученное выражение разместить в вычисляемом поле, соединяющем две текстовые строки и год рождения в подзаголовке группы.

4. Создать отчет *Отчет_Девушки2* на основе таблицы *Девушки*, который позволит получить:

- сведения о девушках в возрасте от 16 до 20 лет с указанием их фамилий, имен, возраста (не даты рождения), роста и веса;
- итоговые значения о количестве лиц в каждой возрастной группе;
- итоговые значения о среднем росте и весе всех лиц, представленных в отчете.

Данные в отчете должны быть сгруппированы по возрастным группам, в каждой из которых должны присутствовать только девушки одного года рождения.

Вторичная группировка в пределах возрастной группы должна быть выполнена по росту.

5. Сохранить базу данных на диске для дальнейшего использования.

Задание 10. Связывание таблиц

Цель: знакомство с использованием ключевых полей для связывания таблиц; изучение типов связей.

1. Открыть базу данных *База№1*.

1.1. Проверить наличие и убедиться в сохранности и работоспособности всех созданных ранее таблиц данной базы.

2. Создать систему первичных ключей для таблиц по следующему образцу (табл. 10.1)

Таблица 10.1

Таблица	Первичный ключ
<i>Личности</i>	<i>ЛичнКод</i>
<i>Адреса</i>	<i>ЛичнКод</i>
<i>Предметы</i>	<i>КодПредм</i>
<i>Экзамены</i>	<i>КодЭкзамена</i>

2.1. В тех полях таблиц, которые становятся ключевыми, удалить повторяющиеся записи.

2.2. В ранее созданных полях *ЛичнКод* и *КодПредм* для всех таблиц базы данных внимательно проверить совпадение масок, использовавшихся при вводе данных. Так, например, маска вида «>L\ -000;0» и маска «>L\ -000» сохраняют данные в разных видах – с сохранением в таблице символа «-» и без него соответственно. Очевидно, что это приведет к несовпадению данных, хранящихся в таблицах, использующих разные маски, и невозможности установить связи между этими таблицами.

2.3. В таблице *Экзамены* создать новое поле *КодЭкзамена*, которое будет уникально определять каждую запись таблицы и использоваться как *первичный ключ*, и установить для него тип поля *Счетчик*.

2.4. Открыть окно *Схема данных* и разместить таблицы так, как показано на рис. 10.1 (сначала без связей).

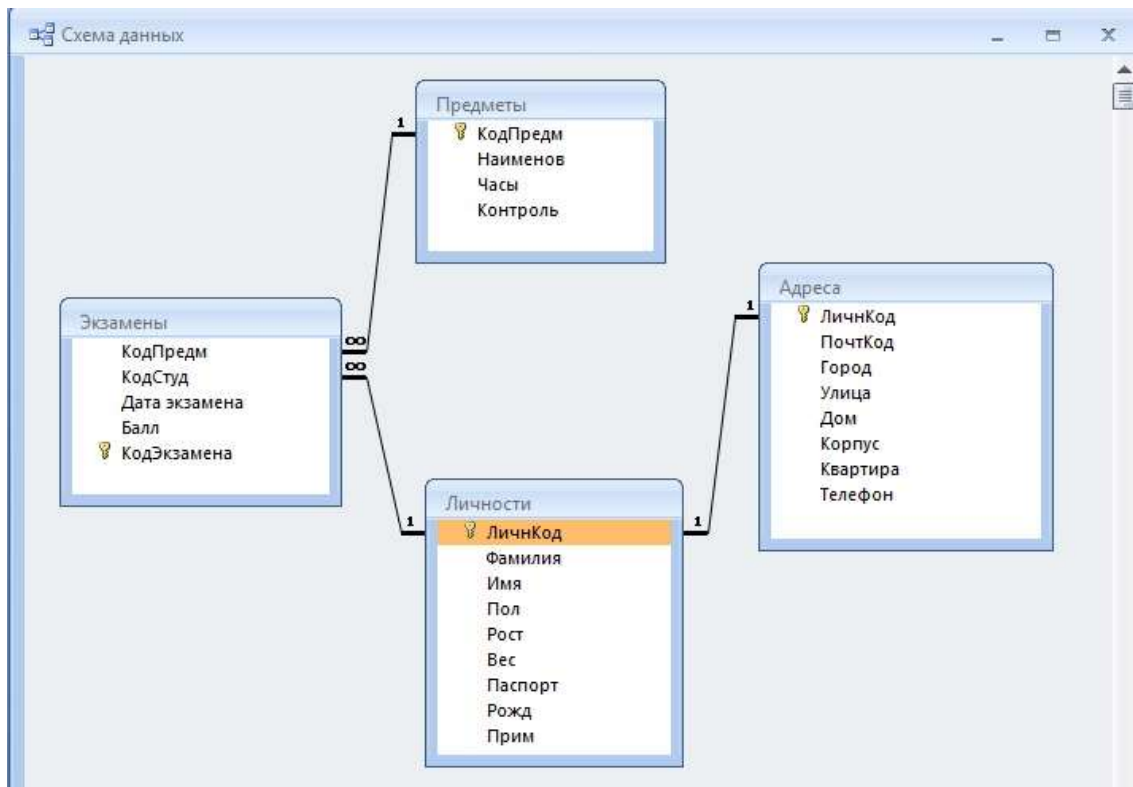


Рис. 10.1

3. Установить связи между таблицами *Личности*, *Адреса*, *Предметы*, *Экзамены*.

3.1. Внимательно прочитать все подсказки, комментарии и пояснения, касающиеся связывания таблиц и параметров объединения.

3.2. Выбрав таблицу *Экзамены*, при помощи мыши «перетащить» поле *КодПредмета* на поле *КодПредм* таблицы *Предметы*, указав тем самым необходимость установления связи между таблицами по этим полям.

3.3. В открывшемся окне *Связи* определить тип и параметры связи между таблицами *Экзамены* и *Предметы*.

3.3.1. Между полями *Предметы.КодПредм* и *Экзамены.КодПредм* должна быть установлена связь типа *Один-ко-Многим* (рис. 10.2).

3.3.2. Установить режим *Обеспечения целостности данных: Каскадное обновление связанных полей* и *Каскадное удаление связанных записей*. Для установки этих режимов необходимо активизировать соответствующие переключатели в группе *Обеспечение целостности* окна *Связи*.

3.4. Установить связь между таблицами *Личности* и *Экзамены* через поля *КодСтудента* и *ЛичнКод* соответственно. Тип отношения между этими таблицами – *Один-ко-Многим*, где стороной *Много* выступает таблица *Экзамены*. Обеспечить целостность данных при обновлении и удалении записей. Установить объединение таблиц по первому типу (когда из связанных таблиц выбираются только записи, для которых совпадают значения полей связи).

3.5. Установить связь типа *Один-к-Одному* между соответствующими полями таблиц *Личности* и *Адреса* с обеспечением целостности данных при обновлении и удалении записей.

3.6. Активизировать и просмотреть окно *Схема данных*. Его внешний вид должен соответствовать образцу, приведенному на рис. 10.1.

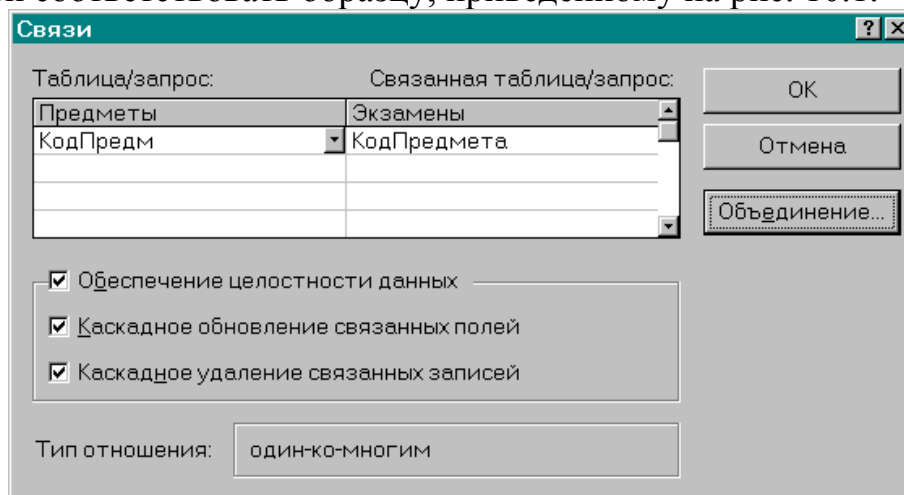


Рис. 10.2

4. Используя подтаблицы просмотреть связанные данные для таблицы *Личности* (вкладка «Главная» – «Записи» – «Дополнительно» – «Подтаблица»).

5. Сохранить все таблицы и базу данных *База№1*.

Задание 11. Создание составных форм, запросов, отчетов

Цель: изучение способов создания составных форм, запросов и отчетов для связанных таблиц.

Создание составных форм

1. Открыть базу данных *База№1*.

2. Создать составную форму (для двух связанных таблиц *Личности* и *Экзамены*).

2.1. На основе таблицы *Экзамены*, пользуясь режимом автоформы *Таблица*, создать подчиненную форму *Экзамены*, содержащую следующие сведения: Код студента, Код предмета, Балл, Дата экзамена, Код экзамена.

2.2. В качестве главной формы использовать форму *Личности*, содержащую все сведения о студентах.

2.3. Разместить подчиненную форму внутри главной, как показано на рис. 11.1. Для этого использовать режим *Конструктора* для главной формы и «перетащить» в нее предварительно созданную подчиненную форму из вкладки *Формы* окна базы данных (также можно использовать элемент управления *Подчиненная форма/отчет* на панели элементов). Для детальной подгонки всех элементов форм воспользоваться услугами *Конструктора форм*, последовательно улучшая общий вид составной формы.

Личности

Заголовок формы личности Анна Иванова

ЛичнКод: Л-001 Прим: Анна Иванова - отличница

Фамилия: Иванова

Имя: Анна


Рожд: 01.01.1978

Пол: Ж

Рост: 175

Вес: 80

Паспорт: 40 08 787878



Экзамены

КодЭкзамена	КодСтуд	Балл	ДатаЭкз	Наименов
1	Л-001	4	10.01.2007	Информатика
4	Л-001	5	15.01.2007	Основы Инф. Техн.
7	Л-001	5	20.01.2007	Математика
*	(Счетчик) Л-001	0		

Запись: 1 из 3

Рис. 11.1

2.4. Для отображения в Коде Предмета названия предмета внести коррективы в форму *Экзамены*. Для этого в Конструкторе формы удалить поле *КодПредм* и поместить на его место раскрывающийся список, использующий значения из поля *Наименов* таблицы *Предметы* и сохраняющий результат в поле *КодПредм* таблицы *Экзамены*. Форма должна принять вид в соответствии с рис. 11.1

2.5. Сохранить созданную форму под именем *Личности_Оценки*.

3. Откорректировать составную форму Личности, добавив в нее информацию об адресе студента.

3.1. В режиме Конструктора перетащить форму *Адреса* в форму *Личности* и разместить ее перед подчиненной формой *Экзамены*.

3.2. Общий вид созданной формы должен соответствовать рис. 11.2.

4. Воспользовавшись созданной составной формой ввести информацию о двух новых студентах (их личные данные, адреса и результаты сданных экзаменов).

Личности

Заголовок формы личности Анна Иванова

ЛичнКод: 1-001 Прим: Анна Иванова - отличница

Фамилия: Иванова

Имя: Анна

Рожд: 01.01.1978

Пол: Ж

Рост: 175

Вес: 80

Паспорт: 40 08 787878

Адреса

Заголовок поля для ввода данных

ЛичнКод: Л-001 Дом: 1

ПочтКод: 111111 Корпус: 1

Город: Москва Квартира: 1

Улица: Мира Телефон: (095)111-11-11

Примечание формы: Подпись:

Запись: 1 из 1

Экзамены

КодЭкзамена	КодСтуд	Балл	ДатаЭкз	Наименов
	1 Л-001	4	10.01.2007	Информатика
	4 Л-001	5	15.01.2007	Основы Инф. Техн.
	7 Л-001	5	20.01.2007	Математика
*	(Счетчик) Л-001	0		

Запись: 1 из 3

Рис. 11.2

Создание составных запросов

5. Создать запрос СтудОценки, при помощи которого будут извлекаться данные из связанных таблиц *Личности*, *Предметы* и *Экзамены*.

5.1. В Конструкторе выбрать поля: *Фамилия*, *Имя*, *Наименов*, *Балл* и *ДатаЭкз*. В качестве параметров запроса должны выступать фамилия студента и название предмета.

5.2. Убедиться, что запросные связи соответствуют структурным связям в Схеме данных.

5.3. Выполнить запрос и убедиться в правильности его работы.

Создание составных отчетов

6. Создать отчет для двух связанных таблиц *Предметы* и *Экзамены*.

6.1. Для создания отчета воспользоваться Мастером отчетов. На первом шаге последовательно из трех таблиц добавить поля: *Наименов*, *Часы*, *Контроль*, *Балл*, *Дата экзамена*, *Фамилия*.

Предметы

Наименов	Информатика		
Часы	74		
Контроль	ЭКЗ		
	Балл	ДатаЭкз	Фамилия
	4	10.01.2007	Иванова
	5	10.01.2007	Сидоров
	3	10.01.2007	Петров
Итоги для 'КодПредм' = П-001 (3 записей)			
Avg	4		
Наименов	Основы Инф. Техн.		
Часы	120		
Контроль	ЗАЧ		
	Балл	ДатаЭкз	Фамилия
	3	15.01.2007	Сидоров
	4	15.01.2007	Петров
	5	15.01.2007	Иванова
Итоги для 'КодПредм' = П-002 (3 записей)			
Avg	4		
Наименов	Математика		
Часы	80		
Контроль	ЭКЗ		
	Балл	ДатаЭкз	Фамилия
	4	20.01.2007	Сидоров
	4	20.01.2007	Петров
	5	20.01.2007	Иванова
Итоги для 'КодПредм' = П-003 (3 записей)			
Avg	4,333333333		

Рис. 11.3

- 6.2. В качестве главной таблицы определить таблицу *Предметы*.
- 6.3. Группировку и сортировку не задавать.
- 6.4. В качестве групповой операции (*Кнопка Итоги*) выбрать среднее по полю *Балл*.
- 6.5. Внешний вид отчета, включая заголовок и общее оформление, должен соответствовать приведенному на рис. 11.3 образцу.
- 6.6. Сохранить отчет под именем *Предметы_Оценки*.
7. Сохранить базу данных на диске для дальнейшего использования.

Задание 12. Создание макросов

Цель: Знакомство с созданием макросов и их использованием для автоматизации работы базы данных.

1. Открыть базу данных *База№1*
2. Создать макрос *Макрос1*, обеспечивающий выполнение следующих действий:
 - выдачу критического сообщения «НАЧАЛО РАБОТЫ МАКРОСА»;
 - выполнение запроса *СтудОценки*. В качестве режима просмотра выбрать режим *Таблица*;
 - выдачу сообщения «КОНЕЦ РАБОТЫ МАКРОСА».
3. Выполнить макрос *Макрос1*. Убедиться в правильности выполнения.

4. Установить с помощью кнопки на панели макросов режим пошагового выполнения и, выполнив макрос по шагам, пронаблюдать сообщения, выдаваемые на каждом шаге.

5. Создать макрос *Макрос2*.

5.1. Добавить в макрос *Макрос1*, после второй команды еще две команды:

- для выдачи информационного сообщения *ПАУЗА* после появления таблицы с результатом запроса;
- для закрытия окна с результатом выполнения запроса.

5.2. Сохранить макрос под новым именем *Макрос2*.

6. Выполнить «*Макрос2*». Убедиться в правильности выполнения.

7. Преобразовать макрос в модуль VBA.

7.1. Открыть *Макрос2* в режиме Конструктора.

7.2. Осуществить преобразование макроса *Макрос2* в модуль VBA, отметив все предлагаемые флажки.

7.3. Открыть преобразованный макрос в режиме Конструктора и проанализировать его структуру.

7.4. Запустить созданный модуль из режима Конструктора (клавиша F5).

8. Создать макрос *Макрос3*, выполняющий следующие действия:

- выдачу стартового сообщения;
- открытие формы *Адреса* в режиме изменения с фильтрацией записей только по Санкт-Петербургу;
- переход к элементу управления – полю *ПочтКод*, представленному в форме *Адреса*;
- экспорт содержимого таблицы *Личности* в формат html с автоматическим открытием созданного файла;
- подачу звукового сигнала;
- выдачу сообщения о завершении работы.

9. Создать командную кнопку в заголовке формы *Личности* с подписью «Адреса», нажатие на которые вызовет открытие соответствующей формы.

9.1. Создать макрос *Адреса*, обеспечивающий открытие соответствующей формы.

9.1.1. На вкладке *Макросы* перейти в режим создания макросов.

9.1.2. Перетащить из вкладки *Формы* окна базы данных форму *Адреса* в область *Макрокоманда* и проверить заполнение аргументов макрокоманды.

9.1.3. Сохранить макрос под именем *Адреса*.

9.1.4. Проверить работоспособность макроса.

9.2. Создать командную кнопку с надписью «Адреса», к которой присоединить соответствующий макрос.

9.2.1. Открыть форму *Личности* в режиме конструктора и активизировать в ней область заголовка.

9.2.2. Перетащить с вкладки *Макросы* окна базы данных макрос *Адреса* в область заголовка формы.

9.2.3. Проверить работоспособность кнопок.

10. Сохранить базу данных.