Учебное пособие
для лабораторных работ по предмету
«Базы данных»

Содержание

Урок 1. Создание таблиц базы данных 3

Урок 2. Ввод данных в таблицы 16

Урок 3. Логическая структура базы данных 23

Урок 4. Однотабличные формы 30

Урок 5. Формы для загрузки двух таблиц 38

Урок 6. Многотабличные формы 50

Урок 7. Запросы 71

Урок 8. Отчет по одной таблице 98

Урок 9. Отчеты по двум таблицам 106

Урок 10. Многотабличные отчеты 112

Урок 11. Разработка отчета на основе запроса 118

Урок 12. Управление приложением пользователя 126

Урок 1. Создание таблиц базы данных

Рассмотрим последовательность необходимых действий при создании файла и таблиц базы данных. База данных «Учебный процесс» будет состоять из семи таблиц, свойства полей которых приведены в конце этой главы в приложении А.

Для создания файлы базы данных выполним команду меню **Файл|Создать.** В окне **Создание** выберем **Новая база данных** (рис 1.1)**.**

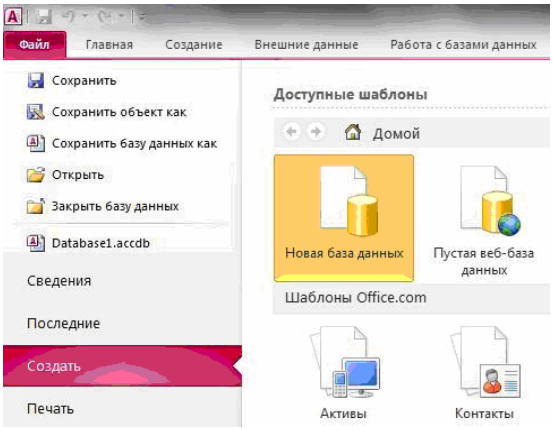


Рис. 1.1. Создание файла базы данных.

Именуем базу данных, определим папку, в которой будет размещен этот файл и щелкаем кнопкой мыши на значке **Создать** (рис 1.2).

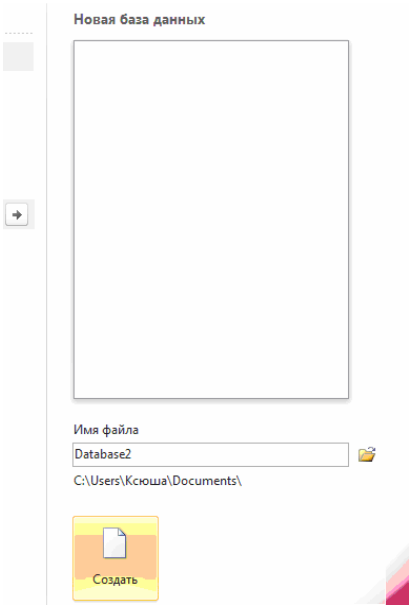


Рис.1.2. Именование базы данных.

Создание структуры таблицы ГРУППА

Начнем создание таблицы ГРУППА с определения ее структуры в режиме конструктора таблиц. Основные параметры структуры таблицы ГРУППА, представлены в табл. 1.1.

Таблица 1.1. Основные параметры структуры таблицы ГРУППА

Имя поля	**Ключевое поле**	**Уникальное**	**Обязательное поле**	**Тип данных**	**Размер**	**Число десятичных знаков**	**Подпись поля**	**Условие на значение** Сообщение об ошибке
НГ	Да	Да	Да	Текстовый	3		Ном. группы	
КОЛ			Нет	Числовой	Байт		Кол. ст. в группе	>=0 And <=35 Кол. студентов больше допустимого
ПБАЛЛ			Нет	Числовой	Одинарное с плавающей точкой(4 байта)	2	Прох. балл	>2 And <5 Or 0 Ошибка в оценке

В окне базы данных выберем вкладку **Создание** и нажмем кнопку **Конструктор таблиц** (рис. 1.3).

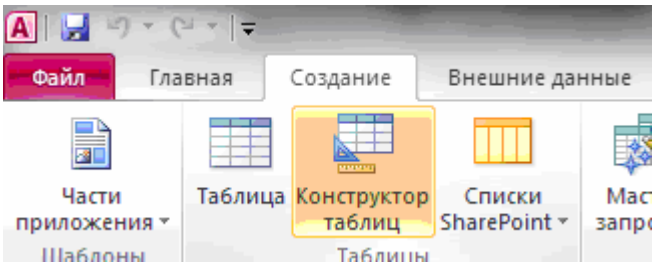


Рис 1.3. Вызов конструктора таблиц.

В соответствии с приведенными в табл. 3.1 проектными параметрами структуры для таблицы ГРУППА в окне конструктора **Таблица1:таблица** (рис. 1.4) нужно:

- В столбец **Имя поля** ввести в нужной последовательности имена полей НГ, КОЛ, ПБАЛЛ
- В столбце **Тип данных** выбрать нужный тип данных для каждого поля, использовав кнопку списка
- На вкладке **Общие** задать свойства полей (рис. 1.5):
    - **Размер поля**, нажав в нем кнопку списка:
    - для текстового поля НГ зададим размер поля 3
    - для числового поля КОЛ выберем значение **Байт**, определяющее длину целого числа, достаточную для размещения максимального значения поля (40 студентов в группе) (см. рис.1.6)
    - для числового поля ПБАЛЛ выберем **Одинарное с плавающей точкой**
    - **Формат поля** для поля ПБАЛЛ – **Фиксированный**
    - **Число десятичных знаков** для поля ПБАЛЛ – 2
    - **Подпись**, **Условие на значение**, **Сообщение об ошибке**– для каждого из полей выберем, как указано в табл. 3.1

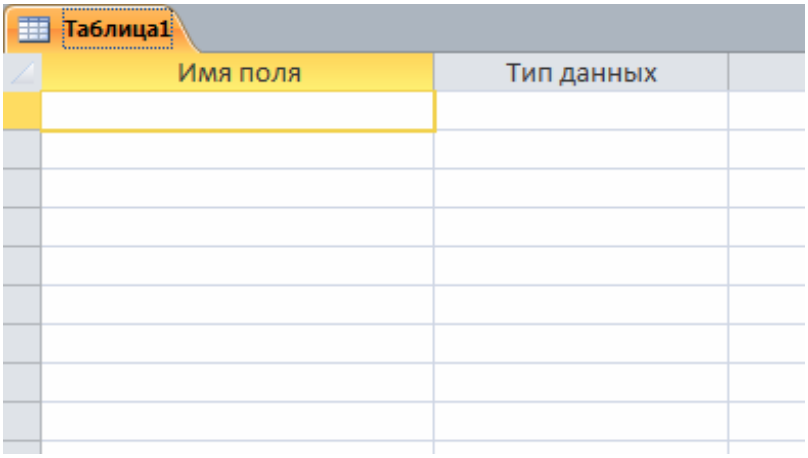


Рис 1.4. Окно конструктора таблиц.

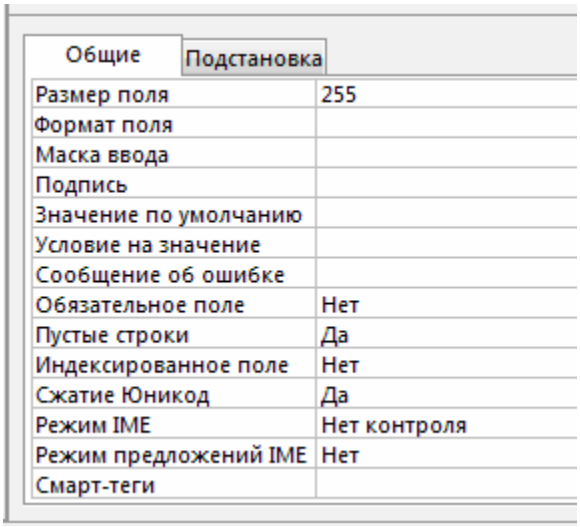


Рис. 1.5. Вкладка **Общие.**

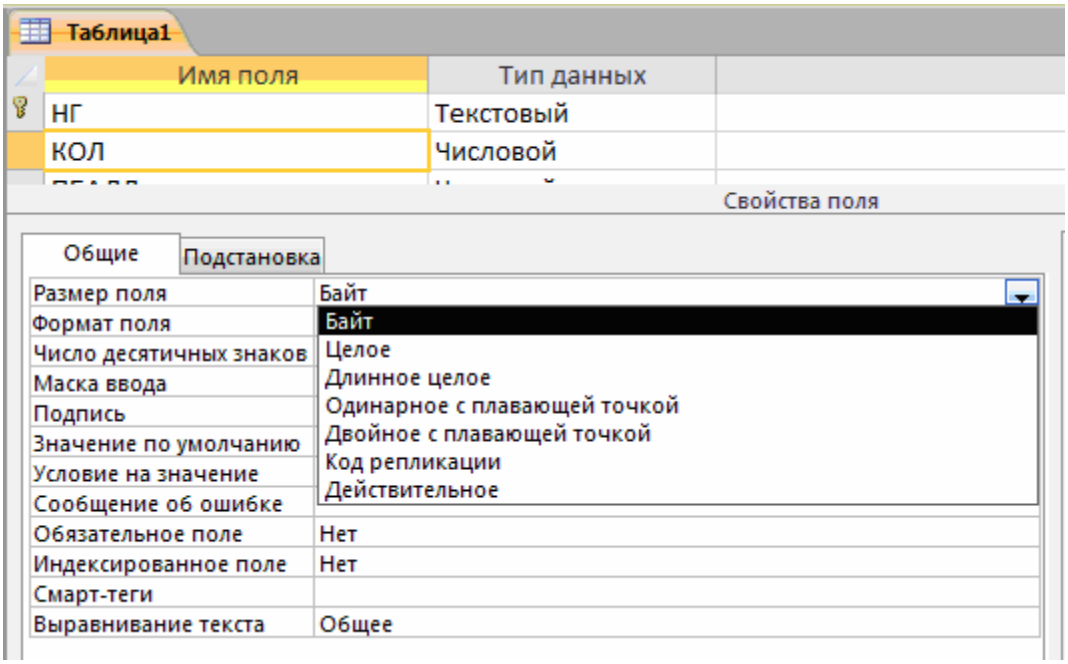


Рис. 1.6. Выбор размера для поля КОЛ (количество студентов в группе) в окне конструктора таблиц

Условие на значение, которое заносится в бланк запроса, является выражением, которое может быть сформировано с помощью построителя выражений (рис. 1.8). Построитель вызывается при нажатии кнопки справа от строки **Условие на значение**, в которую выражение должно быть введено (см. рис. 1.7).

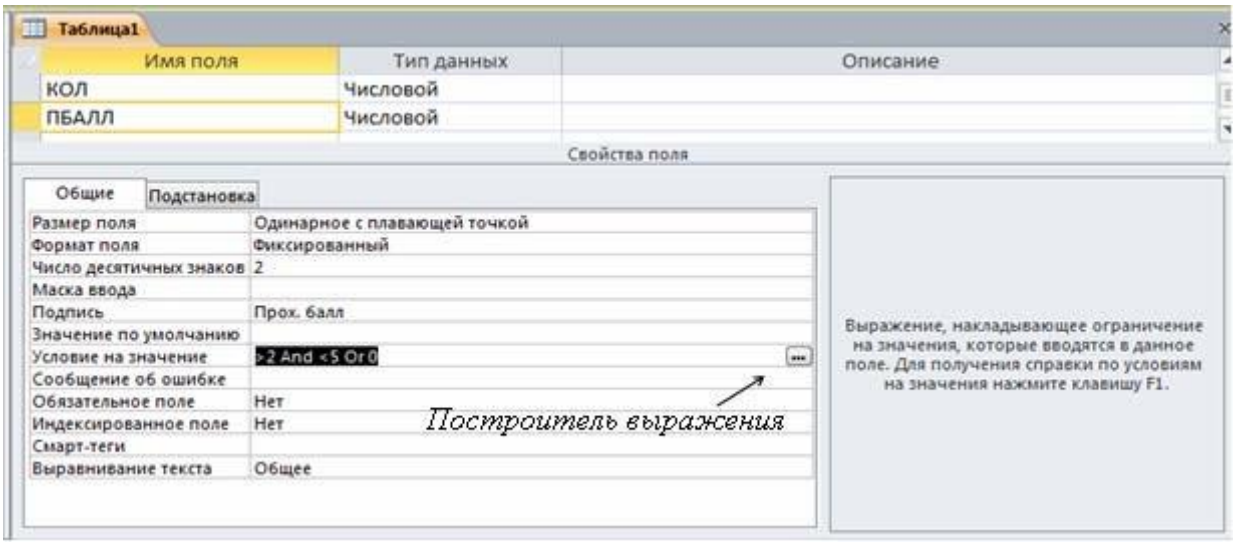


Рис. 1.7. Определение свойств поля ПБАЛЛ (средний проходной балл в группе)

Внимание

Для ввода операторов <больше равно> и <меньше равно> существуют специальные знаки. Не допускайте пробелов при вводе выражения. Нужные пробелы построитель введет сам.

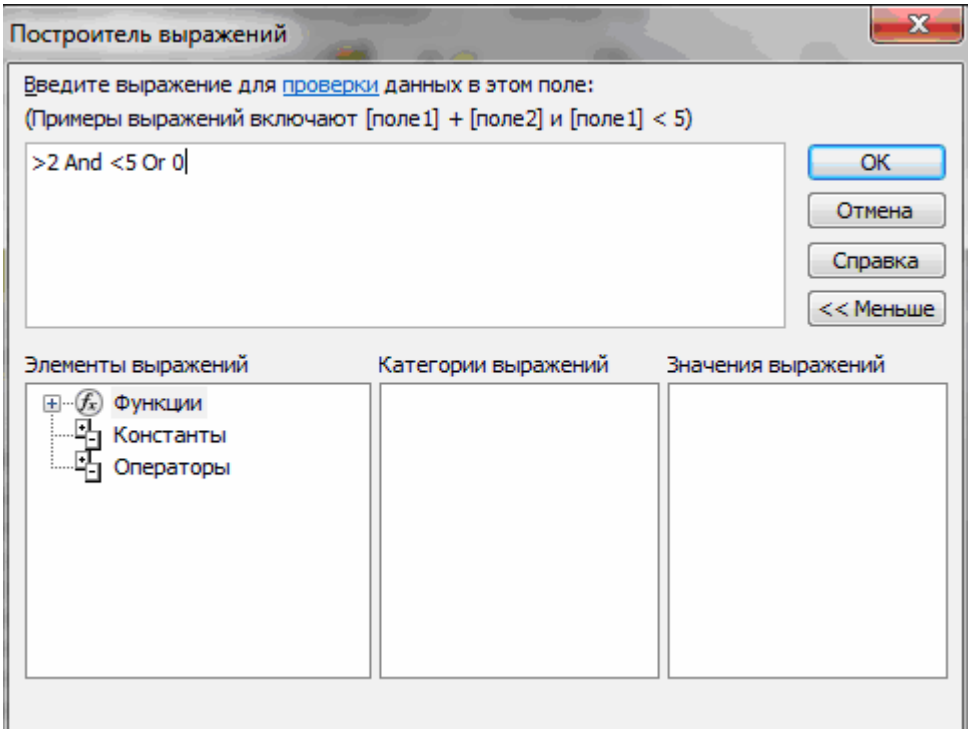


Рис. 1.8. Окно построителя выражений с выведенным списком всех операторов

После ввода выражения в окно построителя и нажатии клавиши <Enter> Access выполняет синтаксический анализ выражения и отобразит его в строке **Условие на значение**.

Теперь определим первичный ключ таблицы. Выделим поле НГ, щелкнув кнопкой мыши на области маркировки слева от имени поля, и нажмем кнопку **Ключевое поле** во вкладке **Конструктор**. Признаком установки ключа является изображение ключа слева от имени поля. Определим свойства ключевого поля в соответствии с табл. 3.1.

Сохраним созданную структуру таблицы и присвоим имя новой таблице – “ГРУППА”. Выполним команду **Файл|Сохранить** и введем это имя в окне **Сохранения** (рис. 1.9).

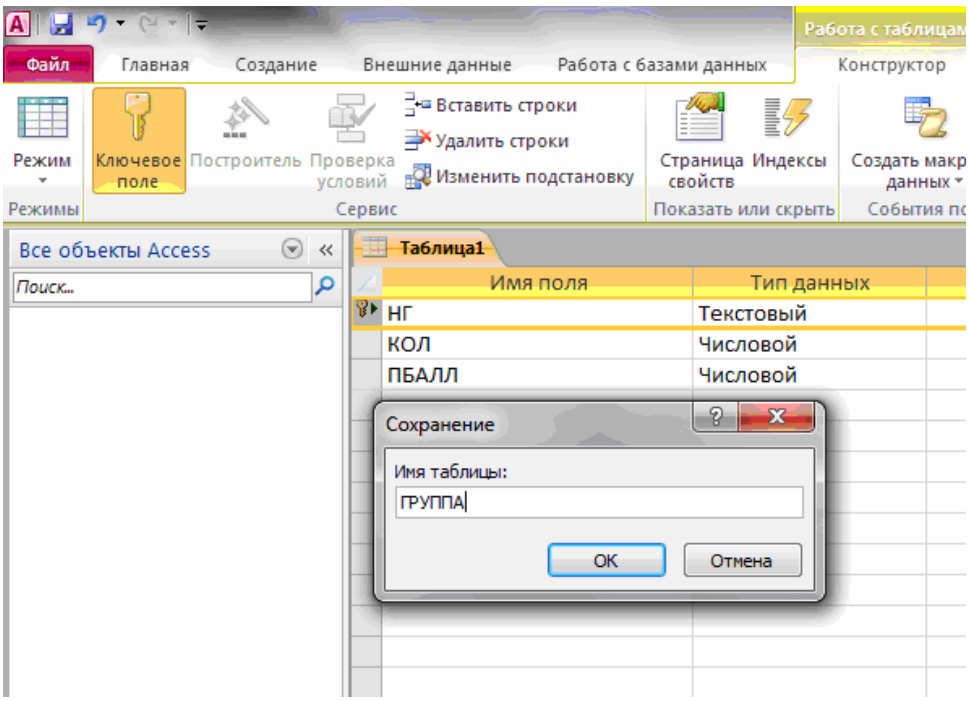


Рис. 1.9. Сохранение таблицы.

Таблица ГРУППА появится в списке объектов **Таблицы** в окне **Все объекты Access** (рис. 1.10).

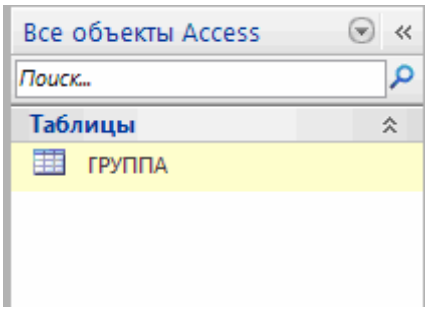


Рис. 1.0. Таблица ГРУППА.

При сохранении таблицы происходит обновление файла базы данных **Учебный процесс.mdb.**

Упражнение

Создайте структуру таблиц КАФЕДРА, ПРЕДМЕТ, ПРЕПОДАВАТЕЛЬ, СТУДЕНТ. При создании таблиц используйте проектные параметры их структуры, которые представлены в приложении А.

Рассмотрим некоторые особенности в создании структуры таблиц КАФЕДРА, ПРЕДМЕТ, ПРЕПОДАВАТЕЛЬ СТУДЕНТ базы данных “Учебный процесс”.

Использование данных типа Поле объекта OLE(OLE Object)

В таблице КАФЕДРА предусмотрено поле ФОТО, которое должно содержать фотографию заведующего, хранящуюся в формате графического редактора Paint в файле с расширением .bmp. Тип данных такого поля должен быть определен как **Поле объекта** OLE (OLE Object). Размещение этого объекта в поле производится на этапе заполнения полей таблицы. Объект может быть внедренным или связанным.

Замечание. OLE (Object Linking and Embedding – связывание и внедрение объекта) – это метод передачи информации в виде объектов между приложениями Windows. Поле объекта OLE является средством, позволяющим установить связь с объектами другого приложения или внедрить объект в базу данных. Объектами могут быть тексты простые и форматированные, рисунки, диаграммы, файлы звукозаписи (.WAV), музыка в формате MIDI (музыкально инструментальный цифровой интерфейс), файлы анимации(.FLI, .MMM), видеоклипы (.AVI), электронные таблицы из других приложений, поддерживающих это средство. Access, поддерживая OLE, полностью интегрирован с другими приложениями пакета Microsoft Office.

Внедренный объект сохраняется в файле базы данных и всегда является доступным. Двойным щелчком мыши на ячейке, содержащей внедренный объект, открывается возможность редактирования объекта средствами приложения, в котором объект был создан.

Связанный объект сохраняется в файле объекта. Файл объекта можно обновлять независимо от базы данных. Последние изменения будут выведены на экран при следующем открытии формы или отчета. При работе с базой данных также можно просматривать и редактировать объект.

Отредактированный связанный объект будет сохраняться в файле объекта, а не в файле базы данных. Связывание объекта удобно при работе с большими объектами, которые нежелательно включать в файл БД, а также с объектами,

используемыми в нескольких формах и отчетах. Если связанный файл объекта перемещен, необходимо повторно установить связь.

Замечание. Для отображения объекта OLE в форме или отчете необходимо создать элемент управления “Присоединенная рамка объекта”.

Использование данных типа Поле MEMO

В таблице ПРЕДМЕТ предусмотрено поле ПРОГР, которое будет содержать текстовое данное большой длины – краткую программу курса. Для такого поля выбирается тип данного – **Поле MEMO** (Memo). Ввод данных в это поле можно выполнить непосредственно в таблице, либо через область ввода, вызываемую нажатием клавиш <Shift>+<F2>.

Если тексты программ по предметам подготовлены в некотором текстовом редакторе, например, Microsoft Word, и хранятся в отдельных файлах, удобно для этого поля задать тип **Поле объекта OLE** (OLE Object) и при вводе значений в него установить связь с файлами.

Использование данных типа Гиперссылка(Hyperlink)

Специальный тип данных **Гиперссылка** (Hyperlink) позволяет хранить гиперссылки в полях таблиц БД.

Поля с типом **Гиперссылка** (Hyperlink) используются для перехода к объектам той же самой или другой базы данных Access, к документам, созданным в различных приложениях Microsoft Office, и расположенным на локальных или сетевых дисках, к страницам Web-серверов и документам других ресурсов в сети Internet или intranet. При щелчке мышью на гиперссылке осуществляется переход к документу, который открывается создавшим его приложением. Подробно создание гиперссылок в таблицах, формах и отчетах будет рассмотрено ниже. Там же рассмотрено создание гиперссылки в поле таблицы ПРЕПОДАВАТЕЛЬ.

Определение составного первичного ключа

В таблице СТУДЕНТ в составной первичный (уникальный) ключ входят поля НГ и НС. Для определения этого в режиме конструктора таблиц надо выделить оба эти поля, щелкая кнопкой мыши на области маркировки при нажатой клавише <Ctrl>. Затем нажать кнопку панели инструментов **Ключевое поле** .

Аналогично определяются составные ключи в таблицах ИЗУЧЕНИЕ УСПЕВАЕМОСТЬ.

Упражнение

Создайте структуру таблиц ИЗУЧЕНИЕ, УСПЕВАЕМОСТЬ. При создании таблиц используйте параметры, которые представлены в приложении А.

Приложение А. Описание свойств полей таблиц БД «Учебный процесс»

Таблица А.1. Описание свойств полей таблицы СТУДЕНТ

Имя поля	Ключевое поле	Уникальное поле	Обязательное поле	Тип данных	Размер	Число десятичных знаков	Подпись поля
НГ	Да		Да	Текстовый	3		Группа
НС	Да		Да	Текстовый	2		Номер студента в группе
ФИО			Да	Текстовый	15		ФИО
ГОДР			Нет	Числовой	Целое		Год рождения
АДРЕС			Нет	Текстовый	25		
ПБАЛЛ			Нет	Числовой	С плав. точкой 4 байта	2	Проходной балл

Таблица А.2. Описание свойств полей таблицы ГРУППА

Имя поля	Ключевое уникальное поле	Обязательное поле	Тип данных	Размер	Число десятичных знаков	Подпись поля	Условие на значение	Сообщение об ошибке
НГ	Да	Да	Текстовый	3		Ном. группы		
КОЛ		Нет	Числовой	Байт		Кол. ст. в гр.	>=0 And <=35	Кол. студ. больше доп.
ПБАЛЛ		Нет	Числовой	С плав. точкой 4 байта	2	Прох. балл	>2 And <5 Or 0	Ошибка в оценке

Таблица А.3. Описание свойств полей таблицы КАФЕДРА

Имя поля	Ключевое поле	Уникальное поле	Обязательное поле	Тип данных	Размер	Число десятичных знаков	Подпись поля
ККАФ	Да	Да	Да	Текстовый	2		Код
НКАФ			Нет	Текстовый	15		Название
ТЕЛ			Нет	Текстовый	9		
ЗАВ			Нет	Текстовый	15		ФИО зав. каф.
ФОТО			Нет	Поле объекта OLE			Фотография заведующего

Таблица А.4. Описание свойств полей таблицы ПРЕПОДАВАТЕЛЬ

Имя поля	Ключевое поле	Уникальное	Обязательное поле	Тип данных	Размер	Подпись поля
ТАБН	Да	Да[1]	Да	Текстовый	4	Таб. номер
ФИО			Да	Текстовый	30	ФИО препод.
СТ			Нет	Текстовый	15	Уч. степень
ЗВ			Нет	Текстовый	10	Уч. звание
ККАФ			Да	Текстовый	2	Код кафедры

Таблица А.5. Описание свойств полей таблицы ПРЕДМЕТ

Имя поля	Ключевое поле	Уникальное	Обязательное поле	Тип данных	Размер	Подпись поля	Условие на значение	Сообщение об ошибке
КП	Да	Да[1]	Да	Текстовый	2	Код предмета		
НП			Нет	Текстовый	15	Название предмета		
ЧАСЫ			Нет	Числовой	Целое	Всего часов	>0 And <=300	Число часов должно быть не более 300
ЛЕК			Нет	Числовой	Целое	Лекции		
ПР			Нет	Числовой	Целое	Практика		
ЧС			Нет	Числовой	Целое	Семестров		
ПРОГР				Поле МЕМО		Программа		

[1] Совпадения не допускаются.

Таблица А.6. Описание свойств полей таблицы ИЗУЧЕНИЕ

Имя поля	Ключевое поле	Уникальное	Обязательное поле	Тип данных	Размер	Число десятичных знаков	Подпись поля
НГ	Да		Да	Текстовый	3		Ном. группы
КП	Да		Да	Текстовый	2		Код. предмета
ТАБН	Да		Да	Текстовый	4		Таб. н. преп.
ВИДЗ	Да		Да	Текстовый	3		Вид занятий
ЧАСЫ			Нет	Числовой	Целое	0	Ср. балл по предм.
СБАЛЛ. ГР			Нет	Числовой	С пл. точкой 4 байта	2	

Таблица А.7. Описание свойств полей таблицы УСПЕВАЕМОСТЬ

Имя поля	Ключевое поле	Уникальное	Обязательное поле	Тип данных	Размер	Число десятичных знаков	Подпись поля
НГ	Да		Да	Текстовый	3		Номер группы
НС	Да		Да	Текстовый	2		Ном. студента
КП	Да		Да	Текстовый	2		Код предм.
ТАБН	Да		Да	Текстовый	4		Таб. н. препод.

ВИДЗ	Да	Да	Текстовый	3		Вид занятия
ОЦЕНКА		Нет	Числовой	Целое	0	

Урок 2. Ввод данных в таблицы

Ввод записей в режиме таблицы

В окне **Области навигации** установим курсор на таблице КАФЕДРА, щёлкнем правой кнопкой мыши на таблице и выберем в контекстном меню пункт **Открыть**. Таблица откроется в режиме таблицы. Заполним строки (записи) открывшейся таблицы в соответствии с названиями столбцов (полей).

Введем несколько записей в таблицу КАФЕДРА, данные для которых представлены в табл. 2.1.

Таблица 2.1. Данные таблицы КАФЕДРА

Код	Название	ТЕЛ.	ФИО зав. кафедрой
01	ИНФОРМАТИКИ	310-47-74	Игнатьева В. В.
02	МАТЕМАТИКИ	310-47-15	Иванов И. И.
03	ИСТОРИИ	310-47-16	Смирнова И. В.
04	ИНОСТРАННОГО ЯЗ.	310-47-17	Жданова А.Е.
05	ФИЗКУЛЬТУРЫ	310-47-67	Плетнев В.А.
06	ФИЛОСОФИИ	310-47-18	Бондаренко В.В.

Корректность вводимых данных (соответствие заданному типу поля, размеру и условию на значение, которые определены в свойствах полей в режиме конструктора) проверяется автоматически при их вводе. Отслеживается уникальность значений ключевых полей.

Отменить ввод значения в поле до перехода к другому полю можно, нажав клавишу **<Esc>** или на **Панели быстрого доступа**.

Если на **Панели быстрого доступа** отсутствует значек , тогда щелкнем правой кнопкой мыши на **Ленте** и в контекстном меню выберем пункт **Настройка панели быстрого доступа…**, в открывшемся окне **Параметры Access** выберем вкладку **Панель быстрого доступа**. В списке найдем команду **Отменить**, щелкнем на ней и далее нажмем кнопки **Добавить** и **ОК**, после чего на **Панели быстрого доступа** появиться нужная нам команда.

Для отмены изменений и в поле, и в записи нажмите клавишу **<ESC>** два раза. Переход от одного поля к другому можно выполнить клавишей **<Tab>, <ВПРАВО>, <ВЛЕВО>** или переводом курсора. Чтобы перейти к следующему полю в столбце, нажмите клавишу **<ВВЕРХ>** или **<ВНИЗ>**, или щелкните нужную ячейку.

Завершение ввода новых значений записи, т.е. редактирования, осуществляется при переходе к любой другой записи (при смене текущей записи). После перехода к другой записи можно отменить ввод (редактирование) всей записи, нажав .

Команда занимает одну позицию, но в зависимости от контекста приобретает различный смысл, а всплывающая подсказка сообщает о текущем ее назначении. По этой команде может быть отменено только одно последнее действие.

Размещение объекта OLE

Рассмотрим размещение объекта OLE на примере поля Фотография заведующего в таблице КАФЕДРА. Пусть фотографии хранятся в формате графического редактора Paint в файлах с расширением .bmp.

Рассмотрим вариант внедрения объекта в файл базы данных. Установим курсор в соответствующее поле таблицы. Щёлкнем правой кнопкой мыши на поле таблицы и выберем в контекстном меню пункт **Вставка объекта**. В окне Вставка объекта (рис. 2.1) надо отметить переключатель **Создать из файла**.

Окно Вставка объекта примет вид (рис. 2.1), который позволяет ввести имя файла с фотографией. Для поиска файла можно воспользоваться кнопкой Обзор, по которой выведется диалоговое окно, позволяющее просмотреть диски|папки и выбрать необходимый файл.

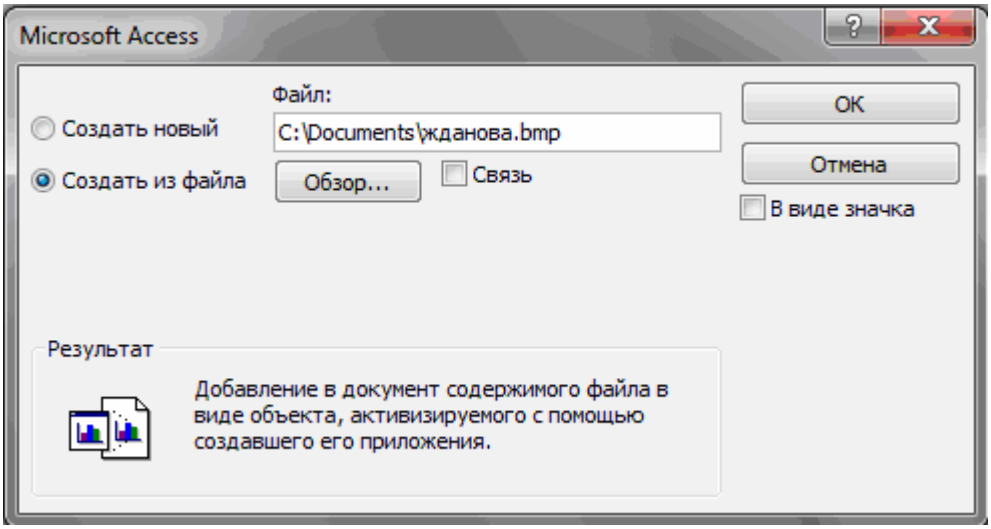


Рис. 2.1. Окно для выбора вставляемого объекта из файла

Флажок **Связь** по умолчанию не отмечен и, следовательно, содержимое файла будет введено в поле как *внедренный объект.* Увидеть содержимое поля можно через форму или отчет. Дальнейшие изменения файла не будут отражаться на встроенном объекте. Для введения в поле *связанного объекта* надо установить флажок **Связь**. Это сэкономит место в базе данных и даст возможность отображать все изменения, вносимые в файл.

В результате заполнения таблицы в соответствующем поле будет указан вид объекта "Точечный рисунок". Для просмотра внедренного объекта необходимо в соответствующем поле установить курсор и дважды щелкнуть кнопкой мыши.

Для отображения содержимого поля в виде значка, представляющего файл с документом, надо в окне **Вставка объекта** установить флажок **В виде значка**. Значок может быть использован для представления *связанного* объекта.

Ввод логически связанных записей

Введем несколько логически взаимосвязанных записей в таблицы ГРУППА и СТУДЕНТ.
Объекты ГРУППА и СТУДЕНТ связаны одно-многозначными отношениями, но пока не создана схема данных и связи между таблицами не установлены, система не может контролировать логическую взаимосвязь вводимых данных. Поэтому для получения целостной базы, в которой все записи подчиненной таблицы имеют логически связанную с ней главную запись, пользователю необходимо самому отслеживать логические связи записей. При вводе подчиненных записей необходимо проверять наличие записи в главной таблице, значение ключа которой совпадает со значением поля связи (внешнего ключа) вводимой подчиненной записи.

При непосредственном вводе в таблицу записей, логически связанных с записями другой таблицей, полезно отобразить на экране обе таблицы (рис. 2.2).

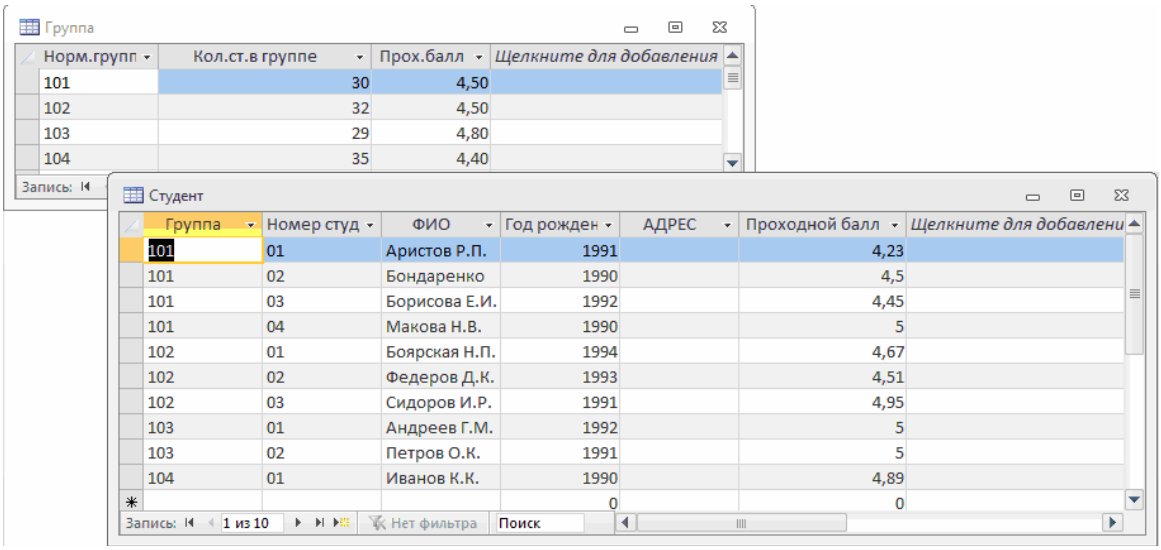


Рис. 2.2. Таблицы ГРУППА и СТУДЕНТ при вводе логически связанных записей

Для одновременного отображения открытых таблиц можно воспользоваться командой **Сверху вниз**, **Слева направо** или **Каскадом** на **Ленте** во вкладке **Главная,** в группе **Окно**.

Очевидно, что в базе данных сложной структуры при вводе данных непосредственно в таблицы не гарантируется надежное и корректное обслуживание данных. В рассматриваемом примере базы данных "Учебный процесс" при вводе данных в таблицы нижних уровней надо отслеживаться несколько вышестоящих.

Ниже будут рассмотрены средства Access, позволяющие автоматизировать контроль связной целостности и сделать загрузку базы данных простой и удобной процедурой. К таким средствам в первую очередь относятся **Схема данных** и **Форма**. Кроме того, в Access 2010 может быть использовано такое простое средство, как отображение в главной таблице записей подчиненных таблиц нескольких уровней.

Приложение В. Данные таблиц БД «Учебный процесс»

Таблица В.1. Данные таблицы СТУДЕНТ

Группа	Номер студента в группе	ФИО	Год рождения	Проходной балл
101	01	Аристов Р.П.	1979	4,25
101	02	Бондаренко С.А.	1978	4,50
101	03	Борисова Е.И.	1979	4,25
101	04	Макова Н.В.	1977	4.75
102	01	Боярская Н.П.	1977	4,50
102	02	Федоров Д. К.	1977	4,25
102	03	Сидоров И. Р.	1977	4,50
103	01	Андреев Г.М.	1978	4,25
103	02	Петров О.К.	1979	4.75
104	01	Иванов К. К.	1977	4,50

Таблица В.2. Данные таблицы ГРУППА

Номер группы	Количество студентов в группе	Проходной балл
101	30	4,50
102	32	4,50
103	29	4,80
104	35	4,40
105	35	4,80
201	35	3,90
202	30	4,00
203	28	4,70
204	25	4,00

Таблица В.3. Данные таблицы КАФЕДРА

Код	Название	ТЕЛ.	ФИО зав. кафедрой
01	ИНФОРМАТИКИ	310-47-74	Игнатьева В. В.
02	МАТЕМАТИКИ	310-47-15	Иванов И. И.
03	ИСТОРИИ	310-47-16	Смирнова И. В.
04	ИНОСТРАННОГО ЯЗ.	310-47-17	Жданова А.Е.
05	ФИЗКУЛЬТУРЫ	310-47-67	Плетнев В.А.
06	ФИЛОСОФИИ	310-47-18	Бондаренко В.В.

Таблица В.4. Данные таблицы ПРЕПОДОВАТЕЛЬ

Таб. номер	ФИО преподавателя	Уч. степень	Уч. звание	Код каф.
101	Андреев А. П.	д-р техн. наук	профессор	01
102	Апухтин И. С.	канд. техн. наук	доцент	01
103	Глухое И. Л.	канд. техн. наук	доцент	01
104	Сеченов Ю.Б.	канд. техн. наук	доцент	01
105	Чернов Л. К.	канд. техн. наук	доцент	01
201	Блюмкина И. П.	д-р физ.мат. наук	профессор	02
202	Львова П. Р.		ассистент	02
203	Шапошников С. И.	д-р техн. наук	профессор	02
204	Новиков П.Н.		ассистент	02
301	Ильясов И. Т.	канд. фил. наук	доцент	03
302	Пустынцев А.П.	канд. ист. наук	доцент	03
303	Романов Р.А.	канд. ист. наук	доцент	03
304	Цветков А.И.	канд. ист. наук	доцент	03
401	Сорокина М.Ф.	канд. фил. наук.	доцент	04
402	Богомолов П. Р.	канд. фил. наук.	доцент	04
403	Лысова М.И.	канд. фил. наук.	доцент	04
404	Шаповалова М.Ф.	канд. фил. наук.	доцент	04
405	Кудряшова Г.М.		ассистент	04
501	Жигарева П. Р.	канд. пед. наук	доцент	05
502	Егорова Т. И.		ст. преп.	05
503	Ермолин Е.Н.		ассистент	05
601	Логинов А.М.	канд. фил. наук	доцент	06
602	Яковлев П. П.	канд. фил. наук	доцент	06
603	Раков А.В.	канд. фил. наук	доцент	06
604	Соловьёв С. И.		ассистент	06

Таблица В.5. Данные таблицы ПРЕДМЕТ

Код предмета	Название предмета	Всего часов	Лекции	Практика	Семестров
01	Информатика	200	80	120	4
02	Математика	200	100	100	4
03	История	140	90	50	3
04	Иностранный яз.	200	0	200	4
05	Философия	100	40	60	2
06	Физкультура	100	0	100	2

Таблица В.6. Данные таблицы ИЗУЧЕНИЕ

Номер группы	Код предмета	Таб. номер преподавателя	Вид занятий	Часы
101	01	101	лек	40
101	01	102	пр	60
101	02	201	лек	50
101	02	202	пр	50
101	03	301	лек	48
101	03	302	пр	20
101	04	401	пр	50
101	05	501	лек	50
101	05	502	пр	50
101	06	601	лек	100
102	01	101	лек	100
102	01	103	пр	180
102	04	401	лек	100
105	01	101	лек	100
201	01	102	пр	180
201	02	201	пр	70
202	04	403	пр	100
203	01	101	лек	100
204	05	503	пр	100

Таблица В.7. Данные таблицы УСПЕВАЕМОСТЬ

Номер группы	Номер студента	Код предмета	Таб. номер преподавателя	Вид занятий	Оценка
101	01	01	101	лек	5
101	01	03	302	пр	0
101	02	01	101	лек	5
101	02	03	302	пр	0
101	03	01	101	лек	4
101	03	03	302	пр	0
101	04	01	101	лек	3
101	04	03	302	пр	0

Урок 3. Логическая структура базы данных

Логическая структура базы данных «Учебный процесс» приведена на рис.3.1. Она является адекватным отображением информационно-логической модели. Каждый информационный объект модели данных отображается соответствующей реляционной таблицей.

Связи между объектами модели данных реализуются одинаковыми реквизитами – *ключами связи* в соответствующих таблицах. При этом ключом связи всегда является уникальный ключ главной таблицы. Ключом связи в подчиненной таблице является либо некоторая часть уникального ключа в ней, либо поле, не входящее в состав первичного ключа (например, код кафедры в таблице ПРЕПОДАВАТЕЛЬ). Ключ связи в подчиненной таблице называется *внешним ключом*.

На этой схеме прямоугольники отображают таблицы БД с полным списком их полей, а связи показывают, по каким полям осуществляется взаимосвязь таблиц. Имена ключевых полей для наглядности выделены и находятся в верхней части полного списка полей каждой таблицы.

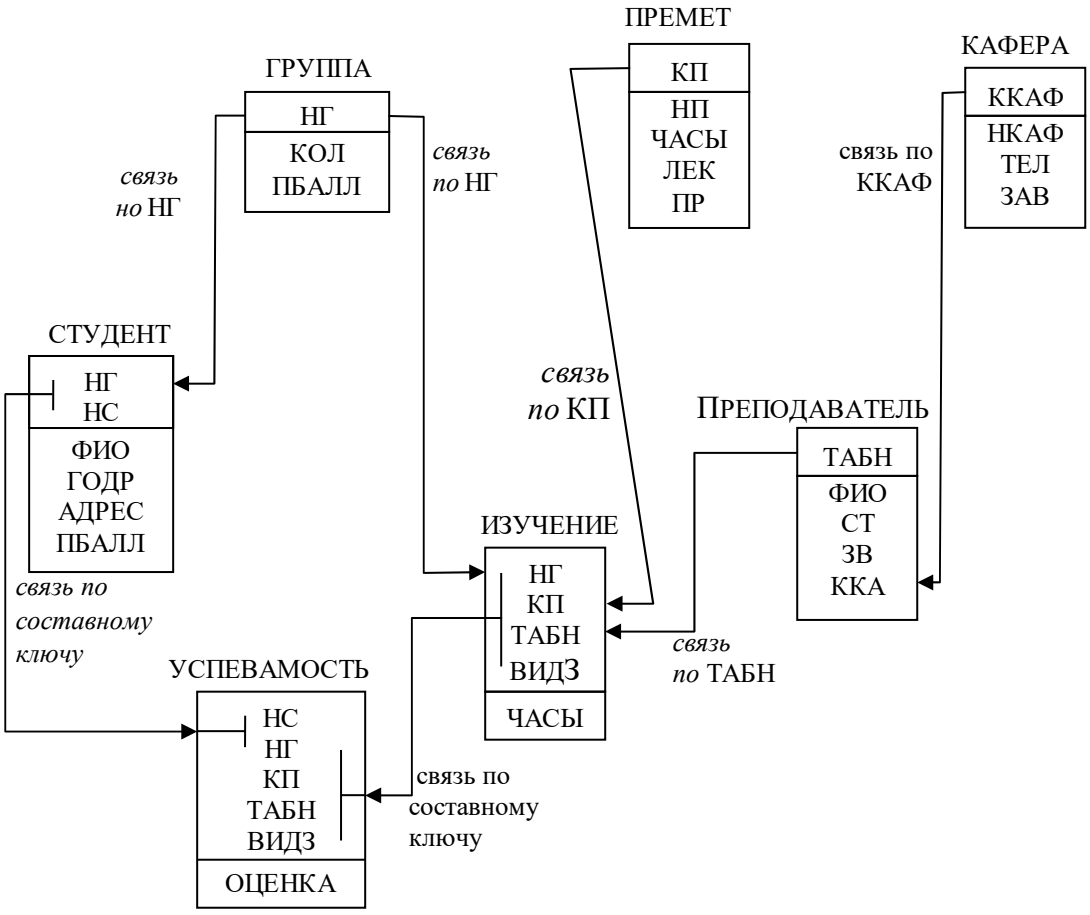


Рис. 3.1. Логическая структура реляционной базы данных “Учебный процесс”

Включение таблиц в схему данных

Рассмотрим процесс создания схемы данных в соответствии с логической структурой БД «Учебный процесс». При активном окне **Учебный процесс**: **База данных** нажмем на вкладку **Работа с базами данных**. В открывшемся меню нажмем **Схема данных**. Нажмем левой кнопкой мыши на свободном участке и выберем **Добавить таблицу**, выберем вкладку **Таблицы** и, нажимая кнопку **Добавить**, разместим в окне **Схема данных** все ранее созданные таблицы базы данных, которые отображены в окне **Добавление таблицы**. Затем нажмем кнопку **Закрыть**. В результате в окне **Схема данных** будут представлены все таблицы базы данных «Учебный процесс» со списками своих полей (см. рис. 3.2).

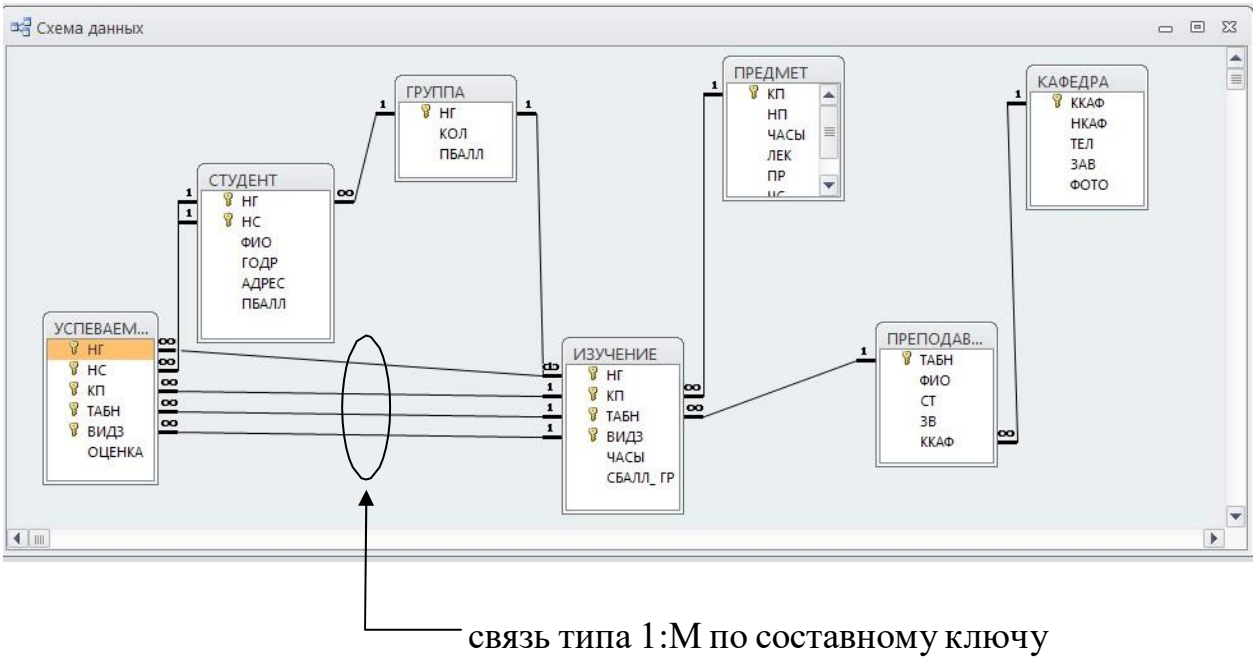


Рис. 3.2. Схема данных БД «Учебный процесс»

Определение связей между таблицами схемы данных

Определение связей по простому ключу. Установим связь между таблицами ГРУППА и СТУДЕНТ по простому ключу НГ.

Для этого в окне **Схемы данных** установим курсор мыши на ключевом поле НГ главной таблицы ГРУППА и перетащим его на поле НГ в подчиненной таблице СТУДЕНТ. В открывшемся окне **Изменение связей** в строке **Тип отно-**

шения установится значение **один-ко-многим**. Отметим параметр **Обеспечение целостности данных**. Если таблица ГРУППА и СТУДЕНТ ранее были заполнены корректными данными, между таблицами будет установлена связь, обозначенная на схеме как **1:∞**. Это свидетельствует о регистрации связи типа 1:М с параметром поддержания целостности. В противном случае появится сообщение о невозможности установить этот тип отношения.

Для обеспечения автоматической корректировки данных во взаимосвязанных таблицах установим флажок **каскадное обновление связанных полей и каскадное удаление связанных записей**.

Аналогичные действия выполняются для других пар таблиц КАФЕДРА → ПРЕПОДАВАТЕЛЬ (ключ ККАФ), ПРЕДМЕТ → ИЗУЧЕНИЕ (ключ КН), ПРЕПОДАВАТЕЛЬ → ИЗУЧЕНИЕ (ключ ТАБН), ГРУППА → ИЗУЧЕНИЕ (ключ НГ).

Определение связей по составному ключу. Определим связи между таблицами СТУДЕНТ → УСПЕВАЕМОСТЬ, которые связаны по составному ключу НГ+НС. Для этого в главной таблице СТУДЕНТ выделим оба этих поля, удерживая клавишу Ctrl. Перетащим оба поля на поле НГ в подчиненной таблице УСПЕВАЕМОСТЬ.

В окне **Изменение связи** (рис. 3.3) для ключевого поля НС главной таблицы ТАБЛИЦА/ЗАПРОС выберем соответствующее поле подчиненной таблица СВЯЗАННАЯ ТАБЛИЦА/ЗАПРОС. В этом же окне установим режимы **Обеспечение целостности данных** и другие параметры связи.

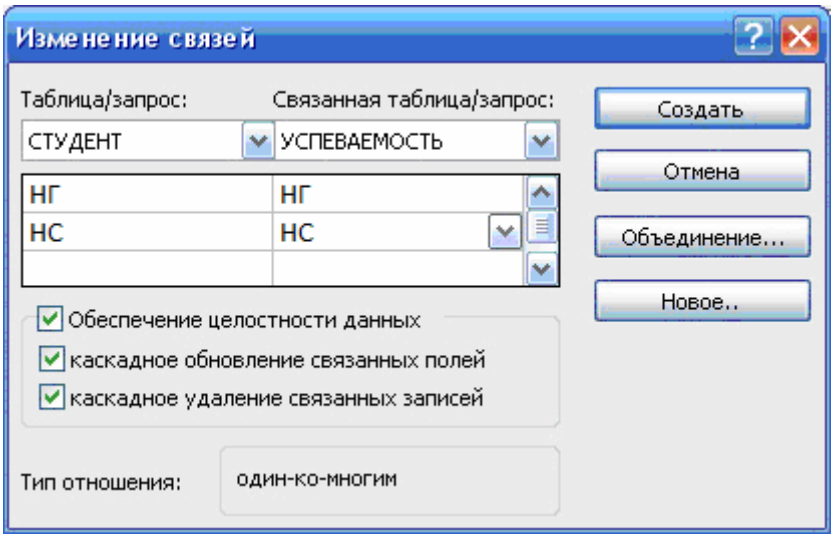


Рис. 3.3. Окно выбора параметров **Связи**

Аналогично определяются связи между парой таблиц ИЗУЧЕНИЕ → УСПЕВАЕМОСТЬ (составной ключ связи – НГ+КП+ТАБН+ВИДЗ).

После определения связей таблицы могут перемещаться в пределах рабочего пространства окна схемы данных. Перемещения и изменения размеров таблиц осуществляются принятыми в Windows способами. На рис. 3.2 показана схема данных «Учебный процесс», где таблицы размещены в соответствии с их относительной подчиненностью.

Проверка работоспособности схемы данных, поддержание целостности осуществляется при конструировании форм, запросов, отчетов и их использовании, а также при непосредственной корректировке таблиц.

Проверка поддержания целостности в базе данных

На рис. 3.2 в созданной схеме данных БД «Учебный процесс» все связи отмечены символами **1,** ∞, что свидетельствует об установлении связей типа 1:М (по простому или составному ключу), для которых будет обеспечиваться целостность данных.

Проверим поддержание целостности при внесении изменений в таблицы ГРУППА → СТУДЕНТ, связанные одно-многозначными отношениями.

Проверка целостности при изменении значений связанных полей в таблицах. Откроем таблицу ГРУППА в режиме таблицы. Изменим значение ключевого поля НГ (номер группы) в одной из записей главной таблицы ГРУППА. Убедимся, что во всех записях подчиненной таблицы СТУДЕНТ, для студентов, обучающихся в этой группе, автоматически также изменится значение поля НГ. Изменение происходит поскольку был установлен параметр **каскадное обновление связных полей** (см рис. 3.3). Причем это изменение осуществляется мгновенно, как только изменяемая запись перестает быть текущей. Для наблюдений за автоматическими изменениями в подчиненной таблице откроем на экране одновременно таблицы ГРУППА и СТУДЕНТ. Открыв другие подчиненные таблицы, можно убедиться, что каскадное обновление распространяется также на подчиненные таблицы следующих уровней ИЗУЧЕНИЕ и УСПЕВАЕМОСТЬ.

Изменим значение ключа связи НГ в подчиненной таблице СТУДЕНТ на значение, не существующее в записях таблицы ГРУППА, и убедимся, что такое изменение запрещено, т.к. при поддержании целостности не может существовать запись подчиненной таблицы с ключом связи, которого нет в главной таблице.

Проверка при добавлении записей в подчиненную таблицу. Убедимся, что вместе с удалением записи в главной таблице ГРУППА удаляются все подчиненные записи, т. к. был установлен параметр **каскадное удаление связанных записей**.

Заметим, если каскадное удаление не разрешено, невозможно удалить запись в главной таблице, если имеются связанные с ней записи в подчиненной.

Если выполнение изменений невозможно, появится сообщение (рис. 3.4).

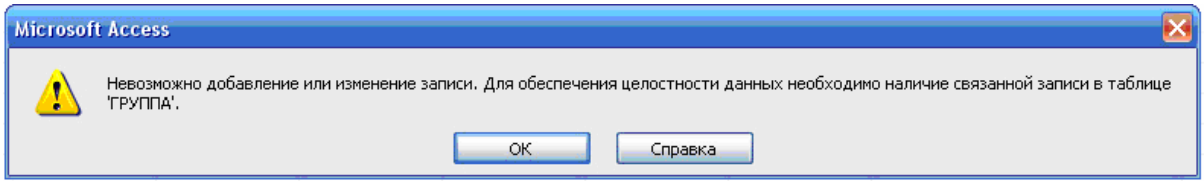


Рис.3.4. Сообщение о невозможности внесения изменений в таблицу СТУДЕНТ

Определение технологии загрузки базы данных

Рассмотрим технологию загрузки на примере базы данных “Учебный процесс”, в котором связи между таблицами соответствуют схеме данных, приведенной в этой главе .

Документы-источники загрузки этой базы данных перечислены при описании предметной области в главе 2.

Таблицы-объекты загрузки

Определим объекты загрузки – группы из взаимосвязанных таблиц, подлежащих загрузке из одного документа:

1. Таблицы ГРУППА →СТУДЕНТ. Загрузка записей этих таблиц будет производиться одновременно из одного документа “Список студентов группы”, что обеспечивает формирование взаимосвязей записей студентов с соответствующей записью группы. При этом будет обеспечен однократный ввод реквизитов ГРУППЫ и однократный ввод значений НГ для всех студентов группы.
2. Таблицы КАФЕДРА→ ПРЕПОДАВАТЕЛЬ. Загрузка записей этих таблиц также будет производиться одновременно из одного документа “Список преподавателей кафедры”, что обеспечивает сразу формирование взаимосвязей записей преподавателей с соответствующей записью кафедры. При этом будет обеспечен однократный ввод реквизитов КАФЕДРЫ и однократный ввод значений ККАФ для всех преподавателей кафедры.
3. Таблица ПРЕДМЕТ. Загрузка этой таблицы может производиться из плана произведения занятий в группах. При этом достаточно ввести составной

идентификатор занятия, не вводя наименование предмета и фамилию преподавателя.

4. Таблица ИЗУЧЕНИЕ. Загрузка этой таблицы может производиться из плана проведения занятий в группах. При этом достаточно ввести составной идентификатор занятия, не вводя наименование предмета и фамилию преподавателя.
5. Таблица УСПЕВАЕМОСТЬ. Загрузка этой таблицы может производиться из заполнения экзаменационной ведомости группы. При этом достаточно точно ввести составной идентификатор, не вводя фамилию студента, наименование предмета, фамилию преподавателя.

Последовательность загрузки таблиц

Загрузка первых трех объектов ГРУППА →СТУДЕНТ, КАФЕДРА, ПРЕПОДАВАТЕЛЬ, ПРЕДМЕТ может производиться в любой последовательности т.к. эти группы таблиц на схеме данных не находятся в подчиненных отношениях.

Загрузка таблицы ИЗУЧЧЕНИЕ должна производиться после загрузки таблиц ГРУППА, ПРЕДМЕТ и ПРЕПОДАВАТНЛЬ, т.к. таблица ИЗУЧЕНИЕ в схеме данных подчинена этим таблицам.

Загрузка таблицы УСПЕВАЕМОСТЬ может производиться только после загрузки таблиц СТУДЕНТ и ИЗУЧЕНИЕ, т.к. таблица УСПЕВАЕМОСТЬ в схеме данных подчинена этим таблицам.

Технология загрузки базы данных “Учебный процесс” может быть представлена в виде таблицы и (см. табл. 3.1).

Таким образом, в результате определена последовательность этапов загрузки данных “Учебный процесс”, а также объекты загрузки на отдельных этапах и соответствующие документы-источники данных.

Таблица 3.1. Технология загрузки базы данных “Учебный процесс”

Таблица БД-объекты загрузки	Документы-источники	Вид информации	Этап загрузки	Примечание
ПРЕДМЕТ	Учебный план, программа	Справочная	|	
ГРУППА → СТУДЕНТ	Список студентов группы	Справочная	|	Независимые
КАФЕДРА → ПРЕПОДАВАТЕЛЬ	Список преподавателей кафедры	Справочная	|	Этапы
ИЗУЧЕНИЕ	План занятий	Учетная	||	

	группы		
УСПЕВАЕМОСТЬ	Экзаменационная ведомость	Учетная	|||

Определив этапы загрузки БД можно приступить непосредственно к проектированию форм и их созданию средствами Access. Технология разработки форм на примере базы данных “Учебный процесс” будет рассмотрена ниже.

Урок 4. Однотабличные формы

Форма на основе таблицы может быть построена как самостоятельная для загрузки, просмотра и корректировки таблиц, а также как вспомогательная для включения в какую-либо составную форму.

Любая форма, с помощью которой можно просматривать, вводить или редактировать записи таблиц БД, должна быть предварительно спроектирована и далее сконструирована средствами Access.

Для создания формы могут быть использованы мастера Access.Однако, точное формирование макета формы в соответствии с требованиями, выработанными пользователем в процессе ее проектирования, обеспечивается средствами конструирования форм. Ниже рассматриваются основные понятия и техника конструирования однотабличных форм.

Конструирование формы

Для конструирования форм в Access используется Конструктор форм. При конструировании однотабличной формы определяется таблица БД, на основе которой создается форма, выбираются поля таблицы, которые должны быть представлены в форме, осуществляется их размещение в макете формы, создаются вычисляемые поля и другие графические элементы: кнопки, выключатели, элементы оформления, поясняющий текст, рисунки. Для настройки различных элементов форм используется типовой набор их свойств.

Области и элементы формы в режиме конструктора

Форма в режиме Конструктора форм имеет три области: **Область данных Заголовок формы**, и **Примечание формы**, которые могут быть образованы по команде меню **Вид**|**Заголовок/примечание формы**. Области формы наполняются различными графическими объектами.

Элементы или графические объекты. Графические объекты, связанные с записями таблиц и предназначенные для отображения данных некоторого поля, называются *элементами управления*. Основными типами элементами управления: **Поле Поле со списком, Список**. Тип элемента управления, выбираемый для поля *по умолчанию*, определяется в свойствах поля таблицы базы данных, с которым связано поле формы. Задается это свойство при определенных типах данных поля в режиме конструктора таблиц на вкладке **Подстановка.**

Графические объекты, не связанные с таблицами или запросами, предназначены прежде всего для создания макета формы и содержат *надписи* полей (пользовательские названия реквизитов).

Создание однотабличной формы с помощью инструмента «Форма»

В области переходов щелкните таблицу или запрос с данными, которые должны отображаться в форме. На вкладке **Создать** в группе **Формы** нажмите кнопку **Форма.** (см. рис. 4.1)

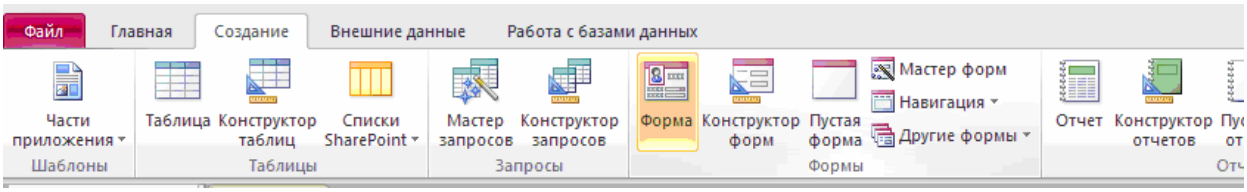


Рис 4.1. Кнопка **Форма** на вкладке **Создать** в группе **Формы**

Будет создана новая форма и отображена в режиме макета. В режиме макета можно внести изменения в структуру формы при одновременном отображении данных. Например, можно настроить размер полей в соответствии с данными.

Работа команды «**Форма**» завершается отображением формы (рис. 4.2). При этом появляется панель инструментов **Режим формы**. Кнопки этой панели по назначению аналогичны кнопкам панели **Таблица в режиме таблицы**, рассмотренной в главе 3.

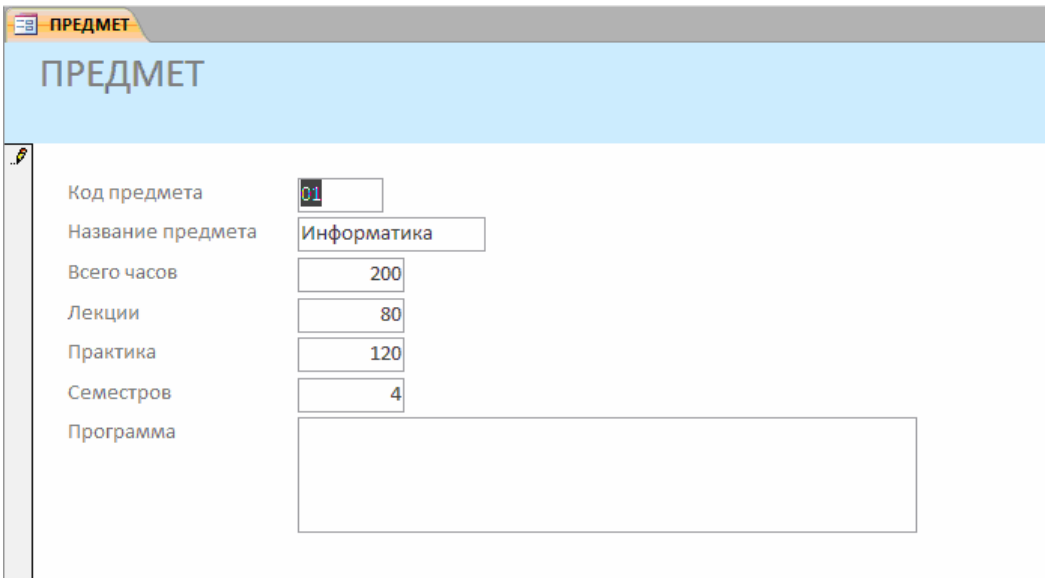


Рис 4.2. Результат работы команды **«Форма»** по таблице ПРЕДМЕТ

Внимание.

Подписи полей в форме соответствуют заданным в их свойствах при определении структуры таблицы. Подпись формы соответствует имени таблицы источника, при сохранении формы можно подтвердить это имя или изменить его.

Редактирование формы

Переход в режим конструктора формы. Для уточнения текста надписей, местоположения, размера, шрифта и других параметров отображения элементов формы необходимо перейти в режим конструктора форм.

При открытой форме переход в режим конструктора можно осуществить нажатием в вкладке **Режимы** кнопки **Режим** (рис 4.3). Кнопка **Режим** обеспечена списком, развернув который можно выбрать необходимый режим представления формы.

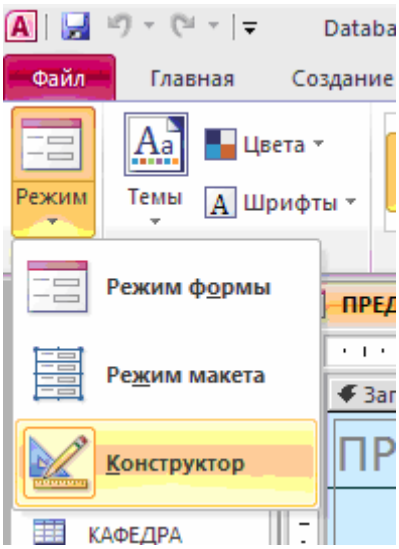


Рис. 4.3. Список режимов кнопки **Режим**

Переход в режим конструктора можно осуществить также с помощью **Контекстного меню** данной формы (рис 4.4)

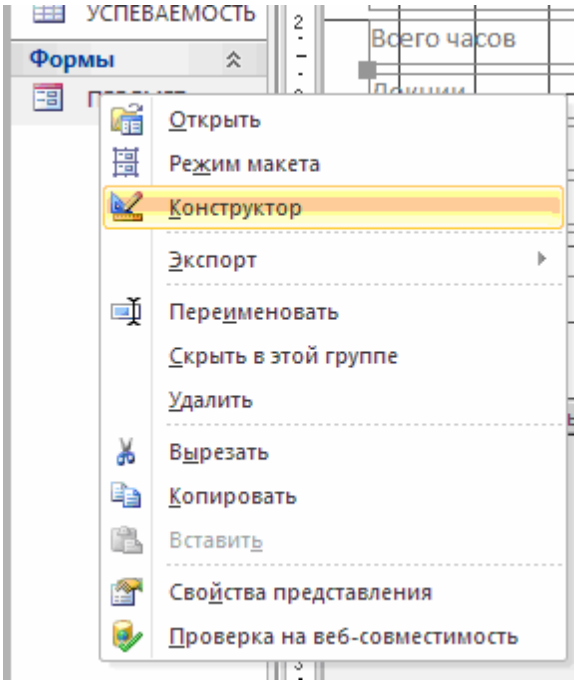


Рис. 4.4

После перехода в режим конструктора созданная форма откроется в окне конструктора форм (рис. 4.5).

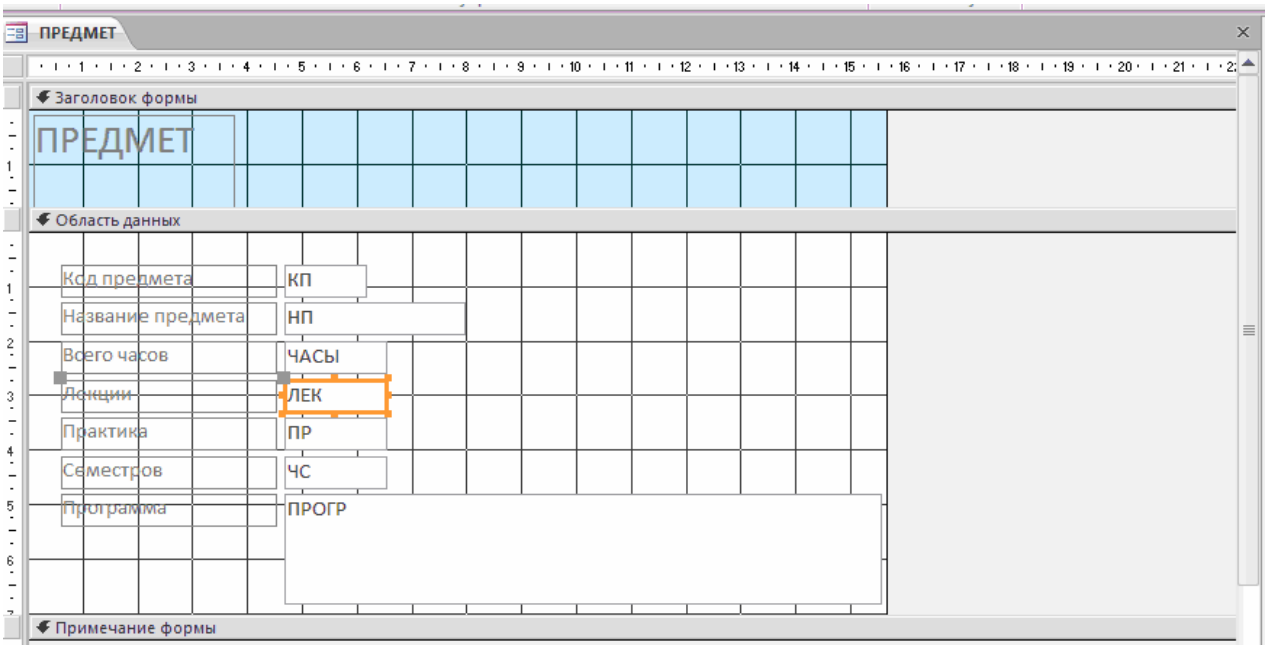


Рис. 4.5. Режим конструктора формы, полученной командой «**Форма**» по таблице ПРЕДМЕТ

После выбора режима конструктора в окне Access появляются панель **Конструктор форм** и **Панель элементов**. Панель форматирования **Формат (Форма/ Отчет)** может быть вызвана при активном окне формы по команде меню **Вид|Панели инструментов|Формат (Форма/Отчет)**.

Создание заголовка. Для ввода текста заголовка в полученную форму в окне конструктора (рис. 4.6) расширим область заголовка формы, установив курсор мыши на границу области данных и перетаскивая эту границу на нужное расстояние.

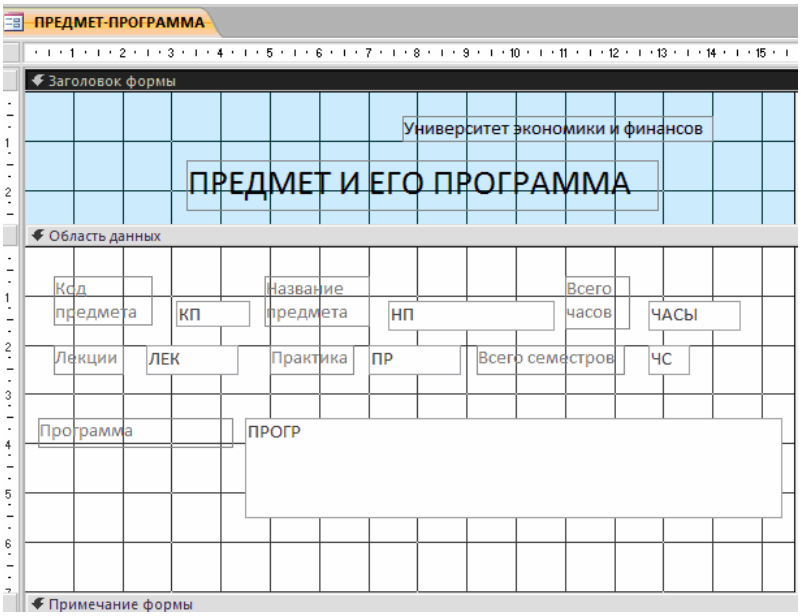


Рис. 4.6. Форма ПРЕДМЕТ-ПРОГРАММА в режиме конструктора форм

Для ввода текста заголовка надо создать графический элемент **Надпись**. Начинается создание элемента щелчком мыши на кнопке панели элементов **Надпись**. Теперь на панели форматирования можно выбрать нужный шрифт и другие параметры оформления. Переместим курсор мыши на место начала текста. Нажмем кнопку мыши и, не отпуская ее, растянем рамку текста до нужного размера. Введем текст. Создание элемента **Надпись** завершается нажатием клавиши <Enter> или щелчком мыши вне рамки элемента.

Форматирование элемента **Надпись** может быть выполнено в любой момент. Для этого элемент надо выделить щелчком мыши внутри его рамки. Для изменения текста надписи курсор должен быть переведен на текст, при этом сам элемент остается невыделенным, а команды форматирования недоступны.

Элемент **Надписи** может быть перемещен в пределах области заголовка, могут быть также изменены размеры рамки элемента. Рамка помеченного эле-

мента может быть растянута или сжата при размещение курсора на специальных точках рамки, в которых появляется двунаправленная стрелка. Перемещение возможно при появлении изображения руки.

Замечание
При установке курсора мыши на любой границе рамки курсор отображается в виде раскрытой ладони и тогда возможно перемещать элемента и за пределы области. При установке курсора в левом верхнем углу курсор отображается в виде указательного пальца и перемещение возможно только в пределах области, которая при этом может автоматически расширяться.

Для удаления элемента его надо выделить и нажать клавишу <Del>.

Изменение надписей и отображения значений полей. При редактировании связанных элементов **Поле** и **Надпись** , если между ними установлена связь, или аналогичной пары элементов, полученной с помощью кнопки **Список полей** на панели конструктора форм, следует иметь в виду, что независимое перемещение поля и его надписи возможно, только если курсор примет вид указательного пальца. В противном случае оба элемента перемещаются синхронно. Остальные действия по внесению изменений в эти элементы осуществляется аналогично рассмотренному при формировании элемента в области заголовка.

Изменение свойств. Редактирование формы и ее элементов может быть выполнено не только графическими средствами, но и путем изменения их свойств. Для этого необходимо с помощью двойного щелчка открыть **Окно свойств** необходимого элемента. А так же его можно открыть с помощью контекстного меню. На рис 4.7 показана вкладка **Макет** в окне свойств **Поле: НП** .

Сохранение формы после редактирования. По завершении редактирования формы она может быть сохранена. Если редактируемая форма еще не сохранялась, выполняется команда меню **Файл**|**Сохранить** или нажимается кнопка панели инструментов **Сохранить**. Можно сохранить форму и при ее закрытии командой **Файл**|**Закрыть** (File|Close) или нажатием кнопки (Закрыть) окна формы. Далее надо подтвердить необходимость их сохранения, и в диалоговом окне **Сохранение** ввести название (ПРЕДМЕТ-ПРОГРАММА) в текстовом поле **Имя формы** (рис. 4.8).

Если редактируемая форма была ранее сохранена, то для сохранения измененной формы под новым именем над выполнить команду меню **Файл**| **Сохранить как**.

Окончательный вид отредактированной формы в режиме конструктора представлен на рис 4.6.

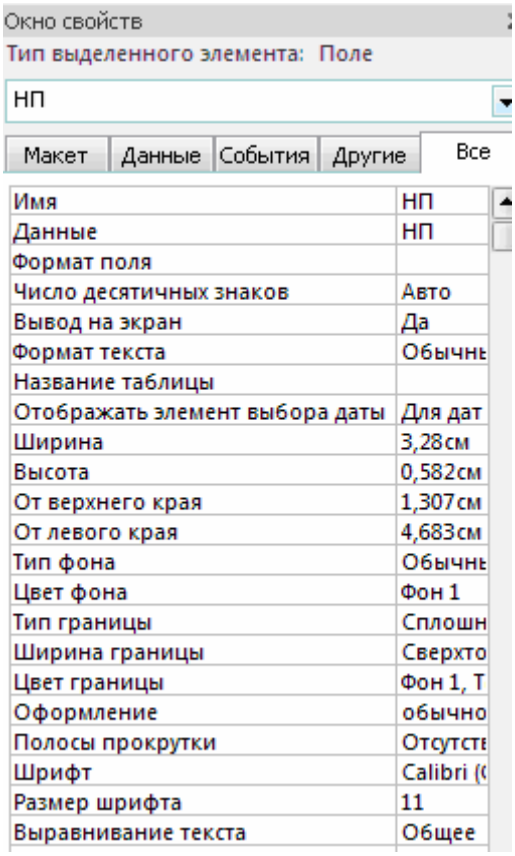


Рис 4.7. Свойства для поля НП в форме, установленные мастером при создании формы для таблицы ПРЕДМЕТ

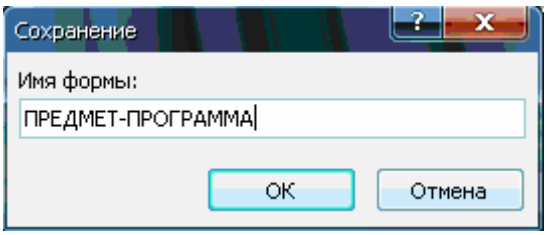


Рис. 4.8. Ввод имени формы при ее сохранении

Работа с данными таблицы в режиме формы

Завершив редактирование формы, приступим к работе с таблицей ПРЕДМЕТ через форму. Для перехода в режим формы из режима конструктора нужно нажать в вкладке **Режимы** кнопки **Режим** и выбрать **Режим формы**.

Для загрузки , просмотра и корректировки данных таблицы ПРЕДМЕТ через ранее сохраненную форму в окне: **база данных** в группе **Объекты** надо перейти к строке **Формы** и ,выделив в рабочей области название формы ПРЕДМЕТ-ПРОГРАММА, нажать кнопку **Открыть.**

На рис.4.9. приводится форма ПРЕДМЕТ-ПРОГРАММА в режиме формы, в которой отображены данные из одной строки таблицы ПРЕДМЕТ.

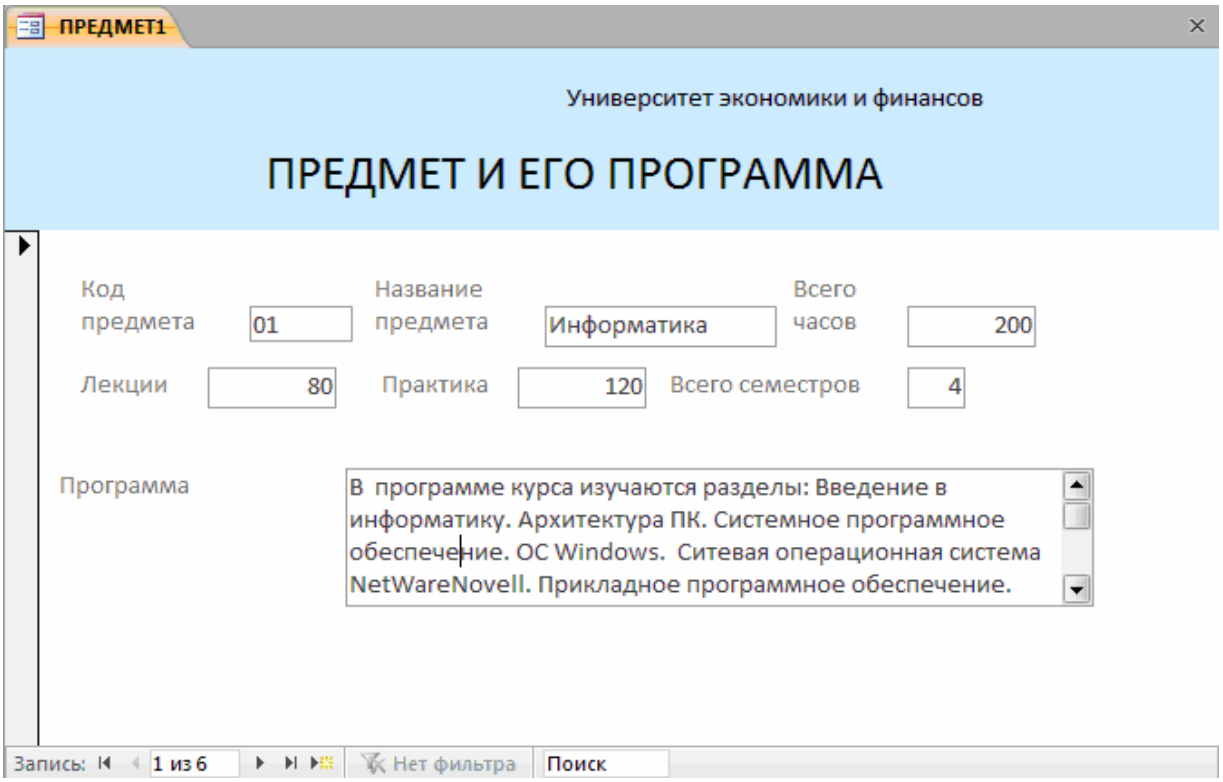


Рис. 4.7. Форма ввода-вывода для работы с данными таблицы ПРЕДМЕТ

Данные для загрузки таблицы ПРЕДМЕТ приведены в соответствующей таблице приложения главы 3.

Значения, вводимые в поля формы, должны соответствовать типам данных и их свойствам, заданным при определении структуры таблицы.

Для завершения создания (редактирования) записи таблицы ПРЕДМЕТ достаточно перейти к другой записи в поле номера записи внизу формы.

Урок 5. Формы для загрузки двух таблиц

В настоящем разделе на примере подробно рассматривается последовательность конкретных действий при разработке составной формы для загрузки двух таблиц, связанных одно-многозначными отношениями. Технология разработки любой многотабличной формы включает *проектирование* макета формы (см. выше раздел «*Технология загрузки базы данных с использованием форм*») и процесс *конструирования* средствами Access.

В соответствии с разделом «*Этапы загрузки базы данных и проектирования форм*» при проектировании составной формы выполним:

- Определение подсхемы данных для разрабатываемой формы
- Определение общей структуры экранной формы, т. е. ее макета в соответствии со структурой входного документа и подсхемой данных
- Определение состава и размещения реквизитов для каждой из частей составной формы

На основе результатов проектирования осуществим конструирование экранной формы средствами Access.

Ниже рассматривается технология разработки составной формы для одновременной загрузки и работы с данными таблиц ГРУППА и СТУДЕНТ в соответствии с этапами загрузки базы данных «Учебный процесс», определенными выше в табл. 3.1.

Проектирование форм на основе двух таблиц

Осуществим проектирование формы для загрузки данных в таблицу ГРУППА и СТУДЕНТ, просмотра и редактирования этих данных. Документом-источником такой формы является «Список студентов группы» (см. рис. 2.3). Из этого документа будут загружаться одновременно две таблицы: ГРУППА и СТУДЕНТ, которые в совокупности образуют объект загрузки.

Определение подсхемы данных для составной формы

Поскольку объект загрузки: ГРУППА → СТУДЕНТ не подчиняется в схеме данных другим таблицам, подсхема, необходимая для построения формы, не должна включать других таблиц. Такая подсхема приведена на рис.5.1.

Определение общей структуры составной формы

В соответствии с приведенной подсхемой определим общую структуру составной формы, которую назовем СПИСОК ГРУППЫ.

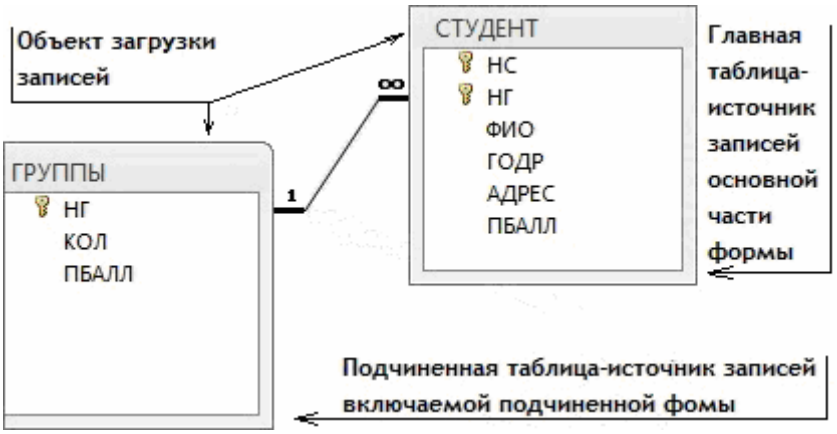


Рис.5.1. Подсхема данных для конструирования формы на базе таблиц ГРУППА → СТУДЕНТ

Для того чтобы обеспечить удобный ввод данных с документа, в форме предусмотрим основную часть с реквизитами группы и подчиненную с записями о студентах группы. Подчиненную форму назовем СПИСОК СТУДЕНТОВ.

Таким образом, составную форму СПИСОК ГРУППЫ определяют:

- *Тип формы* – многотабличная
- *Источник записей для основной части формы* – таблица ГРУППА
- *Включаемая подчиненная форма* – СПИСОК СТУДЕНТОВ

Подчиненную форму СПИСОК СТУДЕНТОВ определяют:

- *Тип формы* – подчиненная, многозаписевая
- *Источник записей* – таблица СТУДЕНТ

На подсхеме (рис.5.1) показано назначение таблиц при создании формы.

Размещение реквизитов основной и подчиненной формы

Размещение реквизитов в основной части формы и подчиненной форме должно соответствовать входному документу «Список студентов группы» (см. главу 2).

В основной части составной формы СПИСОК ГРУППЫ вверху разместим реквизиты, соответствующие полям таблицы ГРУППА:

- Номер группы (НГ - ключ)
- Количество студентов (КОЛ)
- Средний проходной бал в группе (ПБАЛЛ)

В подчиненной форме СПИСОК СТУДЕНТОВ разместим в качестве заголовков столбцов многозаписевой формы названия реквизитов соответствующих полей таблицы СТУДЕНТ:

- Номер студента в группе (НС)
- Фамилия И. О. (ФИО)
- Год рождения (ГОДР)
- Адрес (АДРЕС)
- Средний балл при поступлении (ПБАЛЛ)

Заметим, что ключевое поле НГ не включено в подчиненную форму, т. к. поле связи НГ включено в основную часть формы.

Создание формы для двух таблиц с помощью мастера

Осуществим средствами Access конструирование формы для одновременной загрузки и корректировки двух таблиц ГРУППА и СТУДЕНТ в базе данных «Учебный процесс».

Определение таблиц и полей для основной и включаемой частей формы

В окне: **базы данных** выбираем в группе **Объекты** строку **Формы** и нажимаем кнопку **Создать**.

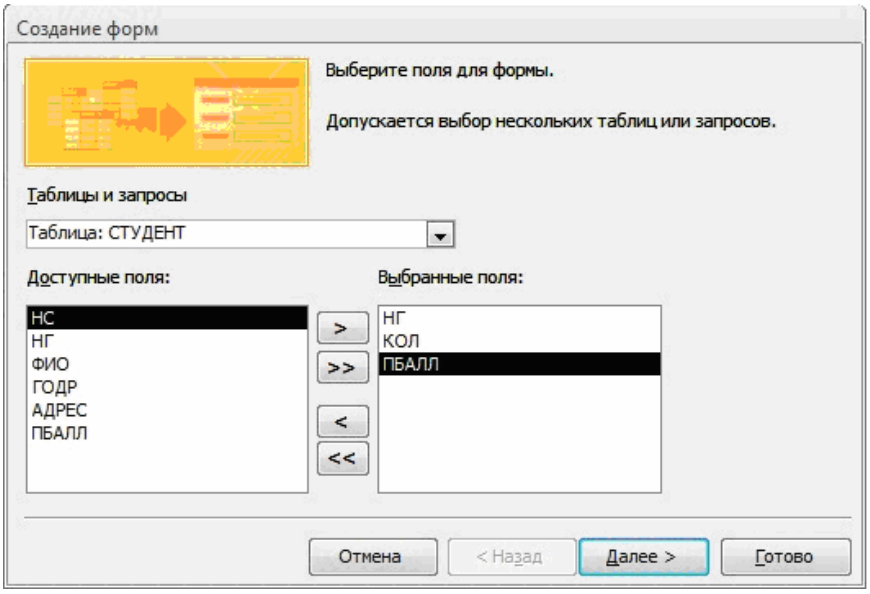


Рис. 5.2. Выбор полей для формы в окне мастера форм

В диалоговом окне **Новая форма** (New Form) выбираем режим создания **Мастер форм** (Form Wizard) и таблицу ГРУППА, которая будет служить источником данных для основной части, создаваемой *многотабличной формы*.

В открывшемся окне **Создание форм** в списке **Таблицы/Запросы** будет уже отражена ранее выбранная таблица ГРУППА. Выберем для нее в списке **Дос-**

тупные поля те поля, которые вошли в спроектированный макет формы, перемещая их в область **Выбранные поля**. Выберем далее таблицу СТУДЕНТ и ее поля (рис. 5.2). Эта таблица будет источником записей подчиненной формы, связанных с записью отображаемой в основной части формы.

Выбор типа формы

В следующем сеансе окна **Создание форм** отображается макет формы с перечнем полей в основной части формы и в подчиненной форме (рис. 5.3). В этом окне уже будет выделена таблица ГРУППА – источник записей основной части формы. Таблица СТУДЕНТ – источник записей подчиненной формы. Для непосредственного включения подчиненной формы выберем первый вариант – **Подчиненные формы**.

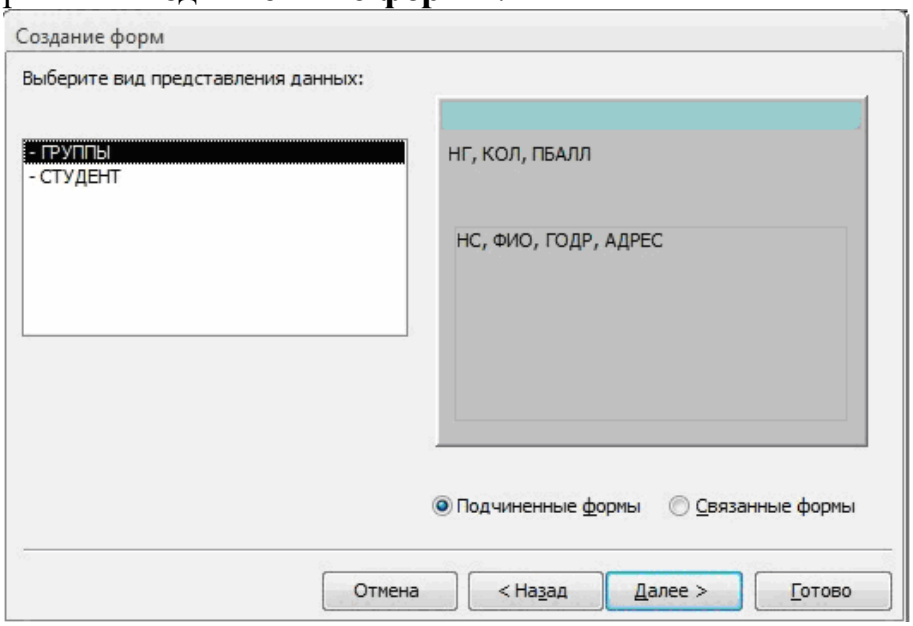


Рис. 5.3. Выбор варианта явного включения подчиненной формы

В следующем сеансе диалогового окна мастера выбираем внешний вид подчиненной формы **ленточный** для получения многозаписевой подчиненной формы и вывода в ней подписей полей (определенных в свойствах таблиц). В очередном сеансе диалогового окна выберем стиль оформления **Стандартный** с утопленными полями.

Присвоение имени форме и ее открытие

В последнем сеансе окна **Создание форм** (рис. 5.4) введем имена (заголовки) составной формы – СПИСОК ГРУППЫ и подчиненной формы – СПИСОК СТУДЕНТОВ. Выберем также дальнейшие действия мастера – **Открытие формы для просмотра и ввода данных**.

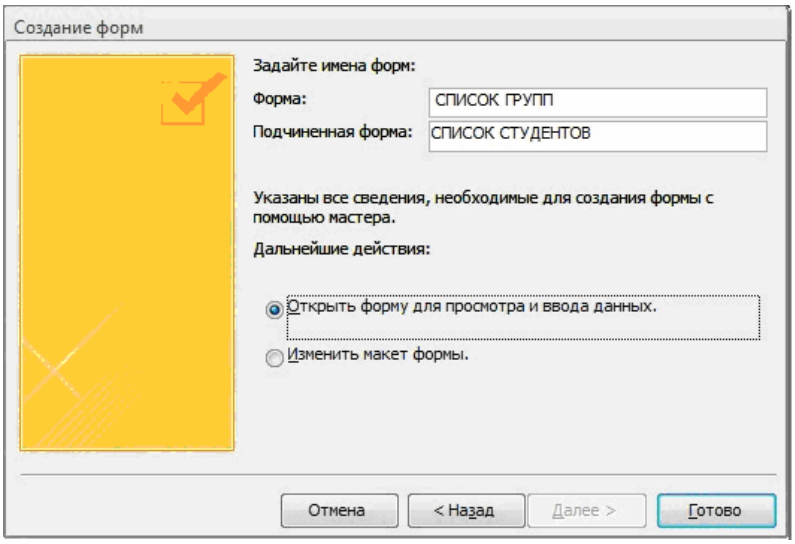


Рис. 5.4. Окно ввода имен форм и выбора дальнейших действий мастера

В соответствии с выбранными действиями после завершения работы мастера выводиться форма с данными из таблиц базы данных.

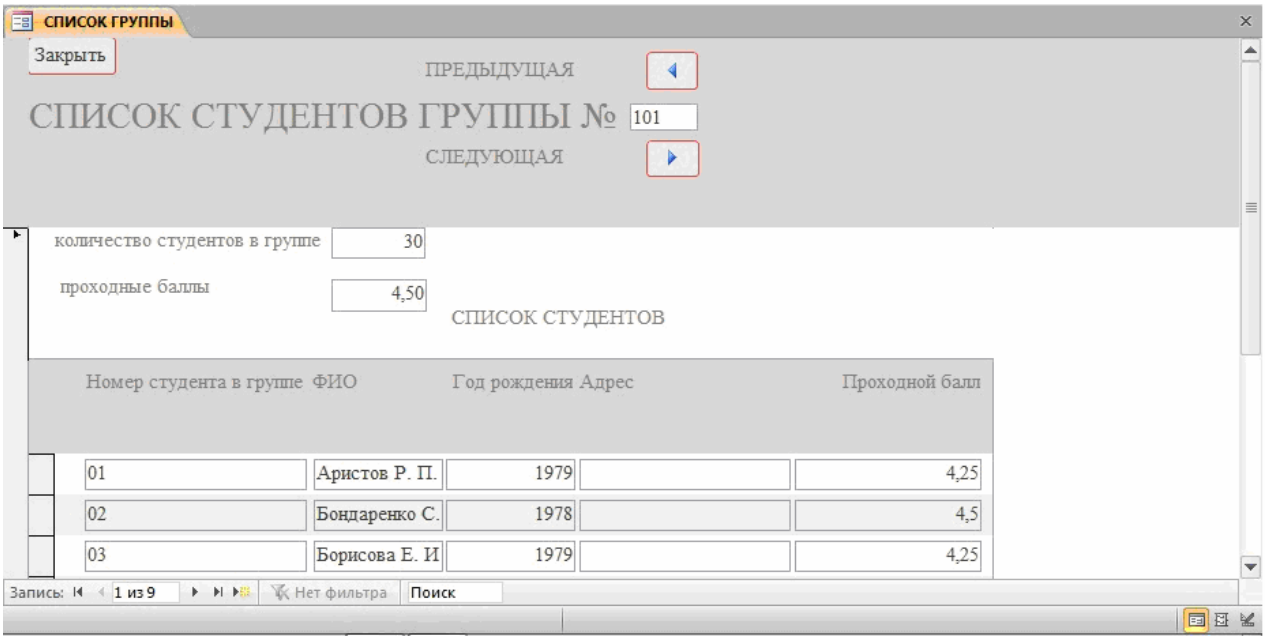


Рис. 5.5. Многотабличная форма, созданная мастером на основе таблиц ГРУППА и СТУДЕНТ

При этом в подчиненной форме выводятся те записи таблицы СТУДЕНТ, которые связаны с текущей записью таблицы ГРУППА, данные которой отображаются в основной части формы (рис. 5.5).

Замечание. При работе с формой в случае необходимости можно отобразить данные подчиненной формы в виде таблицы с именами полей таблицы БД. Для этого в режиме формы выполняется команда Вид|Таблица подчиненной формы (View|Subform Datasheet), работающая как переключатель. Курсор предварительно устанавливается на подчиненной форме.

Редактирование формы в режиме конструктора

Изменение надписей и размещения полей. В окне: **базы данных** (рис. 5.6) в области **Объекты** выделим строку **Формы**. Выберем для редактирования созданную ранее многотабличную форму СПИСОК ГРУППЫ и нажмем кнопку **Конструктор**. Если форма была открыта ранее в режиме просмотра, то для перехода в режим конструктора достаточно нажать кнопку **Вид** на панели конструктора форм .

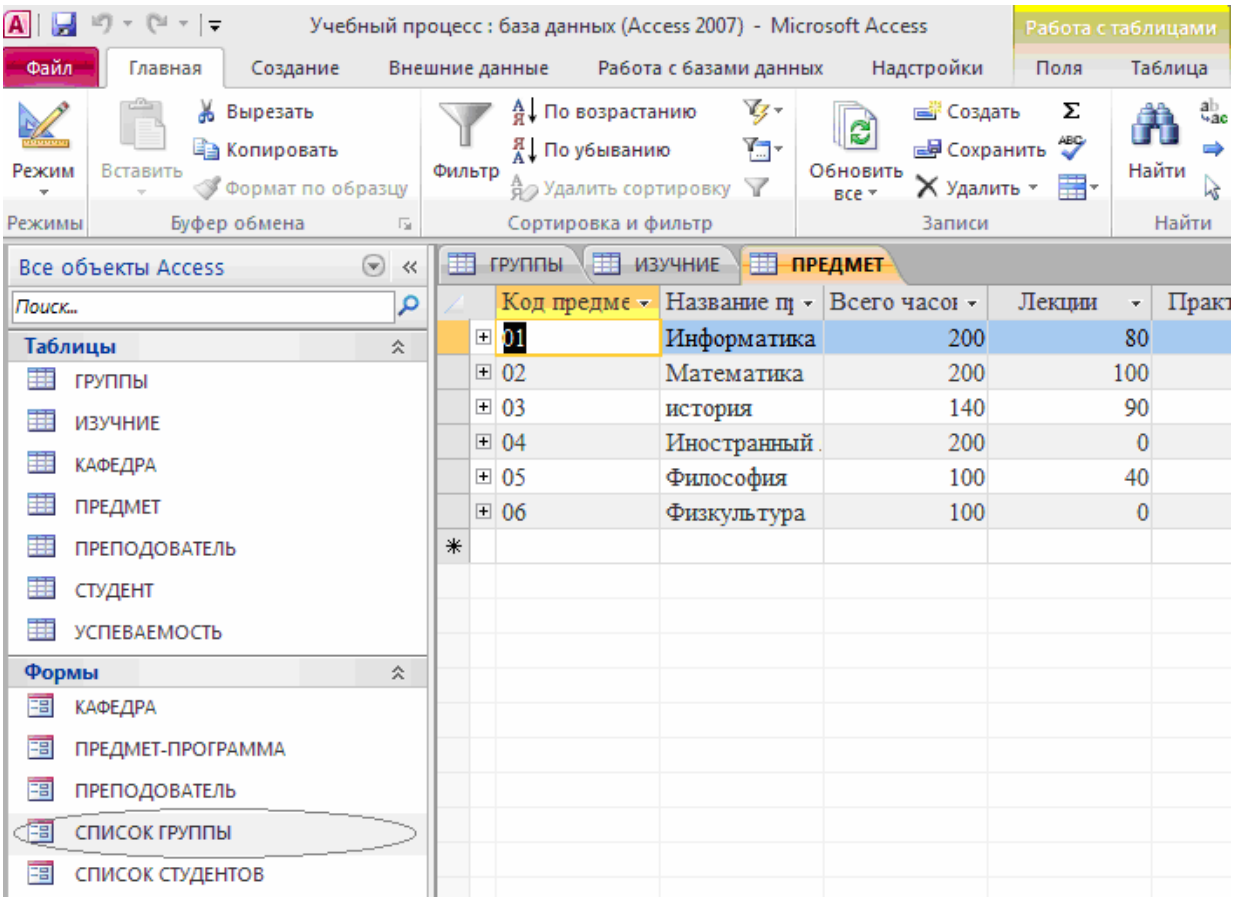


Рис. 5.6. Выбор формы в окне базы данных

В окне конструктора форм (рис. 5.7) указано имя формы, СПИСОК ГРУППЫ, представлены поля с надписями основной части формы, размещенные в области данных, а также поля и надписи подчиненной формы СПИСОК СТУДЕНТОВ в рамке, созданной для нее мастером (в версиях Access 95/97 отобразиться пустая рамка подчиненной формы, в которой приводится ее имя).

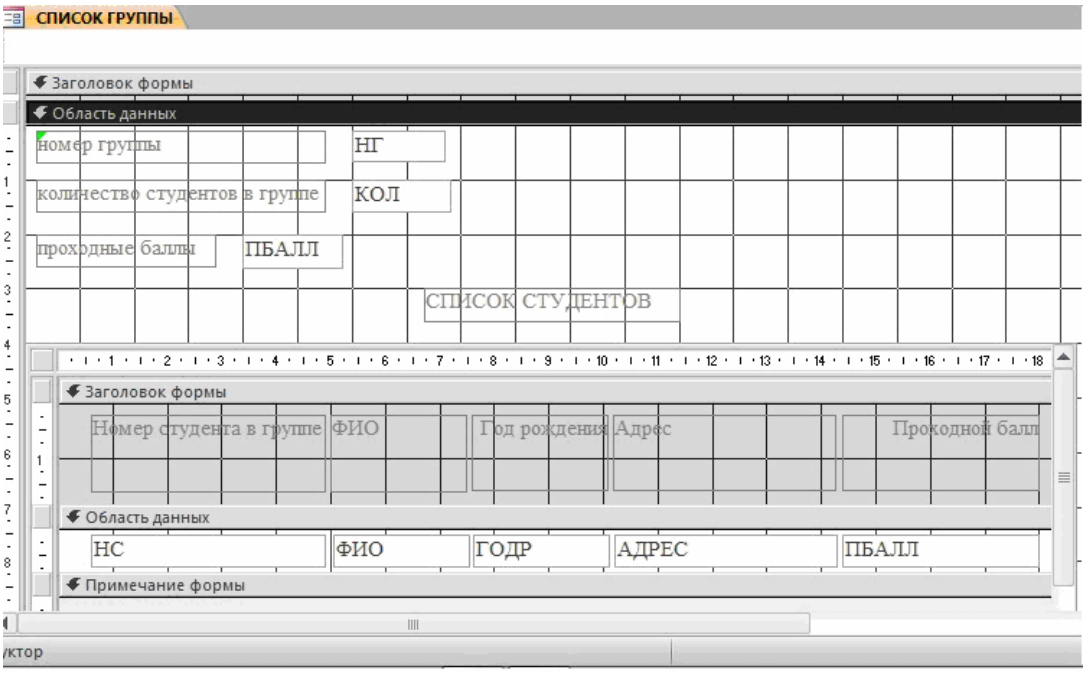


Рис. 5.7. Форма для двух таблиц ГРУППА и СТУДЕНТ в режиме конструктора

Произведем доработку формы СПИСОК ГРУППЫ, используя технику редактирования. Введем в область заголовка полное название формы СПИСОК СТУДЕНТОВ ГРУППЫ №, которое будет выводиться при распечатке формы и будет соответствовать макету документа. Формирование текста в области заголовка подробно было рассмотрено выше при конструировании однотабличной формы.

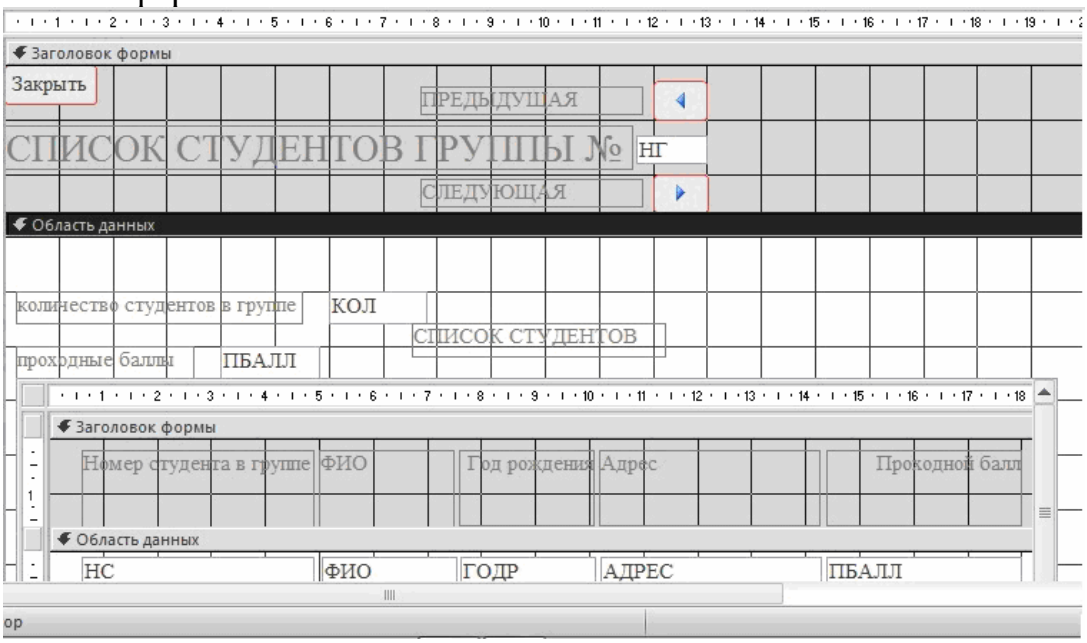


Рис. 5.8. Размещение элементов основной части формы после редактирования

Отмечая курсором мыши и перетаскивая отмеченные элементы, разместим поля так, как это показано на рис. 5.8. Уточним подписи полей, шрифт и размеры полей и подписей, заголовок формы. Можно изменить размер любого элемента, перемещая границы его рамки. Ширину и высоту подписи в соответствии с размером и шрифтом текста можно задать, используя кнопку. **По размеру данных** на панели конструктора форм. Выравнивание предварительно выделенных элементов, например, по горизонтали выполняется командой меню **Формат|Выравнять|по нижнему краю** .

Создание кнопок для перехода к другой записи

Для того чтобы в форме можно было переходить к следующей и предыдущей записей записи таблицы ГРУППА, создадим соответствующие *кнопки управления* в основной части формы.

Нажмем на **Панели элементов** Access кнопку **Мастер элементов**, а затем используем инструмент **Кнопка**. После нажатия, переноса кнопки курсором мыши в нужное место и вычерчивания ее рамки запустится мастер кнопок **Создание кнопок** (рис.5.9).

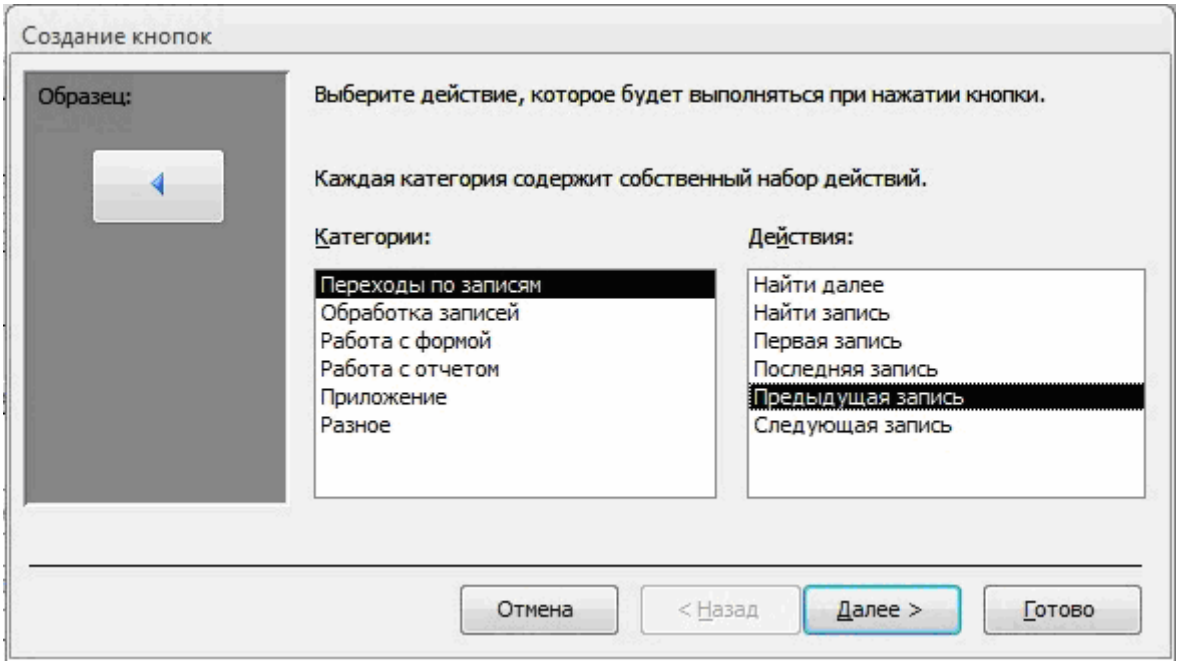


Рис.5.9. Окно мастера кнопок

В окне мастера кнопок выберем действие, которое необходимо выполнять при нажатии кнопки. В группе **Категории** выберем строку **Переходы по записям**, в группе Действия выберем строку **Предыдущая запись**. В следующем сеансе диалогового окна можно выбрать вид кнопки: **Текст** или **Рисунок** и выбрать его из списка. Отметим флажки **Рисунок** и **Показать все рисунки**. Далее выбираем подходящий рисунок из списка, например, **Стрелка вверх** (си-

няя). После нажатия кнопки **Готово** кнопка с выбранным рисунком встраивается в форму. Аналогичные действия выполняются для встраивания кнопки перехода к последующей записи таблицы. При этом выбираются, соответственно, в группе **Действия** – строку **Следующая запись** и рисунок **Стрелка вниз** (синяя). Отредактируем размер в надписи кнопок для перехода к записи другой группы, записав – «ПРЕДЫДУЩАЯ», «СЛЕДУЮЩАЯ».

Для создания кнопки закрытия формы в группе **Категории** надо выбрать строку **Работа с формой**, а в группе **Действия** – **Закрыть форму**. После формирования кнопки заменим название ее название на «ЗАКРЫТЬ».

Редактирование подчиненной формы

Аналогичные действия по доработке выполним для подчиненной формы СПИСОК СТУДЕНТОВ. Перейдем к редактированию подчиненной формы, переводя курсор в область подчиненной формы (см. рис. 5.7) или открывая подчиненную форму в окне базы данных.

Рис. 5.10. Подчиненная форма СПИСОК СТУДЕНТОВ в режиме конструктора после доработки

Используя технику редактирования формы, удалим поле НГ, отображающее номер группы, т. к. это поле является полем связи и его достаточно сохранить в основной части формы. В подчиненной форме это поле имело бы одно и то же повторяющееся значение во всех строках. Уточним подписи полей-столбцов в заголовке формы, а также шрифт, размеры полей и подписей. После редактирования формы сохраним ее, нажав кнопку панели инструментов **Сохранить**. На рис. 5.10 приведена подчиненная форма СПИСОК СТУДЕНТОВ после редактирования.

Переход в режим формы и загрузка таблиц

Завершив редактирование формы, перейдем из режима конструктора в *режим формы*, выбрав его на панели конструктора форм или выполнив команду ме-

ню **Вид|Режим формы**. На рис. 5.11 показана окончательно отредактированная форма в режиме просмотра.

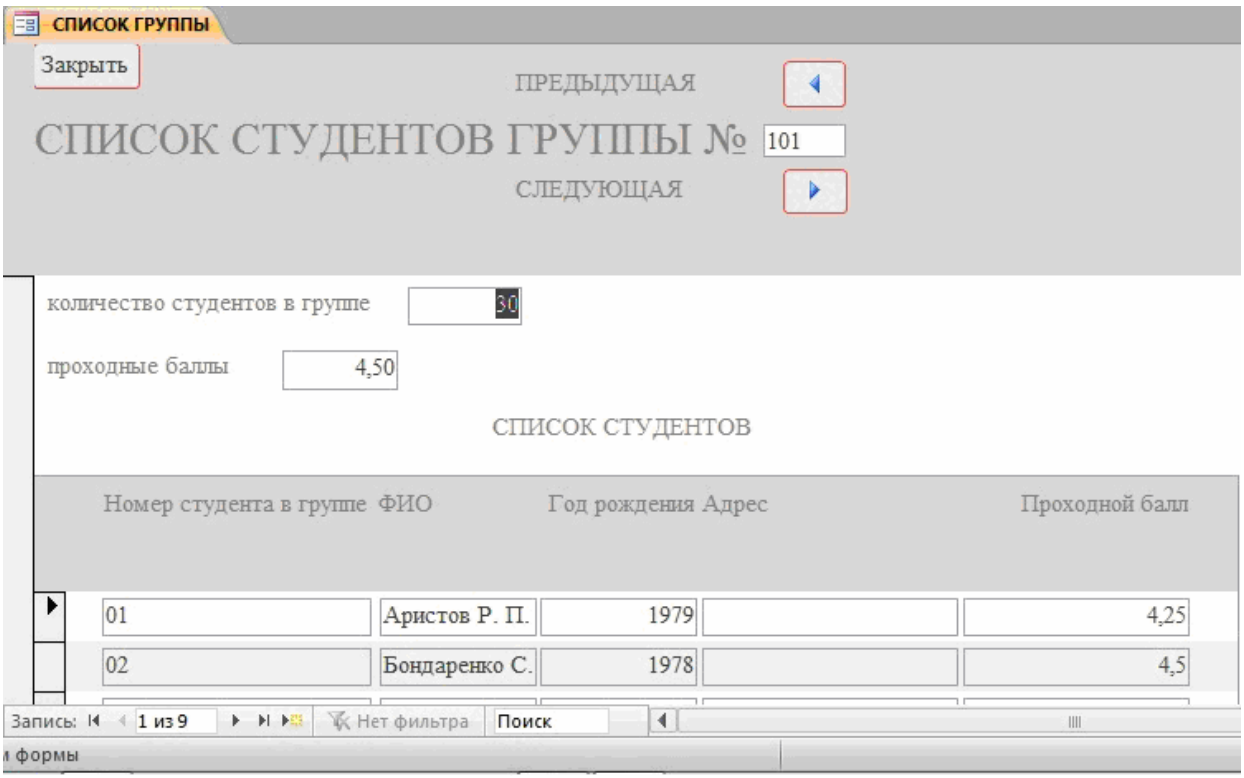


Рис. 5.11. Форма ввода-вывода для работы с данными двух взаимосвязанных таблиц

Если после редактирования форма была закрыта, то для начала сеанса работы с данными через форму необходимо в окне **Учебный процесс: база данных** (см. рис. 5.6) в группе **Объекты** перейти к строке **Формы**, выделить имя созданной многотабличной формы СПИСОК ГРУППЫ и нажать кнопку **Открыть**.

Полученная многотабличная форма СПИСОК СТУДЕНТОВ ГРУППЫ обеспечивает одновременную загрузку и работу с данными таблиц ГРУППА и СТУДЕНТ. Загрузим эти таблицы данными в соответствии с Приложением А.

В процессе загрузки сначала вводятся значения реквизитов группы: номер группы, количество студентов и средний балл группы.

Замечание. Ввод ключевого поля НГ всегда обязателен для создания записи в таблице ГРУППА. Причем эта запись создается независимо от того, будет ли введен сразу список студентов группы. Другие два реквизита группы могут вводиться и позже при просмотре данных через форму, если не введены дополнительные ограничения на их значения в свойствах при конструировании таблицы.

Далее вводятся реквизиты студентов в область подчиненной формы. При этом ввод номера студента в группе всегда обязателен для образования записи в таблице СТУДЕНТ, который вместе с введенным в основную часть формы номер группы образует уникальный ключ в этой таблице. Запись о студенте сохраняется при переходе к очередной строке в подчиненной форме.

Для перехода к записи другой группы можно использовать созданные кнопки со стрелками вверх (вниз), для перемещения по записям студентов – стандартные кнопки перехода в *поле номера записи* в нижней части подчиненной формы. Для завершения работы с формой используется созданная в форме кнопка **ЗАКРЫТЬ** или стандартной кнопкой окна в Windows.

Упражнение. Для одновременной загрузки таблиц КАФЕДРА и ПРЕПОДАВАТЕЛЬ создайте форму, показанную на рис. 5.12. При создании формы выполните действия, аналогичные рассмотренным в примере для таблиц ГРУППА и СТУДЕНТ.

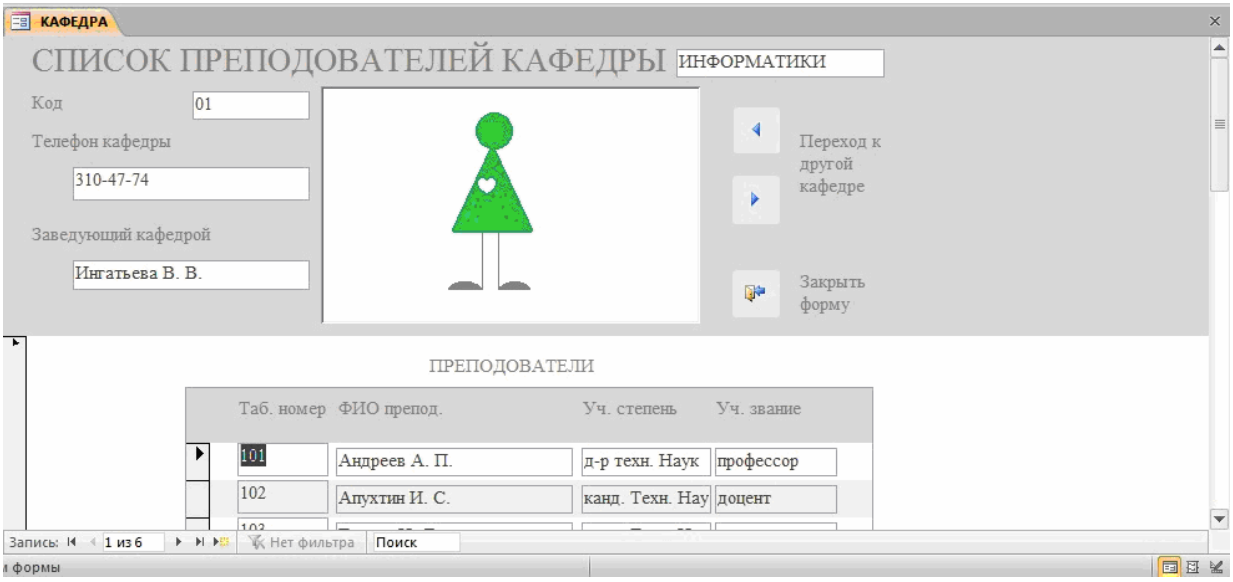


Рис. 5.12. Форма для загрузки и работы с таблицами КАФЕДРА → ПРЕПОДАВАТЕЛЬ

1. В соответствии с технологией загрузки базы данных, рассмотренной в настоящей главе, осуществите проектирование формы для загрузки данных в таблицы КАФЕДРА и ПРЕПОДАВАТЕЛЬ из документа-источника «Список преподавателей кафедры»:
   - Определите подсхему данных для составной формы

- В соответствии с подсхемой определите общую структуру составной формы

Спланируйте размещение реквизитов в макете формы так, чтобы обеспечить удобный ввод данных из документа «Список преподавателей кафедры».

2. Создайте форму средствами мастера форм и отредактируйте в конструкторе форм.
3. Загрузите данные через построенную форму в таблице КАФЕДРА И ПРЕПОДАВАТЕЛЬ. Используйте значения, приведенные в Приложении А.

Урок 6. Многотабличные формы

В соответствии с этапами загрузки базы данных "Учебный процесс", определенными выше (см. табл. 4.1), загрузка записей о занятиях текущего семестра в таблицу ИЗУЧЕНИЕ должна выполняться после загрузки таблиц со справочными данными, что обеспечит установление связей загружаемых записей с соответствующими записями этих таблиц.

На этапе проектирования определим все необходимые требования к создаваемой форме и ее макет. Процесс конструирования сложной формы средствами Access осуществим далее в соответствии с результатами этой работы.

При определении требований к форме рассмотрим особенности назначения и работы с формой, составим подсхему данных для создания формы, определим общую структуру формы и размещение реквизитов в соответствии со структурой входного документа и подсхемой данных.

Проектирование формы

Выполним проектирование формы, обеспечивающей загрузку в таблицу ИЗУЧЕНИЕ данных о занятиях текущего семестра в каждой группе, и определим требования к форме, на основе которых можно перейти к ее созданию.

Документом внемашинной сферы, содержащим необходимые данные для загрузки таблицы ИЗУЧЕНИЕ, может служить "План проведения занятий в группе". Поэтому загружаемые через форму записи целесообразно группировать в соответствии с их подчиненностью записям другой таблицы - ГРУППА.

В таблицу ИЗУЧЕНИЕ в соответствии с ее структурой наряду с номером группы и видом занятия нужно вводить идентификаторы предмета и преподавателя. В документе-источнике указаны наименование предмета и фамилия преподавателя. Для того чтобы при загрузке правильно вводить только идентификаторы, предусмотрим отображение в форме расшифровывающей информации: *наименования предмета* (НП) и *фамилии преподавателя* (ФИО) из таблиц ПРЕДМЕТ и ПРЕПОДАВАТЕЛЬ, которым подчинена загружаемая таблица.

Поскольку форма служит не только для загрузки, но и для просмотра, включим в форму и другие описательные реквизиты из таблиц ПРЕДМЕТ и ПРЕПОДАВАТЕЛЬ для их отображения.

Определение подсхемы данных

В результате загрузки данных о занятиях в группах должны формироваться только записи таблицы ИЗУЧЕНИЕ *(объект загрузки),* которую надо включить в подсхему данных для формы ввода-вывода данных о занятиях в группах (см. рис 6.1).

Поскольку загрузку и просмотр данных удобно производить по каждой группе в отдельности, в подсхему включим таблицу ГРУППА. Выберем по этой причине таблицу ГРУППА в качестве *источника основной части составной формы*. Заметим, что данные таблицы ГРУППА должны только отображаться в форме, но не вводить в нее. В подсхему (рис. 6.1) включим также таблицы ПРЕДМЕТ и ПРЕПОДАВАТЕЛЬ, т.к. в форме необходимо предусмотреть отображение описательных данных о предмете и преподавателе.

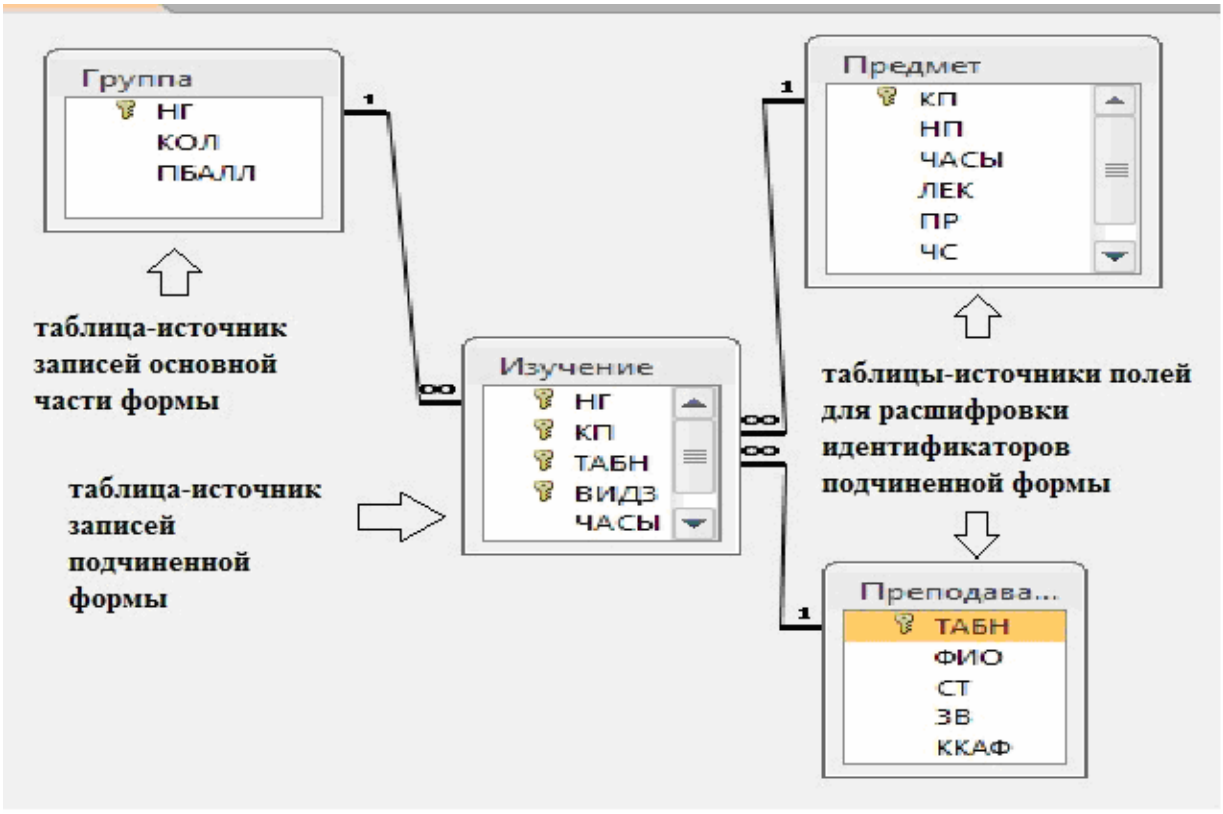


Рис. 6.1. Подсхема для формы ввода-вывода данных в таблицу ИЗУЧЕНИЕ

Определение общей структуры формы

На основе полученной подсхемы (рис. 6.1) определим общую структуру формы, которую назовем ПЛАН ЗАНЯТИЙ.

Основная часть формы. Многотабличная форма ПЛАН ЗАНЯТИЙ будет содержать основную часть на основе таблицы ГРУППА для группировки вводимых данных о занятиях по каждой группе студентов. Для ввода данных в таблицу ИЗУЧЕНИЕ предусмотрим *непосредственное включение подчиненной формы* ИЗУЧЕНИЕ.
Таким образом, форму ПЛАН ЗАНЯТИЙ определяют:

- *Тип формы* – многотабличная
- *Источник записей* для основной части формы – таблица ГРУППА
- Включаемая подчиненная форма ИЗУЧЕНИЕ с *источником записей* - таблица ИЗУЧЕНИЕ

Подчиненная форма, включаемая в основную. Для расшифровки идентификаторов предмета и преподавателя предусмотрим включение в подчиненную форму ИЗУЧЕНИЕ полей из таблиц ПРЕДМЕТ и ПРЕПОДАВАТЕЛЬ, являющихся главными относительно таблицы, на которой строится форма ИЗУЧЕНИЕ.
Таким образом, подчиненную форму ИЗУЧЕНИЕ определяют:

- *Тип формы – подчиненная*
- *Источник записей для основной части формы ИЗУЧЕНИЕ – таблица ИЗУЧЕНИЕ*
- *Источники отображаемых полей формы – таблицы ПРЕДМЕТ и ПРЕПОДАВАТЕЛЕЙ*

Размещение реквизитов формы

Основная часть формы. В основной части составной формы ПЛАН ЗАНЯТИЙ вверху будем размещать поля таблицы ГРУППА:

- *НГ – номер группы (уникальный ключ)*
- *КОЛ – количество студентов*
- *ПБАЛЛ – средний балл в группе при поступлении*

Доступ к перечисленным полям должен быть ограничен *только чтением*, т.к. значения этих полей не должны вводиться и корректироваться из документа «План занятий».

Подчиненная форма ИЗУЧЕНИЕ. В подчиненной форме ИЗУЧЕНИЕ разместим:
Все поля загружаемой таблицы ИЗУЧЕНИЕ, кроме ключевого реквизита НГ (номер группы), включенного в основную часть формы, что обеспечивает однократное отображение одинаковых номеров группы по форме :

- *КП – код предмета*
- *ТАБН – идентификатор преподавателя*
- *ВИДЗ – вид занятий*
- *ЧАСЫ – число часов занятий*
- *СБАЛЛ-ГР – средний балл по предмету в груп*пе

Поля из таблиц ПРЕДМЕТ и ПРЕПОДАВАТЕЛЬ, позволяющие отобразить справочную информацию о предмете и преподавателе, ведущем занятие:
поля таблицы ПРЕДМЕТ:

- *НП – название предмета*
- *ЧАСЫ – всего часов*
- *ЛЕК – часов лекций*
- *ПР – часов практики*

Поля таблицы ПРЕПОДАВАТЕЛЬ:

- *ФИО – фамилия преподавателя*
- *СТ – ученая степень*
- *УЗ – ученое звание*

Создание многотабличной формы с подчиненной формой средствами мастера

Используя мастера форм, создадим в соответствии с результатами проектирования составную форму ПЛАН ЗАНЯТИЙ, включающую подчиненную форму ИЗУЧЕНИЕ.

В окне базы данных выберем вкладку **Создание,** в группе **Формы** нажмем кнопку **Мастер форм**. В качестве таблицы, которая будет источником записей основной части сложной формы, выберем таблицу ГРУППА.

В первом сеансе диалоговом окна **Создание форм** выберем включаемые в форму поля таблицы ГРУППА, а затем поля таблиц ИЗУЧЕНИЕ, ПРЕДМЕТ, ПРЕПОДАВАТЕЛЬ. В следующем окне (рис. 6.2) уже будет выделена таблица ГРУППА, которая выбрана для создания основной части формы. Поскольку эта таблица является главной в схеме данных относительно используемой в

форме таблицы ИЗУЧЕНИЕ, представляется возможность выбрать тип формы – Подчиненные формы.

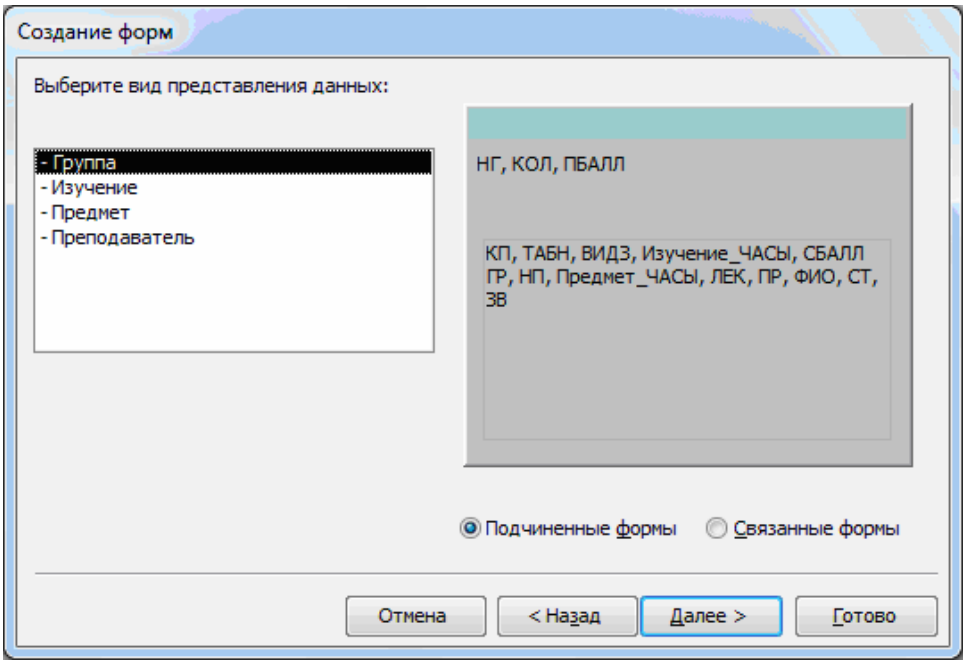


Рис. 6.2. Окно мастера форм при выборе типа включения подчиненной формы

Для получения многозаписевой подчиненной формы выберем в следующем сеансе окна мастера вид формы **Ленточный** (Tabular) и стиль оформления **Стандартный** (Standard). В последнем сеансе окна **Создание форм** (Form Wizard) зададим имя составной формы – ПЛАН ЗАНЯТИЙ, а также имя подчиненной формы – ИЗУЧЕНИЕ. Выберем также дальнейшие действия мастера – **Открыть форму для просмотра и ввода данных** (Open the form to view or enter information).

По завершению работы мастера выводится форма с данными из таблиц, которые были заданы пользователем в диалоге с мастером. В подчиненной форме выводятся все записи о занятиях для одной группы (рис. 6.3). При этом запись о занятии содержит, кроме полей из таблицы ИЗУЧЕНИЕ, также поля из главных по отношению к ней таблиц ПРЕДМЕТ и ПРЕПОДАВАТЕЛЬ. Мастер, проанализировав схему данных, сам формирует подчиненную форму ИЗУЧЕНИЕ, и включает поля этих таблиц в качестве расшифровывающих.

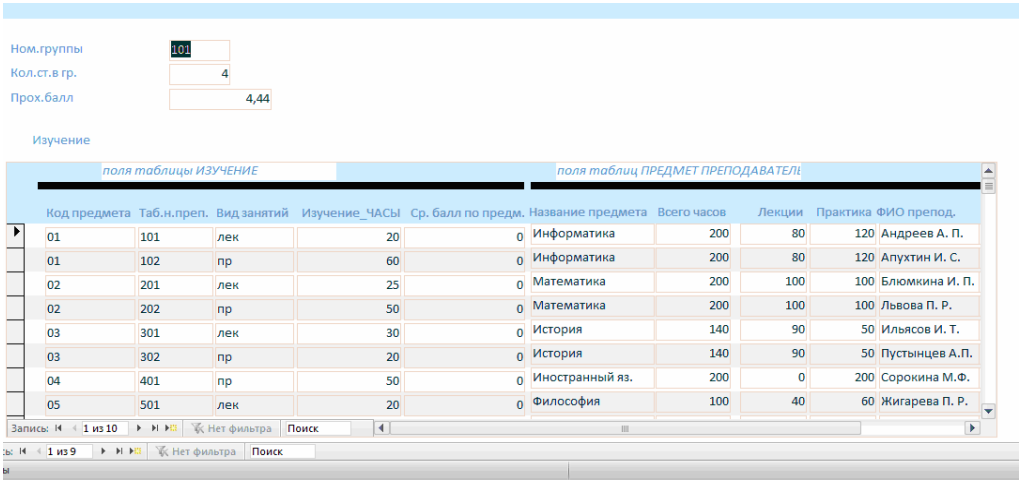


Рис. 6.3. Форма с подчиненной формой, созданная мастером на основе четырех таблиц

Заметим, что запись в подчиненной форме имеет очень много полей, причем лишь часть этих полей служит для загрузки данных в подчиненную таблицу ИЗУЧЕНИЕ. Другие поля должны отображаться из ранее загруженных таблиц ПРЕДМЕТ, ПРЕПОДАВАТЕЛЬ. Такое размещение полей неудобно для работы пользователя. Форму целесообразно доработать средствами конструктора.

Разработка формы, обеспечивающей удобный интерфейс пользователя для загрузки подчиненной таблицы

Рассмотренные выше два варианта построения формы для загрузки данных о занятиях недостаточно удобны для работы пользователя. В первом варианте данные о каждом занятии были представлены компактно в виде *одиночной записи,* но не обеспечивался удобный доступ к занятиям каждой группы. Вариант формы с подчиненной таблицей также имел недостатки, поскольку данные, вводимые в таблицу ИЗУЧЕНИЕ, входили в одну длинную запись вместе с отображаемыми справочными данными.

Выполним разработку формы, более удобной пользователю для просмотра занятий группы и загрузки новых занятий, и защитим в ней поля справочных данных.

Требования к создаваемой форме. Обеспечим возможность компактного отображения на экране всей информации об одном занятии и сохраним в то же время возможность объединения записей о занятиях по группам, как в форме на рис. 6.3. Кроме того, целесообразно в форме в одну группу объединить поля, в которые вводятся значения при загрузке таблицы ИЗУЧЕНИЕ, а в другую - поля, которые содержат только справочную информацию о предмете

и преподавателе, которая отображается для расшифровки идентификаторов занятия.

В процессе конструирования обеспечим защиту справочных данных в таблицах ГРУППА, ПРЕДМЕТ и ПРЕПОДАВАТЕЛЬ от случайных изменений при загрузке данных о занятиях в таблицу ИЗУЧЕНИЕ. Предусмотрим для удобства пользователя кнопки перехода к просмотру занятий для другой группы и кнопку закрытия формы. Для визуальной проверки правильности вводимых идентификаторов преподавателя и предмета используем поля со списком.

В соответствии с перечисленными требованиями для первоначального размещения полей и создания подчиненной формы можно воспользоваться формой ПЛАН ЗАНЯТИЙ, полученной мастером (см. рис. 6.3). Откроем эту форму в режиме конструктора. Для этого в **Границе области переходов (область навигации)** выберем для редактирования созданную ранее многотабличную форму ПЛАН ЗАНЯТИЙ. Если форма была открыта ранее в режиме просмотра или в режиме макета, то для перехода в режим конструктора достаточно выбрать соответствующий тип представления в группе **Режимы** на вкладке **Главная.**

Редактирование основной части формы. В основной части формы разместим и отредактируем поля таблицы ГРУППА так, как это показано на рис. 6.4. Уточним текст подписей полей, шрифт и размеры полей и подписей, введем текст в заголовок формы. Удалим элемент с подписью подчиненной формы. Удалим разделительные линии между разделами формы: заголовком, областью данных и примечания. Для этого в свойствах формы на вкладке **Макет** в строке **Разделительные липни** выберем **Нет**. Уберем область выделения записи, проставив в свойствах формы в соответствующей строке "Нет" Создадим две кнопки для перехода к следующей или предыдущей группе, а также кнопку для закрытия формы.

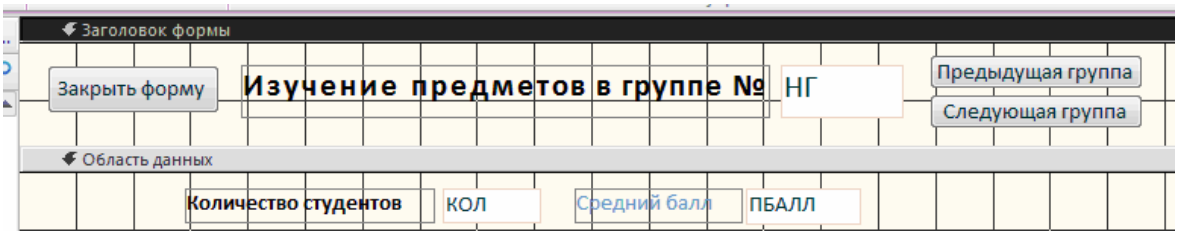


Рис. 6.4. Основная часть многотабличной формы после редактирования в режиме конструктора

Ограничение доступа к полям таблицы-источника основной части формы. Защитим данные записей таблицы ГРУППА от непроизвольных измене-

ний при работе с формой, т.к. они должны использоваться только для отображения. Это все поля основной части формы. Для защиты поля выделим рамку поля и с помощью контекстно-зависимого меню вызовем свойства поля. В окне свойств на вкладке **Данные** в строке **Блокировка** выберем **Да**. После установки этого свойства поле доступно только для чтения.

Сохраним форму под новым именем –«ПЛАН ЗАНЯТИЙ В ГРУППЕ», используя команду **Сохранить как**.

Редактирование подчиненной формы ИЗУЧЕНИЕ.

Ранее мастером было получена подчиненная ленточная форма, которая в режиме конструктора показана на рис. 6.5.

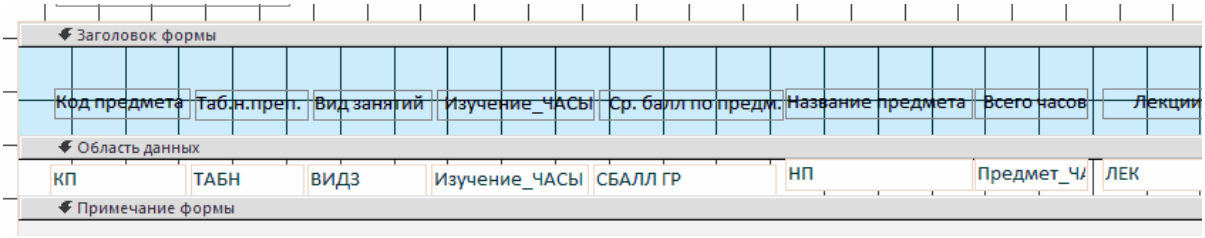


Рис. 6.5. Подчиненная форма ИЗУЧЕНИЕ в режиме конструктора

Для изменения вида подчиненной формы вызовем ее свойства. На вкладке **Макет** в окне **Окно свойств** (прав.кнопка мыши - пункт «свойства формы») (рис. 6.6) заменим в строке **Режим по умолчанию** значение **Ленточная форма** на **Простая форма**. Это позволит отображать в подчиненной форме одну запись о занятии. Вид формы в конструкторе останется прежним.

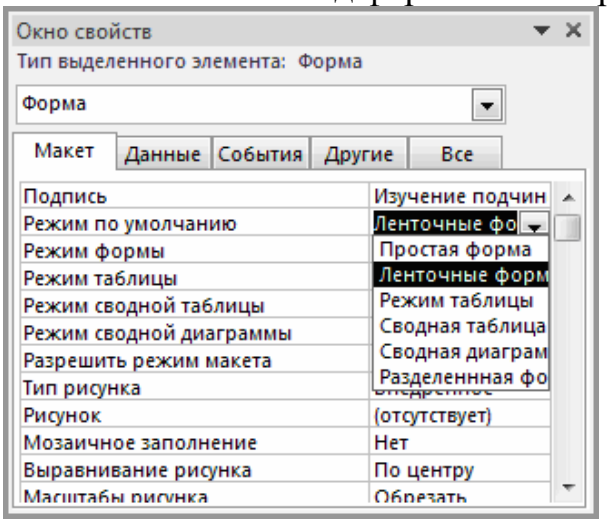


Рис. 6.6. Отображение списка режимов по умолчанию в окне свойств подчиненной формы

В подчиненной форме разместим поля так, как это показано на рис. 6.7. После перемещения всех подписей полей из заголовка в область данных можно сократить его размер до нуля перемещением границы заголовка и области данных. Поля таблицы ИЗУЧЕНИЕ, в которой надо вводить данные из документа «План занятий», разместим в верхней части области данных. В нижней части области данных разместим поля, в которые будут автоматически выводиться справочные данные из таблиц ПРЕДМЕТ и ПРЕПОДАВАТЕЛЬ для расшифровки вводимых идентификаторов занятия. Эти поля служат только для отображения сведений о предмете и преподавателе. Для создания рамок используем кнопку панели элементов управления **Прямоугольник** .

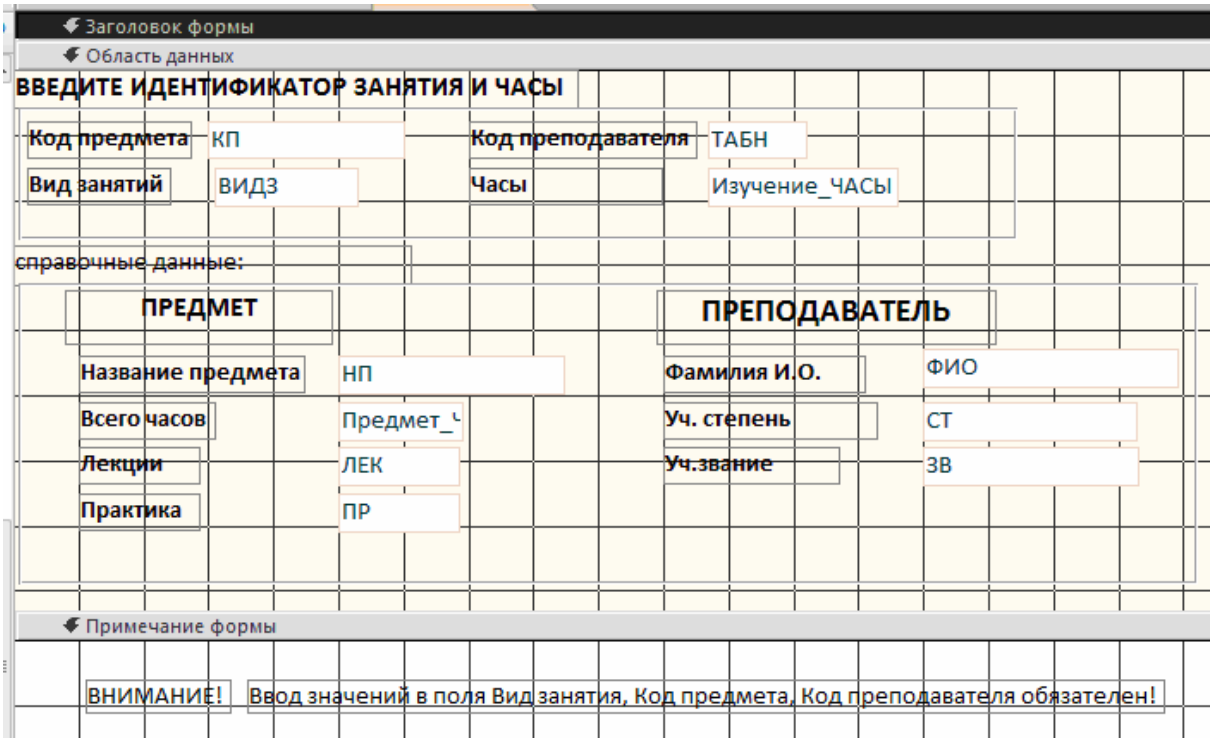


Рис. 6.7. Подчиненная форма ИЗУЧЕНИЕ в режиме конструктора форм после редактирования

Уточним текст подписей полей, шрифт и размеры полей и подписей. Выполним относительное выравнивание надписей и полей с помощью команды **Главная|Форматирование текста|Выровнять.**

В область примечаний формы введем инструкцию пользователю, требующую обязательного ввода данных в поля, идентифицирующие занятие: код предмета — КП, номер преподавателя — ТАБН и вид занятия — ВИДЗ. Без этого не может быть создана запись в таблице ИЗУЧЕНИЕ.

Защита справочных данных от изменений. Защитим поля НП, ЧАСЫ, ЛЕК, таблицы ПРЕДМЕТ и поля ФИО, СТ, ЗВ таблицы ПРЕПОДАВАТЕЛЬ от случайных изменений при работе с формой. Для зашиты поля выделим рамку поля и с помощью контекстно-зависимого меню вызовем свойства поля. В окне свойств на вкладке **Данные** в строке **Блокировка** выберем **Да**. После установки этого свойства поле доступно только для чтения.
Для визуального контроля правильности ввода идентификаторов занятия: КП и ТАБН можно использовать **Поле со списком**. Процесс создания такого поля рассматривается ниже.

Создание полей со списком

При вводе идентификационных данных через форму в Access имеется возможность получить справочную информацию из ранее загруженных таблиц, что позволяет выбрать уже имеющиеся значения в базе и тем самым повысить достоверность вводимой информации. Отображение данных из справочных таблиц при вводе идентификатора свидетельствует о наличии в базе данных главных записей для загружаемой подчиненной записи, что необходимо для успешного завершения ввода при установленном параметре целостности в схеме данных.

Поле со списком объединяет поле формы, в которое нужно ввести данные, и список. Список содержит записи из связанной главной таблицы. В списке можно выбрать из соответствующего поля нужное значение и ввести его в поле формы (см., пример, на рис. 6.8).

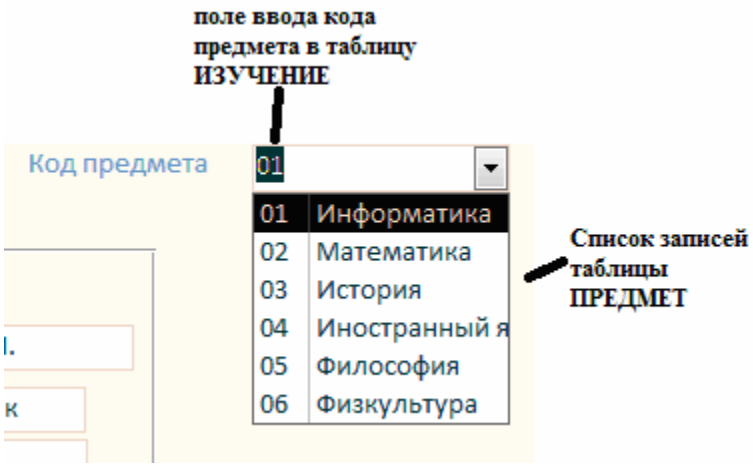


Рис. 6.8. Отображение списка

Создание поля со списком с помощью мастера

Создадим поле со списком для ввода значений кода предмета – КП в таб- лицу ИЗУЧЕНИЕ. Это позволит просматривать и вводить значения, кото- рые уже имеются в главной таблице ПРЕДМЕТ, а также проверять соответ-ствие кода и наименования предмета, имеющихся в документе-источнике загрузки «План занятий».

Выберем кнопку **Поле со списком** в группе **Элементы управления** на вкладке **Конструктор,** установим курсор мыши в нужное место, нажмем кнопку мыши и, не отпуская ее, вы-чертим рамку элемента. После отпускания кнопки мыши запустится мастер и откроет диалоговое окно **Создание полей со списком.** В этом окне определим способ, которым список поля получает свои значения. Для формирования списка из связанной таблицы выберем вариант **Таблица или запрос содержат значения, которые использует поле со списком** (рис 6.9).

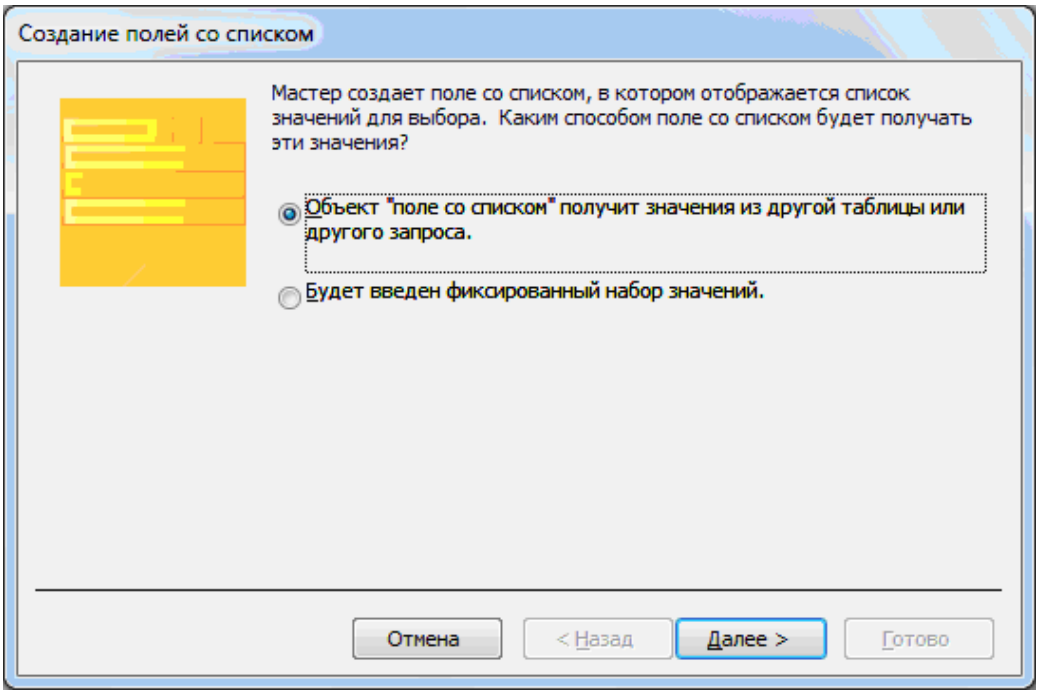


Рис. 6.9. Выбор способа форматирования списка

В следующем окне выберем таблицу ПРЕДМЕТ, которая будет поставлять значения в список поля (рис 6.10).

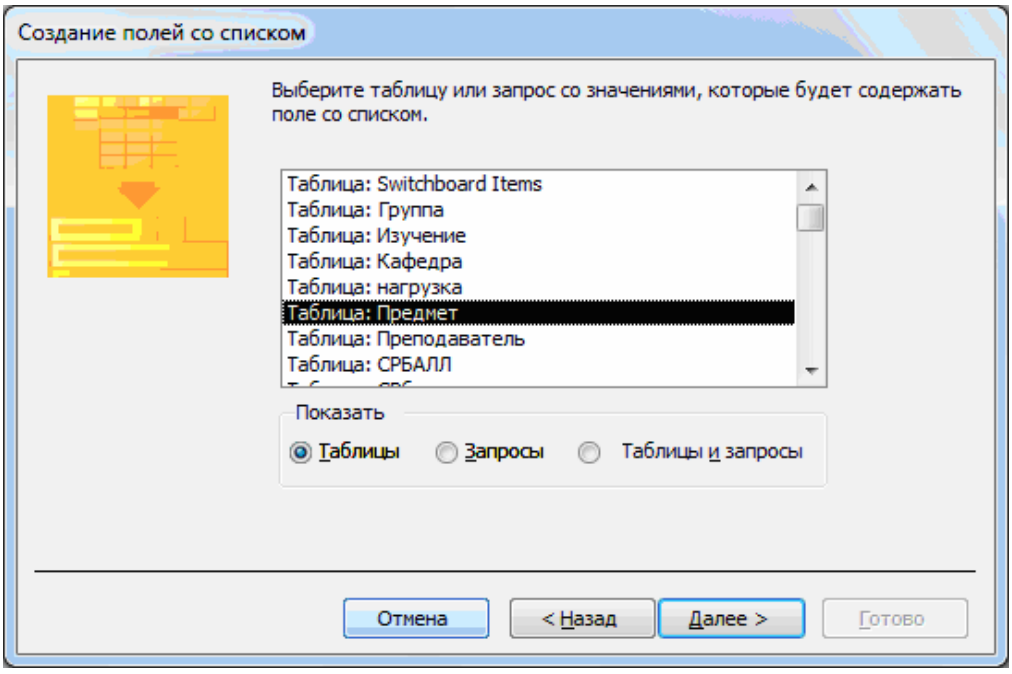


Рис. 6.10. Выбор источника данных для формирования списка

Затем выберем поле КП, а также поле НП для расшифровки кода КП (рис. 6.11). Эти поля образуют записи списка.

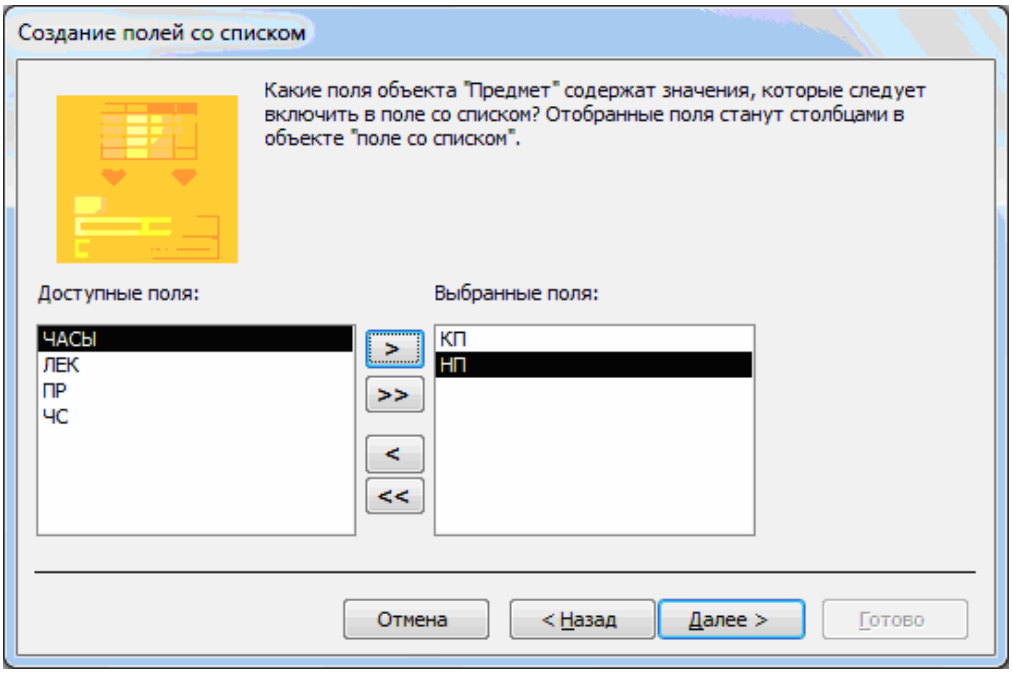


Рис. 6.11. Выбор полей для формирования записи списка

Далее в появившейся таблице определим ширину столбцов списка в соответствии с размером значений. Для этого курсор мыши установим на ли- нию, разделяющую имена столбцов, и переместим ее в нужное место (рис. 6.12).

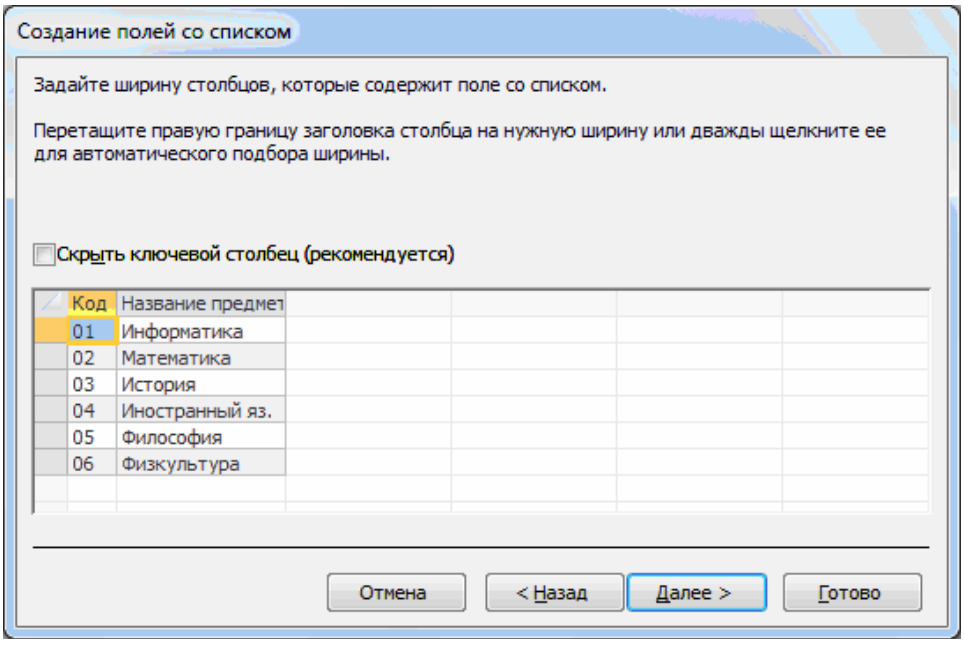


Рис. 6.12. Окно для оформления столбцов списка

Далее выберем поле списка КП, являющееся ключом связанной таблицы ПРЕДМЕТ. Из этого поля будет выбираться значение для ввода в поле фор мы (рис. 6.13).

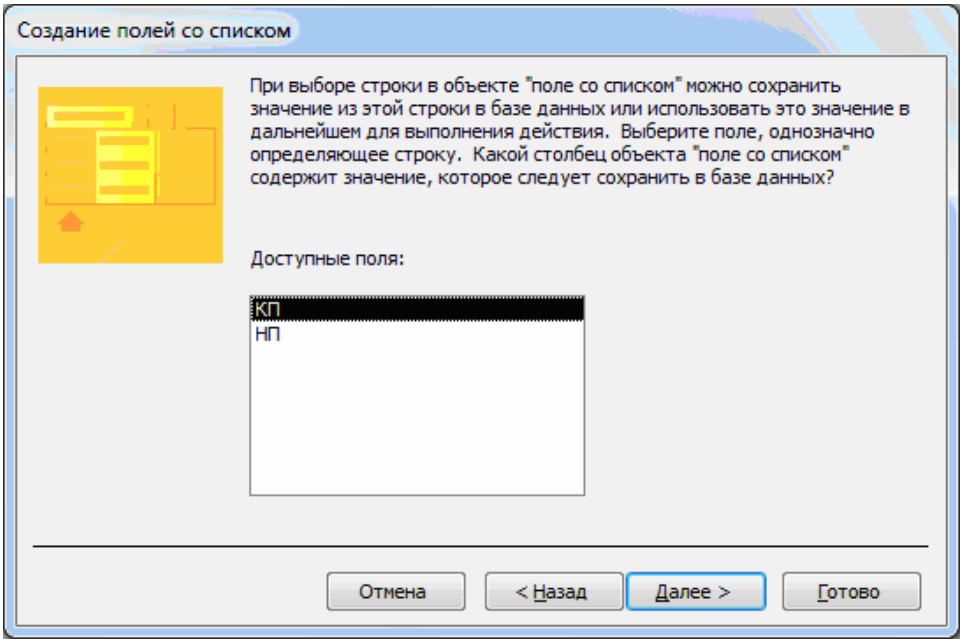


Рис. 6.13. Окно для выбора поля источника значений для поля формы

В следующем окне отметим переключатель **Сохранить в поле** и выберем поле формы КП (поле таблицы ИЗУЧЕНИЕ), в кото-рое будут вводиться значения из списка (рис. 6.14).

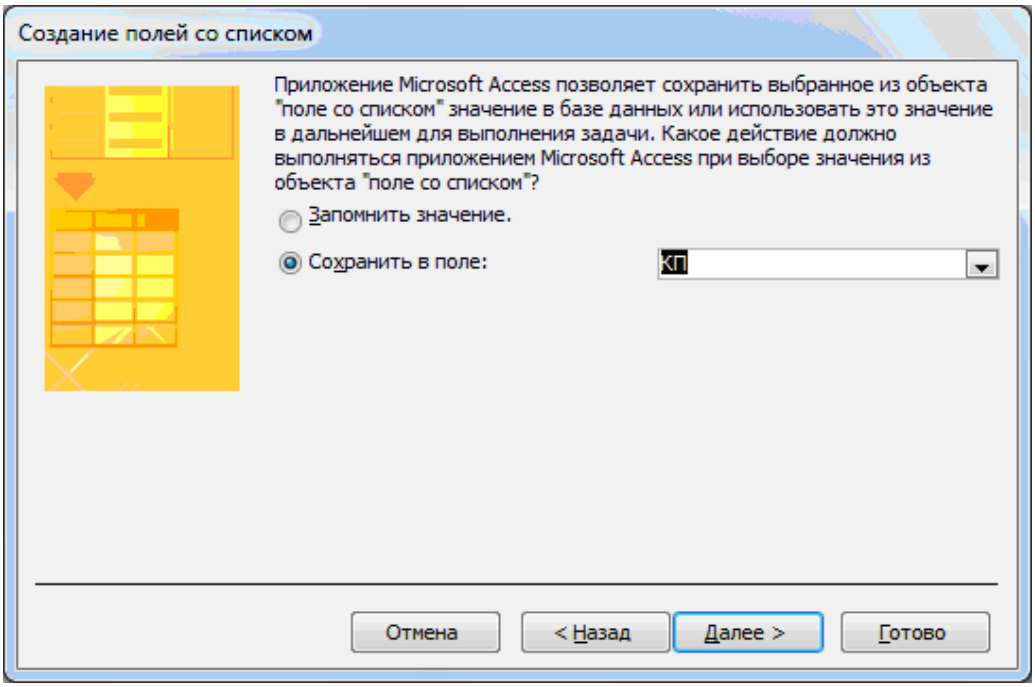


Рис. 6.14. Выбор поля формы, в которое вводится значение из списка

Далее введем подпись поля со списком – «Код предмета» (рис. 6.15).

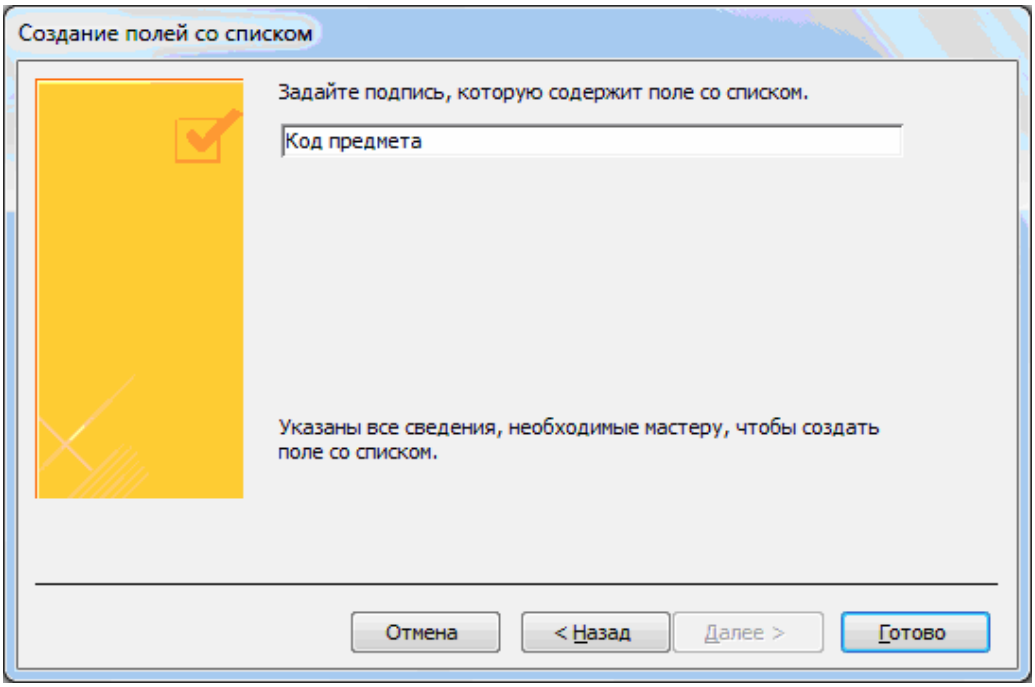


Рис. 6.15. Определение подписи поля со списком

Нажмем кнопку **Готово**. В результате получим поле КП со списком, которое содержится в окончательной форме, представленной на рис. 6.18. Использование поля со списком предметов возможно в режиме формы. Для удобства поиска нужного значенияв списке можно воспользоваться

операциями поиска и сортировки. Доступ к этим операциям возможен при помощи кнопок **Найти (**группа **Найти**), **Сортировка по возрастанию**, **Сортировка по убыванию** группы **Сортировка и фильтр**.

Создание поля со списком без использования мастера

Создадим поле со списком для ввода значений идентификатора преподавателя ТАБН. Это позволит просматривать и вводить значения, которые уже имеются в главной таблице ПРЕПОДАВАТЕЛЬ, а также проверять соответствие номера и фамилии преподавателя, имеющихся в документе-источнике загрузки.

Нажмем на вкладке **Конструктор** в группе **Элементы управления** кнопку **Поле со списком**.

Установим курсор мыши на появившийся элемент **Свободный** (рис. 6.16).

Рис. 6.16. Элемент формы **ПолеСоСписком**

Нажмем правую кнопку мыши, чтобы вызвать контексно-зависимое меню. Выберем пункт **Свойства**, затем – вкладку **Данные**. В стро-ке **Данные** выберем поле ТАБН, которое необходимо запол-нять через форму в таблице ИЗУЧЕНИЕ (рис. 6.17). В строке **Тип источника строк** выберем элемент **Таблица|Запрос**, а в строке **Источник строк** – таблицу ПРЕПОДАВАТЕЛЬ. Поля, включаемые в список, и их порядок определяются в построителе, который вызывается в этой же строке нажатием кнопки ... .

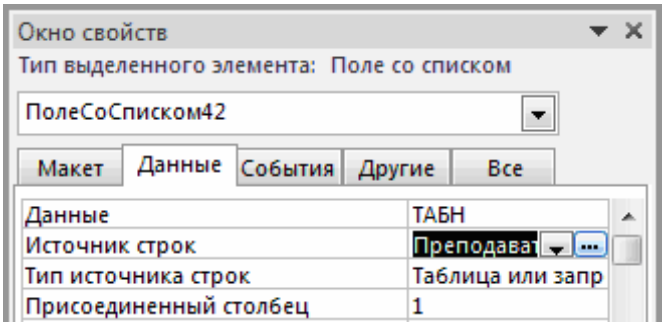


Рис. 6.17. Окно определения свойств данных поля со списком

Построитель выводит бланк запросов, в который перетащим из таблицы ПРЕПОДАВАТЕЛЬ поля ТАБН и ФИО.

Для того чтобы в списке выводились два поля, на вкладке **Макет** необходимо в строке **Число столбцов** указать «2». Для настройки ширины столбцов спмска в строках **Ширина списка** и **Ширина столбцов** зададим подходящие значения. Закроем окно свойств **Поле со списком**. В результате получим поле ТАБН со списком, которое содержится в окончательной форме, представленной на рис. 6.18.

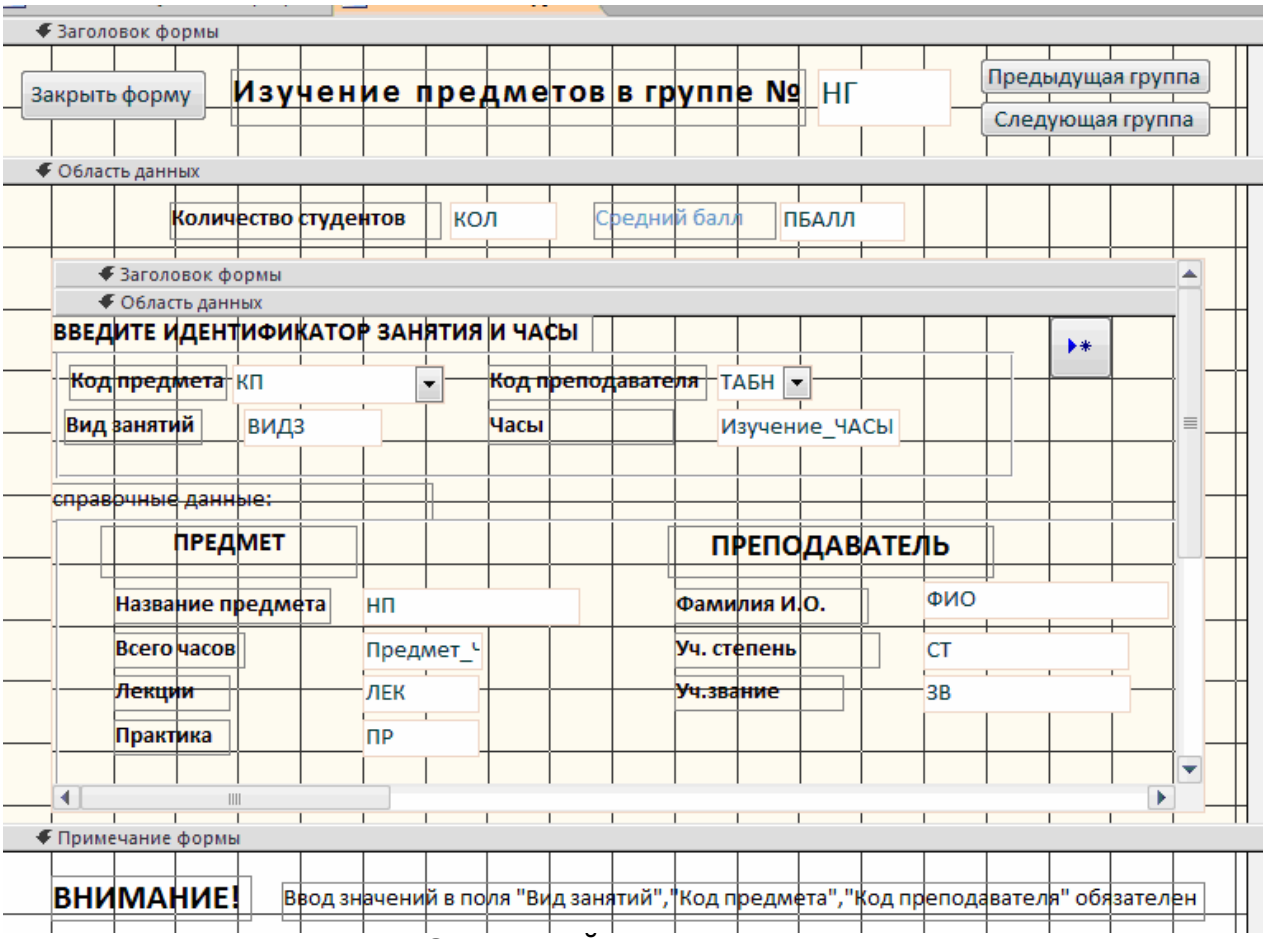


Рис. 6.18. Форма ПЛАН ЗАНЯТИЙ в режиме конструктора

Замечание Если необходимо преобразовать обычное поле в поле со списком , надо в контекстно-зависимом меню поля выбрать **Преобразовать элемент в** и далее выбрать строку **Поле со списком** При включенной кнопке **Мастера элементов** элемент будет преобразован мастером. При выключенной кнопке необходимо для получения поля со списком установить свойства этого поля, как описано выше.

Загрузка данных в подчиненную таблицу через форму

Для загрузки данных в подчиненную таблицу ИЗУЧЕНИЕ через форму ПЛАН ЗАНЯТИЙ можно сразу перейти из режима конструктора в **Режим формы.**

Для этого на вкладке **Главная** в списке кнопки **Режим** выбирается **Режим формы**. Если форма была закрыта, необходимо в окне **Область навигации** выбрать форму ПЛАН ЗАНЯТИЙ. На рис. 6.19 приводится форма ПЛАН ЗАНЯТИЙ в режиме формы, обеспечивающая загрузку и корректировку данных о занятиях в таблице ИЗУЧЕНИЕ, подчиненной таблицам ГРУППА, ПРЕДМЕТ, ПРЕПОДАВАТЕЛЬ, данные из которых только отображаются в форме.

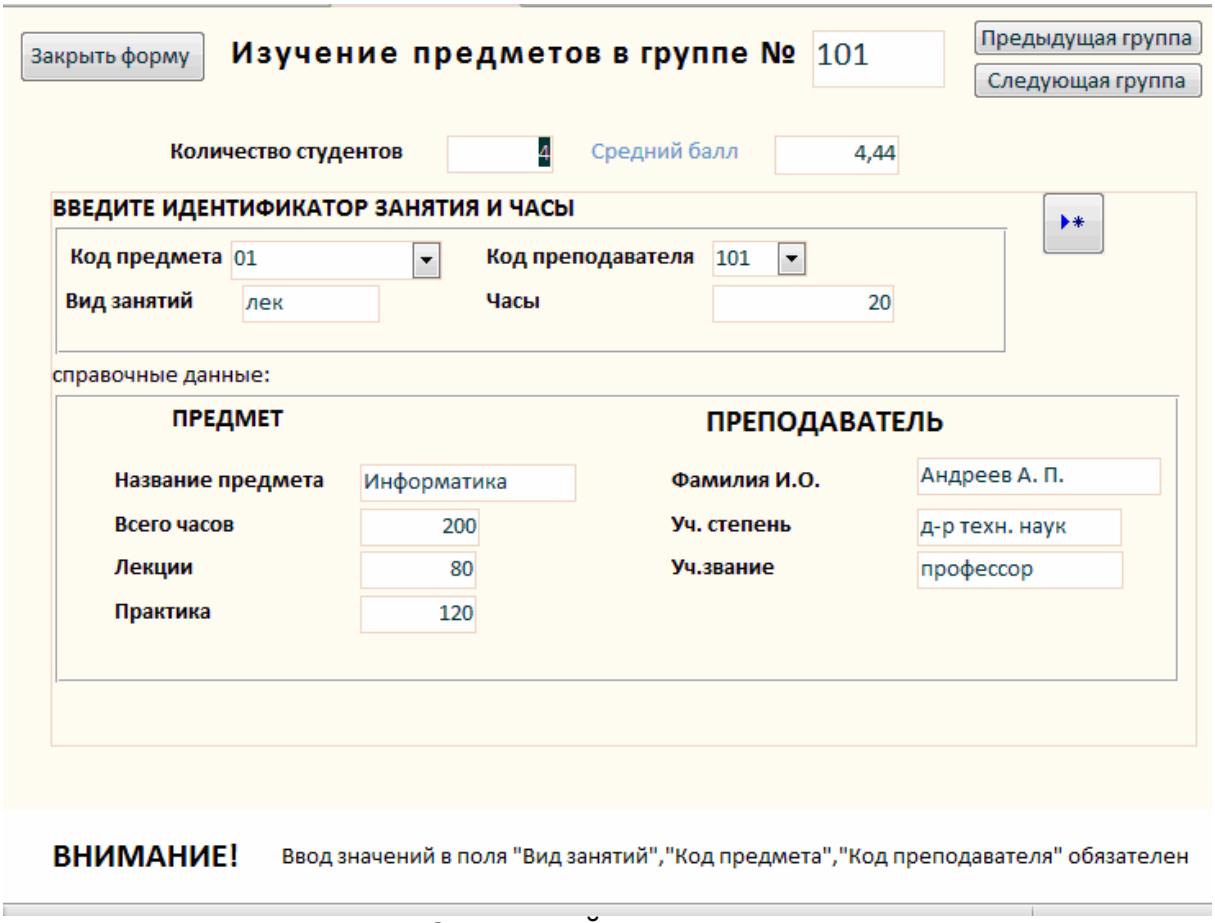


Рис. 6.19. Форма ПЛАН ЗАНЯТИЙ в режиме формы

Загрузка подчиненной таблицы. Для загрузки записи нового занятия в таблицу ИЗУЧЕНИЕ через вызванную форму необходимо сделать текущим номер группы, для которой вводятся данные о занятиях из документа «План занятий». Это можно сделать путем просмотра записей групп при помощи кнопок **Предыдущая группа** и **Следующая группа** (рис. 6.19).

При большом числе групп целесообразно использовать функцию поиска нужной записи. Чтобы осуществить поиск, установим курсор в поле номера группы, нажмем на вкладке **Главная** в группе **Найти** кнопку **Найти** и в открывшемся окне **Поиск и замена** (рис. 6.20) зададим в качестве образца поиска номер нужной группы. После нажатия в этом окне кнопки **Найти далее** в форме отобразятся данные занятиях заданной группы.

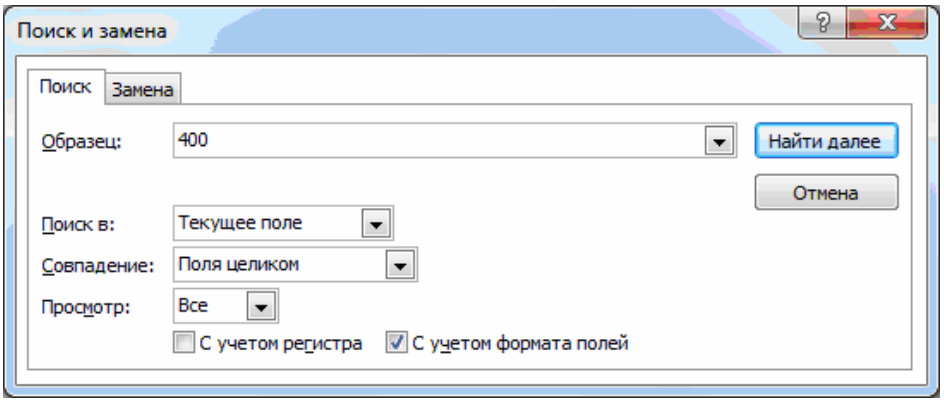


Рис. 6.20. Диалоговое окно поиска записи

Для ввода новых записей должен быть обеспечен переход в режим добавления новой записи в подчиненной форме. Для этого может быть создана специальная кнопка перехода к пустой записи **Добавить запись** (см. рис. 6.19).

Для формирования такой кнопки используем мастер кнопок, в диалоговом окне которого **Создание кнопок** (рис. 6.21) надо выбрать соответствующую категорию **Обработка записей** и действие **Добавить запись**, которые обеспечат формирование нужной процедуры обработки события.

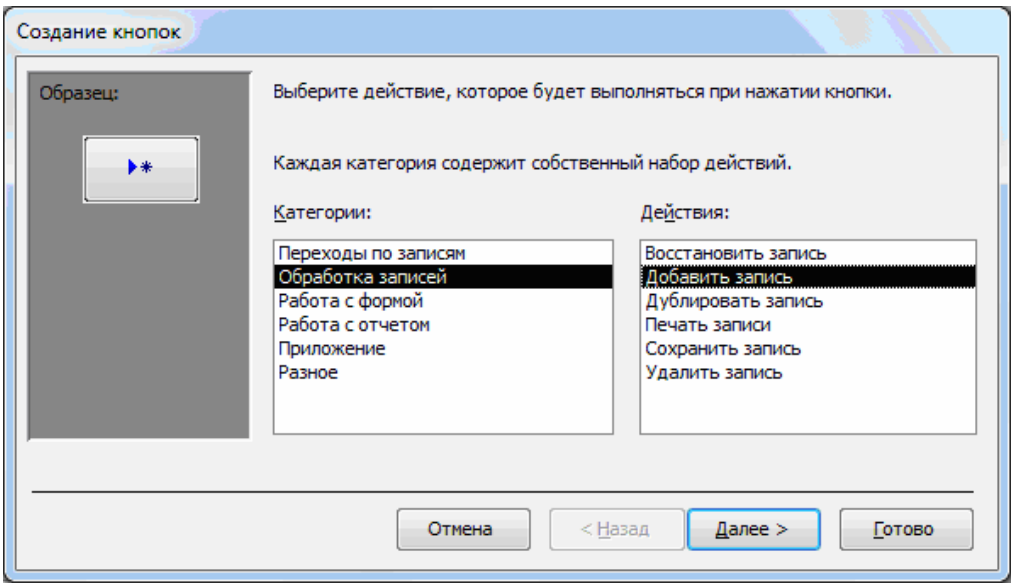


Рис. 6.21. Выбор параметров при создании кнопки для добавления записи

Установка свойств **Разрешить добавление**, **Разрешить удаление, Разрешить изменение**, обеспечивающих возможность добавления, удаления и изменения записей при загрузке и корректировке записей таблицы ИЗУЧЕНИЕ – источника записей, показана в окне свойств подчиненной формы (рис. 6.22).

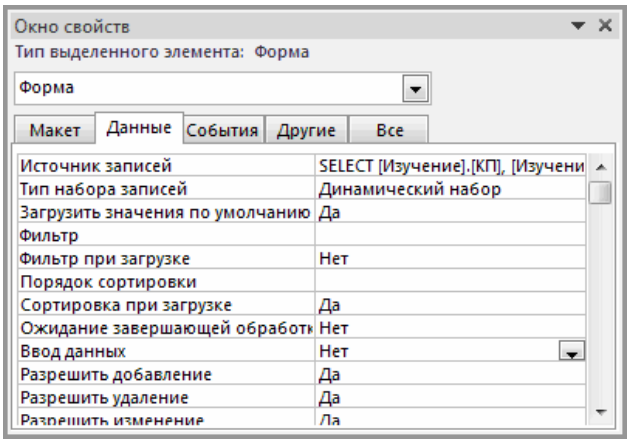


Рис. 6.22. Установка свойств подчиненной формы ИЗУЧЕНИЕ

Упражнение. Создайте многотабличную форму, для загрузки результатов сдачи экзаменов в таблицу УСПЕВАЕМОСТЬ и их просмотра. При создании формы произведите действия, аналогичные рассмотренным для таблицы ИЗУЧЕНИЕ.

В соответствии с технологией загрузки базы данных, рассмотренной в начале настоящей главы, осуществите проектирование формы для загрузки данных в подчиненную таблицу УСПЕВАЕМОСТЬ из документа-источника «Экзаменационная ведомость».

Определите общую структуру составной формы для ввода (просмотра) данных об оценках студентов группы по предмету в соответствии с подсхемой данных для составной формы, приведенной на рис. 6.23.

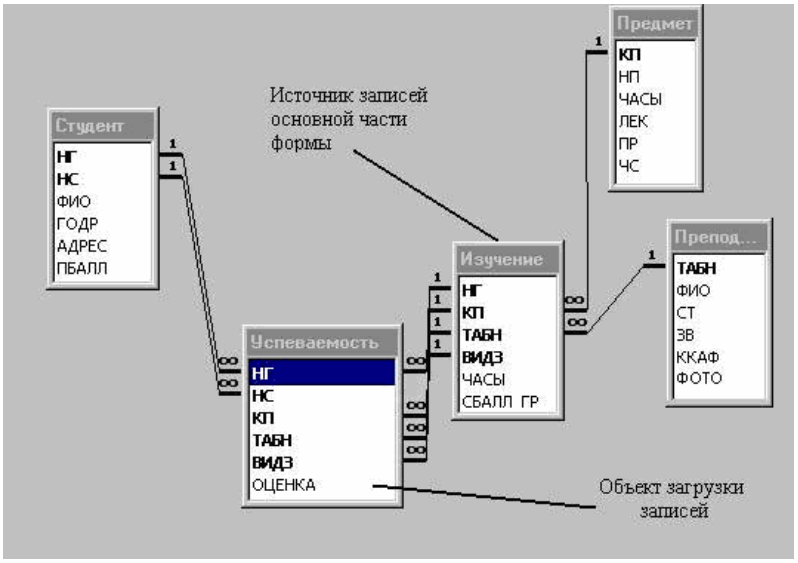


Рис. 6.23. Подсхема для составной формы ВЕДОМОСТЬ УСПЕВАЕМОСТИ

В результате загрузки в БД данных об оценках студентов группы по предмету в БД должны формироваться только записи таблицы УСПЕВАЕМОСТЬ (объ-

ект загрузки). Загрузку и просмотр этих данных удобно производить по каждому проведенному в группе занятию в отдельности из соответствующей «Экзаменационной ведомости». Поэтому в подсхему для формы ввода включается таблица ИЗУЧЕНИЕ, которую по этой причине целесообразно выбрать в качестве источника основной части составной формы. Кроме того, в форме предусмотрен вывод (отображение) данных о предмете и преподавателе, проводящем занятие, а в списке студентов необходимо предусмотреть отображение его фамилии. Для этого в подсхему включены таблицы СТУДЕНТ, ПРЕДМЕТ и ПРЕПОДАВАТЕЛЬ.

Спланируйте размещение реквизитов в макете формы так, чтобы обеспечить удобный ввод данных в таблицу УСПЕВАЕМОСТЬ из документа «Экзаменационная ведомость», а также отображение справочной информации о студенте, предмете и преподавателе, в списке студентов предусмотрите вывод фамилии студента.

Выполните конструирование экранной формы (рис. 6.24), через которую будет осуществляться ввод, добавление и изменение записей таблицы базы данных УСПЕВАЕМОСТЬ.

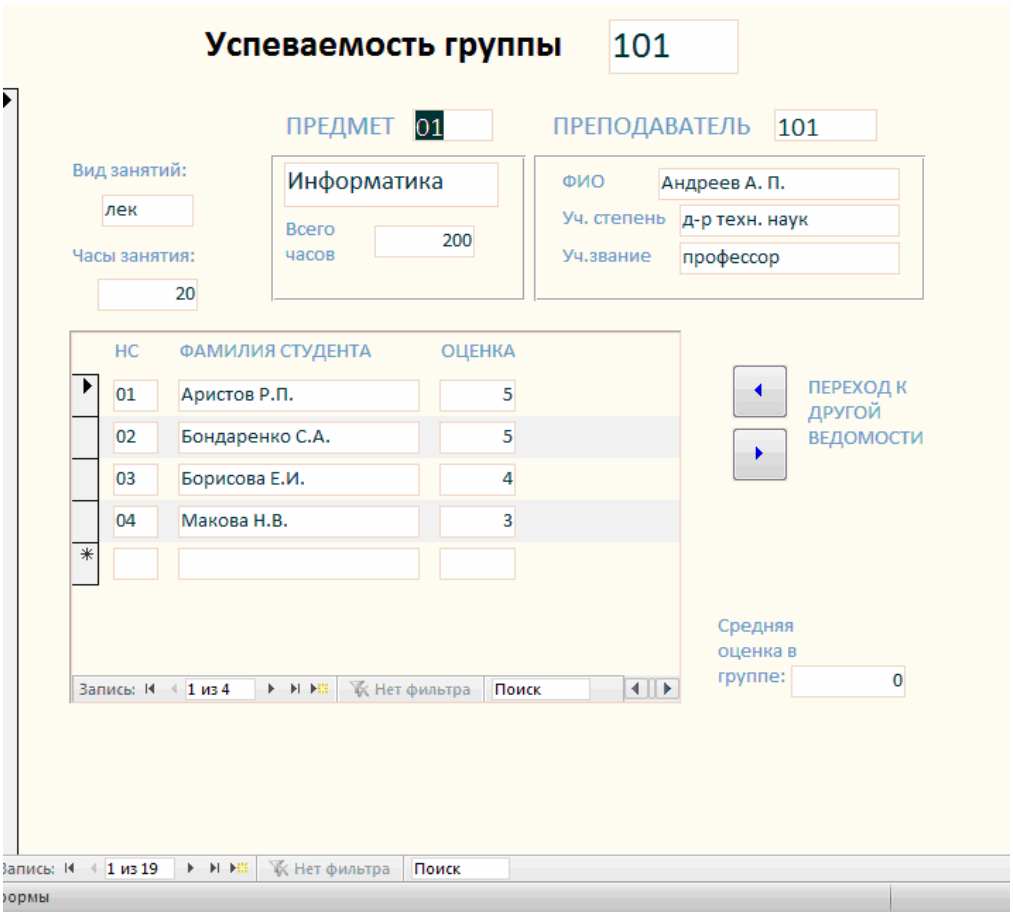


Рис. 6.24. Форма ввода и просмотра данных об успеваемости студентов по теории и практике

Загрузите через построенную форму данные из документа «Экзаменационная ведомость» в таблицу УСПЕВАЕМОСТЬ. Используйте значения данных, приведенные в Приложении В второй главы.

Урок 7. Запросы

В этом разделе на конкретных примерах рассматривается технология конструирования запросов различного вида. Дано описание конкретных действий пользователя в процессе конструирования запросов.

Подробно описан процесс конструирования однотабличного и многотабличного запроса, показано формирование вычисляемых полей, использование групповых операций и параметров запроса.

Рассмотрено обновление таблиц с помощью запроса. Кроме того, в конце раздела приведен пример решения задачи на основе нескольких запросов.

Конструирование однотабличного запроса на выборку

Рассмотрим процесс конструирования однотабличного запроса на выборку на примере получения информации из таблицы ПРЕДМЕТ базы данных «Учебный процесс».

Использование логических операций в условии отбора

Пусть надо выбрать предметы, по которым общее число изучения не более 100, и есть лекции, а также выбрать предметы, по которым общее число часов больше 150 и число семестров изучения не более двух.

Результат должен содержать наименование предмета (НП), общее число часов по предмету (ЧАСЫ), количество лекционных часов (ЛЕК) и число семестров (ЧС).

Для создания запроса в режиме конструктора выберем вкладку **Создание** на панели быстрого доступа и нажмем кнопку **Конструктор запросов**.

После нажатия кнопки появляется окно запроса на выборку в режиме конструктора **Запрос1** (рис. 7.1) и диалоговое окно **Добавление таблицы**. В диалоговом окне выберем таблицу ПРЕДМЕТ и нажмем кнопку **Добавить**.

Выбранная таблица будет отображена в области схемы данных запроса. Закроем окно **Добавление таблицы**.

В окне конструктора (рис. 7.1) перетащим из списка полей таблицы ПРЕДМЕТ поля НП, ЧАСЫ, ЛЕК и ЧС в столбцы бланка запроса в строку **Поле**.

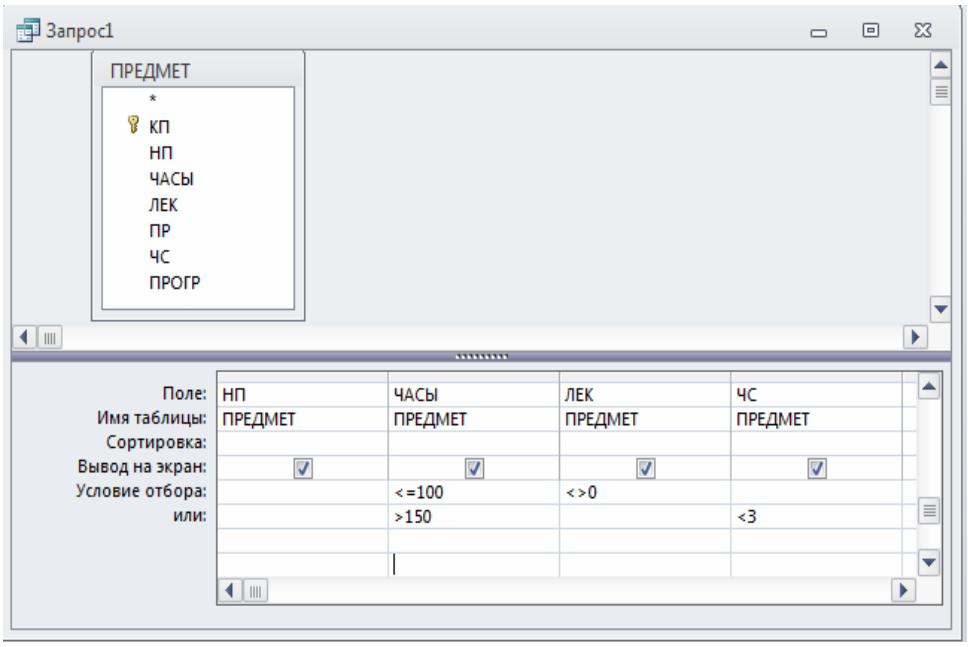


Рис. 7.1. Окно конструктора запроса на выборку с логическими операциями в условии отбора.

Сформулированные в задаче условия требуют формирования следующего логического выражения:

(ЧАСЫ<=100 AND ЛЕК<>0) OR (ЧАСЫ>150 AND ЧС<3)

Здесь ЛЕК<>0 (число лекций не равно нулю), соответствует заданному в задаче условию выбрать предметы, в которых есть лекции.

Условия из первых скобок запишем в соответствующих полях ЧАСЫ и ЛЕК первой строки **Условия отбора**. Между условиями в разных полях одной строки выполняется логическая операция . Условия из вторых скобок запишем в соответствующих полях ЧАСЫ и ЧС второй строки **Условие отбора**. Между условиями, записанными в разных строках, выполняется логическая операция .

Выполним запрос, нажав на панели конструктора запросов кнопку **Выполнить.**

На экране появится окно запроса в режиме таблицы с записями из таблицы ПРЕДМЕТ, отвечающими заданным условиям отбора.

Сохраним запрос, нажав кнопку **Сохранить** на вкладке **Файл** и задав нужное имя запроса. Закроем текущий запрос нажав кнопку окна запроса **Закрыть**. Сохраненный запрос можно выполнить, выделив запрос в окне **Все объекты Access**, нажатием правой кнопки мыши и выбрав кнопку **Открыть**.

Использование в условии отбора выражений с именами полей

В предыдущем примере в условии отбора в качестве операндов использовались только значения для отбора по конкретным полям. Создадим запрос, в условии отбора которого сравниваются значения в разных полях.

Пусть необходимо проверить правильность задания общих часов в таблице ПРЕДМЕТ. По запросу должны отбираться только те записи, в которых значение в поле ЧАСЫ не равно значению, получаемому при сложении значений полей ПР и ЛЕК.

Такое условие записывается в бланке запроса в столбце ЧАСЫ и в нем используются имена полей [ПР] и [ЛЕК], как показано в бланке запроса на рис.7.2.

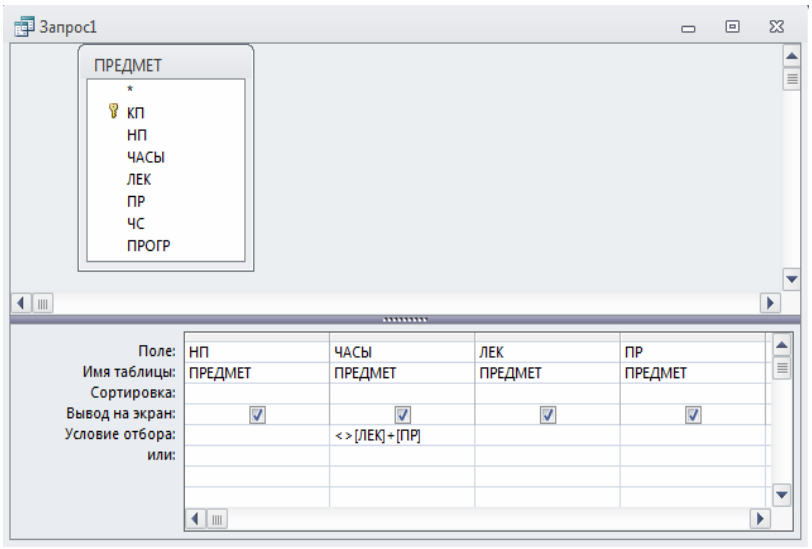


Рис.7.2. Окно запроса на выборку из таблицы ПРЕДМЕТ записей, в которых количество часов не равно сумме часов лекций и практики

Конструирование многотабличного запроса на выборку

Рассмотрим технологию конструирования многотабличного запроса на выборку на примере получения информации об успеваемости студентов из БД «Учебный процесс».

Запрос на основе нескольких взаимосвязанных таблиц

Пусть необходимо получить информацию об оценках полученных студентами по всем предметам. Результат должен содержать фамилию студента, наименования сданных предметов и оценки.

Для создания запроса на панели быстрого доступа выберем вкладку **Создание** и нажмем кнопку **Конструктор запросов**.

Формирование схемы данных запроса

В окне Добавление таблицы выберем таблицы:

- СТУДЕНТ- для выборки фамилия студента из поля ФИО
- УСПЕВАЕМОСТЬ- для определения кодов предметов (поле КП), по которым студент сдал экзамены, выборки оценок по предмету (из поля ОЦЕНКА).
- ПРЕДМЕТ- для выборки наименования предмета (из поля НП), представленного кодом КП в таблице УСПЕВАЕМОСТЬ.

Закроем окно **Добавление таблицы**.

В окне конструктора запросов (рис. 7.3) представлена схема данных запроса, содержащая выбранные таблицы. Между таблицами автоматически установлены необходимые связи:

- *Одно-многозначная* связь между таблицами СТУДЕНТ и УСПЕВАЕМОСТЬ по составному ключу НГ+НС в соответствии с построенной ранее схемой данных.
- *Связь-объединение* между УСПЕВАЕМОСТЬ и ПРЕДМЕТ поскольку эти таблицы имеют поля с одинаковым именем КП и одинаковым типом данных.

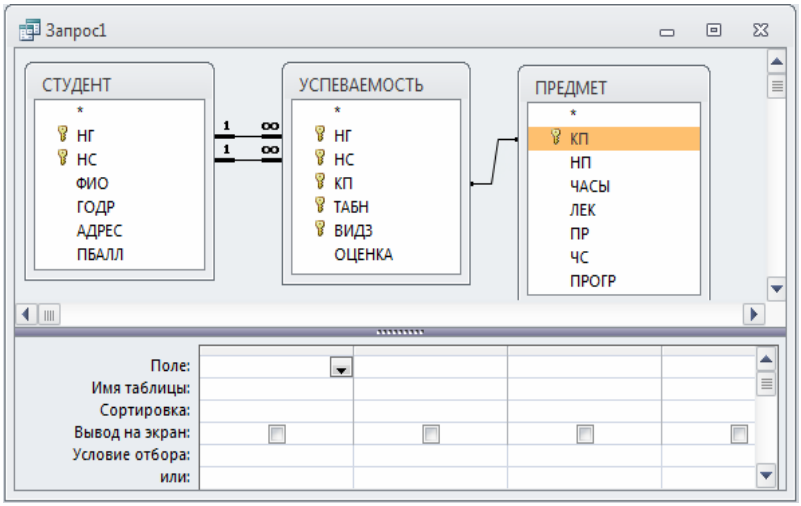


Рис. 7.3. окно запроса об успеваемости студента с созданной схемой данных.

Подготовка бланка запроса

Поскольку в запросе используется несколько таблиц, в бланке запроса удобно видеть имя таблицы наряду с именем поля. Для отображения имен таблиц в бланке запроса (рис. 7.4) нажмем кнопку **Имена таблиц** на панели инстру-

ментов конструктора или нажмем соответствующую кнопку в контекстном меню, который вызовем правой кнопкой мыши.

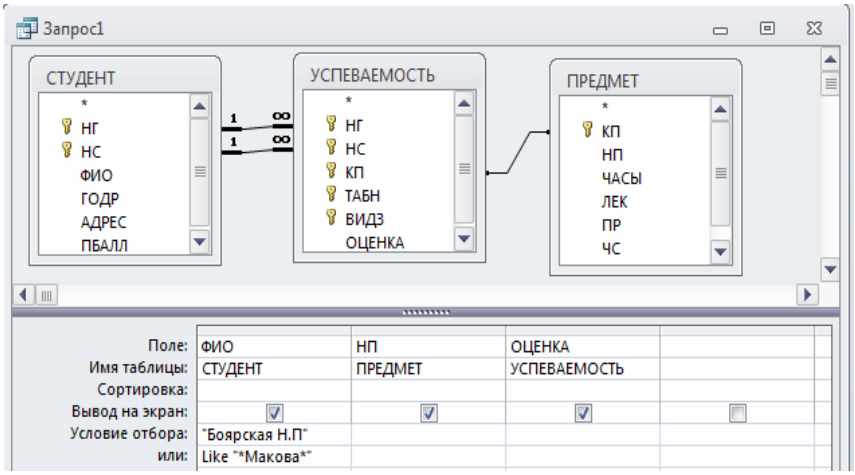


Рис. 7.4. Запрос на получение информации о сдаче экзаменов студентками Боярской Н.П. и Маковой.

Перетащим с помощью мыши поля, включаемые в результат выполнения запроса, в строку бланка запроса **Поле**:

- ФИО- из таблицы СТУДЕНТ
- НП- из таблицы ПРЕДМЕТ
- ОЦЕНКА- из таблицы УСПЕВАЕМОСТЬ

Ввод значений в условия отбора записей

Пусть необходимо получить информацию об успеваемости конкретных студентов: Боярской Н.П. и Маковой.

Зададим в строке **Условие отбора** их фамилии. Запишем фамилии студентов в разных строках бланка запроса, поскольку необходимо выбрать записи со значением в поле ФИО- Боярская или Макова. Поскольку инициалы студентки Маковой неизвестны, ее фамилию зададим с использованием символа шаблона «*». Заметим, что фамилия с инициалами содержит точки, поэтому ее надо брать в кавычки. После ввода фамилии с символом шаблона система сама вставляет оператор Like, определяющий поиск по образцу. Заполненный бланк запроса представлен на рис. 7.4.

Выполним запрос, нажав на панели конструктора запросов кнопку **Выполнить**.

Замечание. Записи о заданном студенте появятся в результирующей таблице запроса только в том случае, если запись об этом студенте содержится в таблице СТУДЕНТ, а в таблице УСПЕВАЕМОСТЬ имеются записи, связанные с записью о студенте.

Формирование записей результата при выполнении запроса

По заданной фамилии студента- Боярская Н.П.- в таблице СТУДЕНТ отыскивается запись. По значению ключа связи НГ+НС осуществляется выборка подчиненных записей из таблицы УСПЕВАЕМОСТЬ с оценками данного студента по разным предметам (в поле ОЦЕНКА). Для каждой из этих записей по значению ключа связи КП выбирается одна запись с наименованием предмета (НП) из таблицы ПРЕДМЕТ. Таким образом, таблица с результатом запроса будет содержать по одной записи о каждом предмете, сданном студентом. Аналогично формируются записи для второго заданного в запросе студента-Маковой.

Ввод параметров в запрос

В предыдущем примере для задания фамилии конкретного студента необходимо было корректировать бланк запроса. Чтобы избежать этого, целесообразно использовать в запросе параметры. При этом Access перед выполнением запроса через диалоговое окно будет запрашивать у пользователя конкретные значения параметров и введет их в условия отбора.

Пусть необходимо получить информацию об оценке студента по заданному предмету.

В условие отбора поля ФИО вместо конкретной фамилии введем название параметра, по которому будет запрашиваться фамилия при выполнении запроса. Название параметра введем как текст, заключенный в квадратные скобки:

[Фамилия и инициалы студента]

Этот текст Access воспринимает как имя параметра (рис. 7.5). В условие отбора поля НП введем второй параметр запроса:

[Наименование предмета]

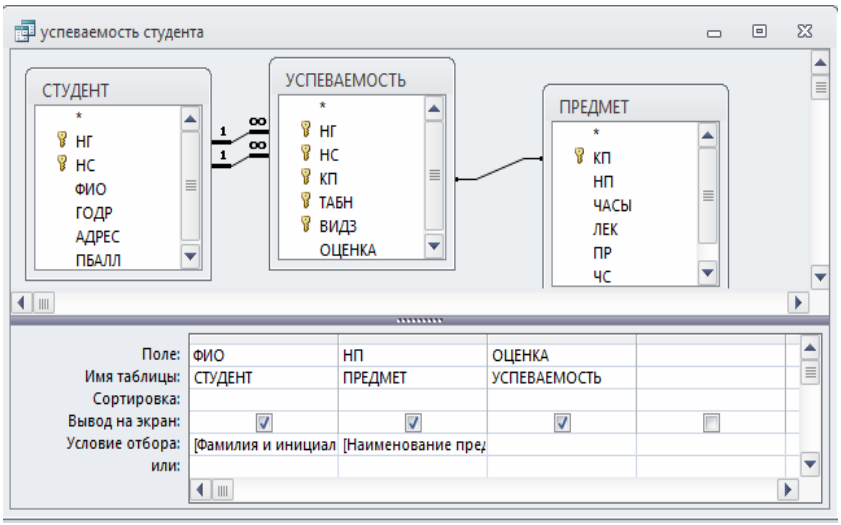


Рис. 7.5. Бланк запроса с параметрами для ввода ФИО и НП.

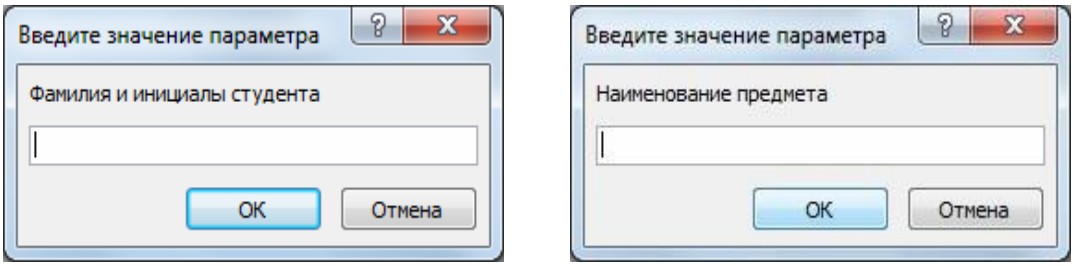


Рис. 7.6. Диалоговые окна для ввода параметров запроса.

При выполнении запроса Access выведет диалоговые окна, представленные на рис. 7.6, в которые пользователь сможет ввести нужные значения параметров.

Использование имен полей различных таблиц в условии отбора

Пусть необходимо выбрать записи из таблицы ИЗУЧЕНИЕ, в которых часы практических занятий по информатике не соответствуют равномерному распределению по семестрам всех часов практики.

Для решения этой задачи необходимо использовать таблицы:

- ИЗУЧЕНИЕ, в которой содержатся сведения о плановых занятиях в группах (в текущем семестре), в том числе о продолжительности (поле ЧАСЫ) каждого вида занятия (поле ВИД3).
- ПРЕДМЕТ, в которой содержатся сведения о наименовании (поле НП), общей продолжительности изучения предмета (поле ЧАСЫ), числа часов практики (ПР) и числе семестров изучения (ЧС).

Для отбора записей о практических занятиях по информатике из таблицы ИЗУЧЕНИЕ надо в строке **Условие отбора** для поля НП (ТАБЛИЦА ПРЕДМЕТ) задать значение «Информатика», а для поля ВИД3 (таблицы ИЗУЧЕНИЕ) задать значение «пр» (практическое занятие).

При равномерном распределении практики по семестрам общее число часов практических занятий по предмету (ПР) должно равняться произведению часов практики (ЧАСЫ) из таблицы ИЗУЧЕНИЕ на число семестров (ЧС) из таблицы ПРЕДМЕТ. Для решения рассматриваемой задачи надо включить в результат только те записи, для которых число часов не соответствует этому произведению. Для этого запишем в **Условие отбора** поля ПР (таблицы ПРЕДМЕТ) выражение:

[ИЗУЧЕНИЕ] ! [ЧАСЫ]*[ЧС]

Замечание. Указывать таблицу ИЗУЧЕНИЕ для поля ЧАСЫ обязательно, потому что поле с таким же именем имеется и в таблице ПРЕДМЕТ. Запрос на выборку с условиями отбора записей, удовлетворяющих сформулированной задаче, приведен на рис. 7.7. Результаты выполнения запроса приведены на рис. 7.8.

Замечание. Если результат выполнения запроса не содержит записей, то это означает, что для заданного предмета в каждой из студенческих групп часы практических занятий соответствуют равномерному распределению по семестрам всех часов практики.

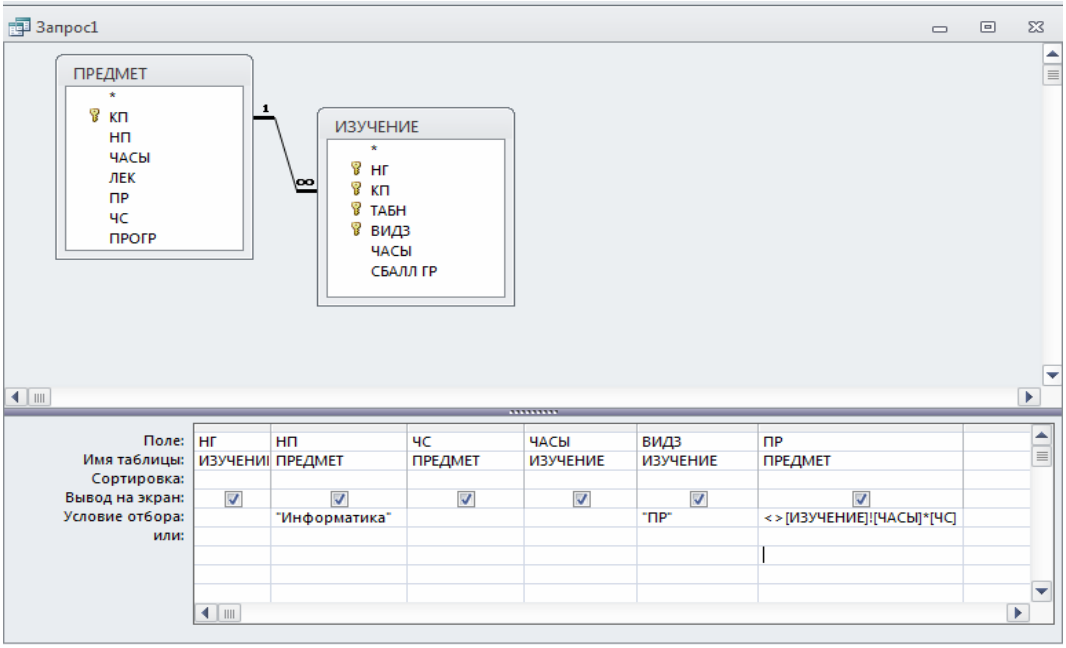


Рис. 7.7. Запрос с использованием в условиях отбора имен полей из разных таблиц.

Ном. группы	Название пр	Семестров	Всего часов	Вид занятий	Практика
101	Информатика	4	60	пр	120
102	Информатика	4	180	пр	120
201	Информатика	4	180	пр	120

Рис. 7.8. Записи о практических занятиях по информатике, в которых число часов не отвечает заданным условиям.

Создание вычисляемых полей в запросах

Вычисляемое поле, включенное в запрос, позволяет получить новое поле с результатами вычисления, отображаемыми только в таблице запроса, и не создает полей в исходных таблицах базы данных.

Рассмотрим технологию создания запроса с вычисляемым полем на примере таблицы ПРЕДМЕТ.

Пусть необходимо найти записи о предметах, в которых общее число часов по предмету не совпадает с суммой часов лекций и практики. Для решения этой

задачи рассчитаем разность между общим числом часов по предмету (поле ЧАСЫ) и суммой часов лекций (поле ЛЕК) и практики (поле ПР). в ответ включим только те записи, для которых эта разность не равна нулю.

Создадим запрос на выборку для таблицы ПРЕДМЕТ. Перетащим в бланк запроса поля НП, ПР, ЛЕК, ЧАСЫ (рис.7.9).

Создание вычисляемого поля

Для получения разности создадим вычисляемое поле в пустой ячейке строки **Поле**, записав туда выражение:

[ЧАСЫ] - [ПР] - [ЛЕК]

Для отбора записей с ненулевым значением разности в вычисляемом поле в строку **Условие отбора** введем <>0 (не равно 0).

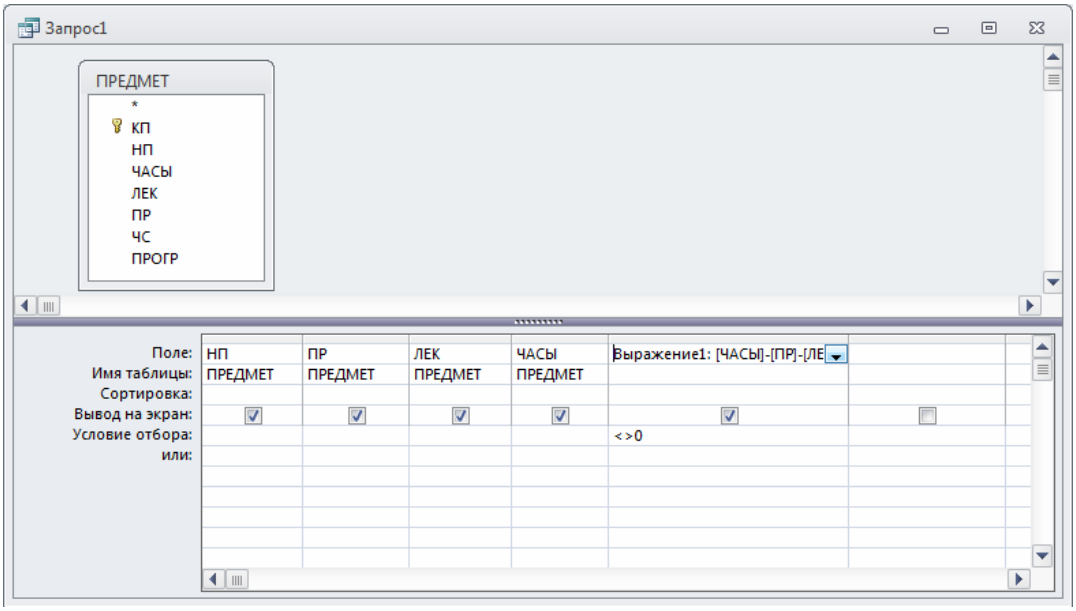


Рис. 7.9. Запрос с вычисляемым полем и условием отбора по его значению.

После ввода выражения система формирует имя вычисляемого поля по умолчанию- «Выражение 1». Это имя вставится перед выражением. Для изменения имени установим курсор мыши в вычисляемом поле бланка запроса и нажмем правую кнопку мыши. В контекстно- зависимом меню выберем **Свойства** поля, а в строку **Подпись** введем новое имя поля- «ЧАСЫ не равны ПР+ЛЕК». Имя поля может быть исправлено также непосредственно в бланке запроса.

Использование построителя выражений

Для формирования сложного выражения в вычисляемом поле целесообразно использовать построитель выражений. Построитель позволяет выбрать необходимые имена полей из таблиц, запросов, форм, знаки операций, функции.

Вызовем построитель выражений, нажав команду **Построить** в контекстно-зависимом меню (курсор мыши должен быть установлен на строке **Поле** вычисляемого поля).

В левой части окна **Построитель выражений** (рис. 7.10) выберем таблицу ПРЕДМЕТ, на которой построен запрос. Справа отобразится список ее полей. Последовательно выберем нужные поля, добавляя их двойным щелчком мыши, знаки операций вводятся с клавиатуры. При этом в верхней части окна сформируется выражение.

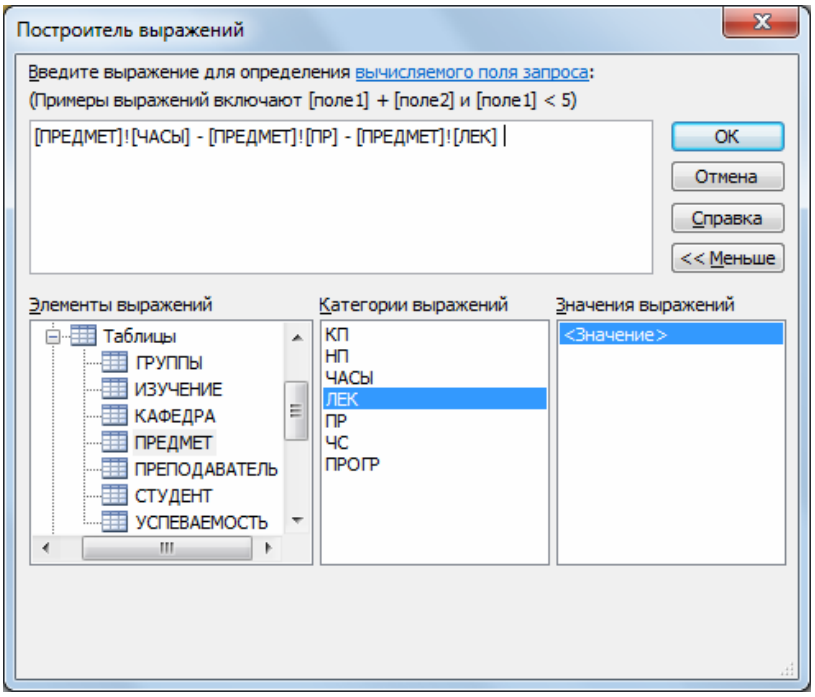


Рис. 7.10. Окно построителя выражений при формировании вычисляемого поля.

Сохраним запрос под именем «Разность часов по предмету». Сохранить. Сохраненный запрос можно выполнить, выделив запрос в окне **Все объекты Access**, нажатием правой кнопки мыши и выбрав кнопку **Открыть**.

Построенный запрос может быть использован для проверки правильности заполнения поля ЧАСЫ в таблице ПРЕДМЕТ.

Использование групповых операций в запросах

Назначение групповых операций

Групповые операции позволяют выделить группы записей с одинаковыми значениями в указанных полях и использовать для этих групп одну из статистических функций. В Access предусмотрено девять статистических функций:

- Sum - сумма значений некоторого поля для группы

- Avg - среднее от всех значений поля в группе
- Max, Min - максимальное, минимальное значение поля в группе
- Count - число значений поля в группе без учета пустых значений
- Stdev - среднеквадратичное отклонение от среднего значения поля в группе
- Var - дисперсия значений поля в группе
- First и Last - значение поля из первой или последней записи в группе

Результат запроса с использованием групповых операций содержит по одной записи для каждой группы. В запрос включаются поля, по которым производится группировка, и поля, для которых выполняются групповые функции.

Порядок создания запроса с использованием групповых операций

Для создания запроса с использованием групповых операций формируется запрос на выборку. В бланк запроса включаются поля, по которым надо произвести группировку, и поля, по которым надо произвести статистические вычисления.

Выполните команду **Создать/Конструктор запросов** и на панели инструментов конструктора запросов нажмите кнопку **Итоги**.

Для групповых вычислений по некоторому полю нужно заменить в нем слово **Группировка** на нужную статистическую функцию. Выбрать нужную функцию можно через раскрывающийся в поле список.

Конструирование однотабличного запроса с групповой операцией

Рассмотрим технологию конструирования однотабличного запроса с групповой операцией на примере таблицы СТУДЕНТ.

Запрос с функцией Count

Определим фактическое число студентов в группе. Создадим запрос на выборку из таблицы СТУДЕНТ. Из списка таблицы СТУДЕНТЫ перетащим в бланк запроса поле НГ (номер группы). Таким образом мы укажем, что по этому полю должна производиться группировка. Перетащим в бланк запроса поле НС, по которому будет вычисляться функция Count для подсчета числа студентов в группе.

Нажмем кнопку **Итоги**. Заменим слово "Группировка" в столбце НС на функцию Count. Для этого вызовем список и выберем эту функцию. Бланк запроса примет вид, показанный на рис. 7.11. Результат запроса показан на рис. 7.12.

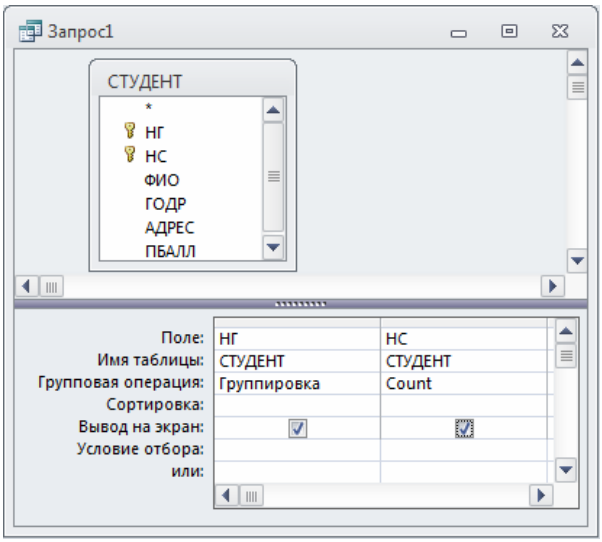


Рис. 7.11. Запрос с групповой операцией подсчета числа студентов в группе

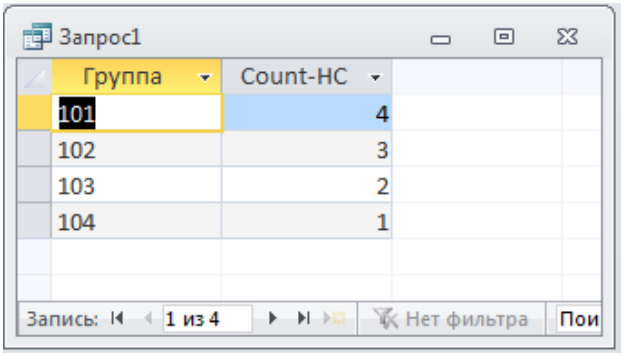


Рис. 7.12. Результат подсчета числа студентов в группе

Подпись поля "Count_НС" можно заменить на "Фактическое число студентов". Для ввода этой подписи в бланке запроса установим на поле НС курсор мыши и нажмем правую кнопку. В контекстно-зависимом меню выберем команду **Свойства** . В **Окне свойств** наберем в строке **Подпись** "Фактическое число студентов". Таблица результата после доработки запроса показана на рис. 7.13. Сохраним запрос под именем "Число студентов в группах".

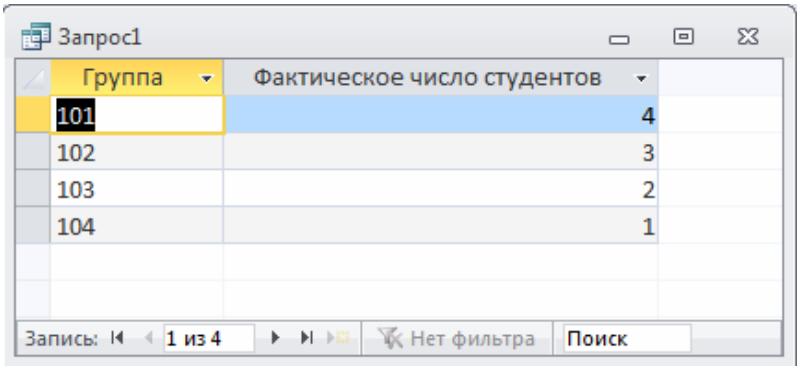


Рис. 7.13. Таблица результата с измененной подписью поля

Запрос с функцией Avg

Подсчитаем средний проходной балл в группе. Сформируем запрос на выборку для таблицы СТУДЕНТ с функцией Avg для поля ПБАЛЛ (проходной балл студента). В бланке запроса заполним поля, как показано на рис. 7.14.

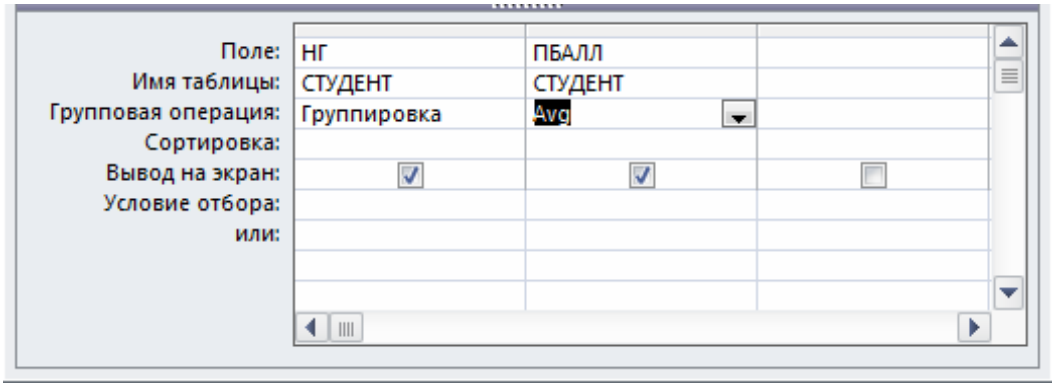


Рис. 7.14. Запрос с групповой операцией подсчета среднего значения

Для ограничения точности результата двумя знаками выберем в **Окне свойств** в строке **Формат поля** значение **Фиксированный**. Результат выполнения запроса представлен на рис. 7.15.

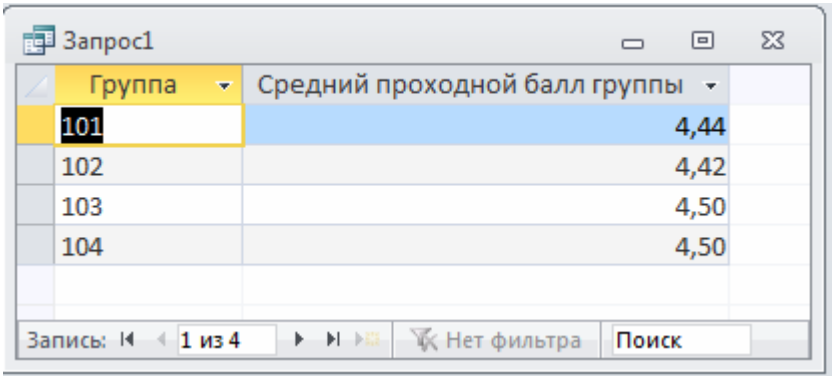


Рис. 7.15. Результат подсчета среднего значения

Сохраним этот запрос под именем "Средний проходной балл группы"

Запрос с несколькими групповыми функциями

Выполним расчет числа студентов и среднего проходного балла в группе в одном запросе. Это возможно, т. к. группы записей в обоих случаях формируются одинаково (рис. 7.16). Сохраним этот запрос под именем "Число студентов и средний ПБАЛЛ группы".

Задание условий отбора в запросах с групповыми операциями

В запрос с групповыми операциями можно включать поля для задания условий отбора записей из таблиц.

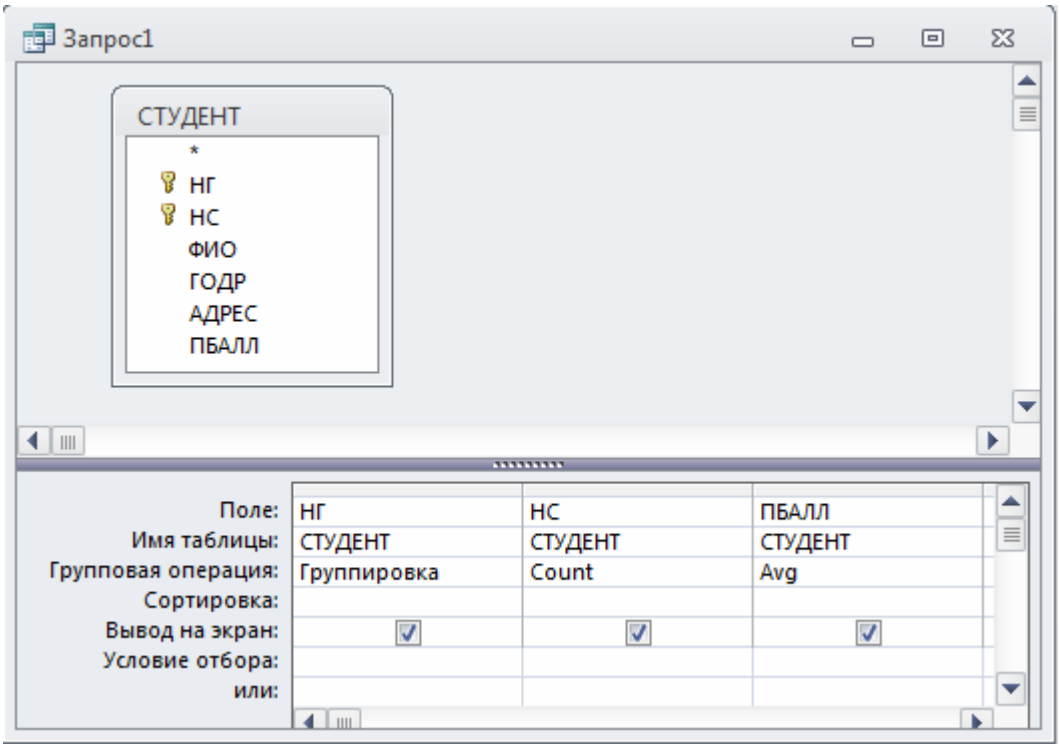


Рис. 7.16. Запрос с одновременным использованием двух групповых функций

Подсчитаем число студентов в каждой из групп с проходным баллом больше 4,7.

Для этого в запрос **Число студентов и средний ПБАЛЛ группы** вторично включим поле ПБАЛЛ и в строке **Групповые операции** заменим значение **Группировка** на значение **Условие**, выбрав его из списка. После этого введем в строку **Условие отбора** ">4,5" (рис. 7.17).

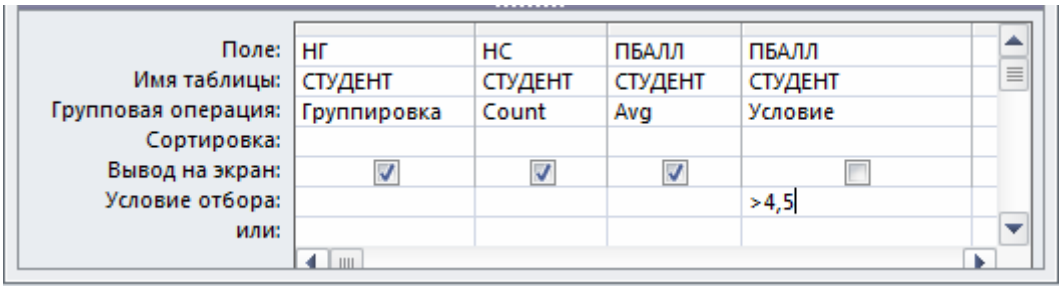


Рис. 7.17. Запрос с групповыми операциями и полем, введенным для определения условий отбора записей из таблицы СТУДЕНТ

Заметим, что средний балл в этом запросе также вычисляется только для студентов с проходным баллом, превосходящим 4,7.

Условие отбора, заданное в поле, по которому проводится группировка, или в поле, где записана функция группировки, позволяет отобрать только нужные

группы записей, например, группы студентов с заданным номером или с заданным средним проходным баллом.

Конструирование запроса на создание таблицы

Запрос на *создание таблицы* используется для сохранения результата запроса. Этот вид запроса основан на *запросе на выборку*, но, в отличие от него, сохраняет таблицу с результатами запроса.

Необходимость в сохранении результатов запроса возникает, например, когда невозможно построить запрос непосредственно на другом запросе. К этому случаю относится, в частности, построение запроса на обновление полей на основе запроса с операцией группировки.

Сформируем запрос на создание таблицы на примере ранее полученного запроса на выборку с групповыми вычислениями **Число студентов в группах** (см. рис. 7.11-7.13).

В области навигации вызовем названный запрос в режиме конструктора запросов. Преобразуем этот запрос в запрос на создание таблицы, выбрав тип запроса на панели конструктора **Создание таблицы**. В окне **Создание таблицы** введем имя создаваемой таблицы "Число студентов" (рис. 7.18).

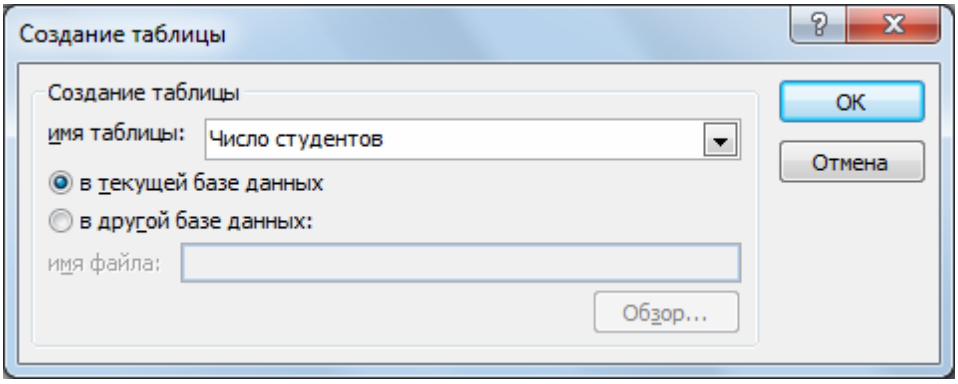


Рис. 7.18. Определение имени таблицы, создаваемой в запросе

Для того, чтобы просмотреть, какие записи будут помещены в новую таблицу, щелкните по кнопке панели инструментов **Выполнить**. Выполните запрос, чтобы таблица ЧИСЛО СТУДЕНТОВ была сохранена в базе данных. Теперь эту таблицу можно увидеть в списке таблиц окна БД.

Упражнение. Преобразуйте запрос на выборку Средний проходной балл группы (см. рис. 7.14, 7.45) в запрос на создание таблицы, а создаваемую таблицу назовите “СРБАЛЛ”.

Конструирование запроса на обновление

Обновление полей значениями, рассчитанными с использованием групповых операций

Рассмотрим технологию создания запроса на обновление на примере обновления поля КОЛ (количество студентов группы в таблице ГРУППА).

Количество студентов в группах ранее было подсчитано в запросе на выборку **Число студентов в группах** (см. рис. 7.11-7.13) с использованием статистической функции Count. Запрос на обновление непосредственно на таком запросе построить нельзя. Поэтому используем для обновления не сам запрос, а таблицу ЧИСЛО СТУДЕНТОВ, полученную по запросу на создание таблицы в предыдущем пункте.

Для формирования запроса на обновление сначала создадим запрос на выборку на основе двух таблиц: обновляемой таблицы ГРУППА и таблицы ЧИСЛО СТУДЕНТОВ, содержащей данные для обновления. Заметим, что в подсхеме данных запроса автоматически устанавливается связь этих таблиц по полю с именем НГ. Для преобразования запроса на выборку в запрос на обновление (рис. 7.19) выберем на панели конструктора тип запроса **Обновление**.

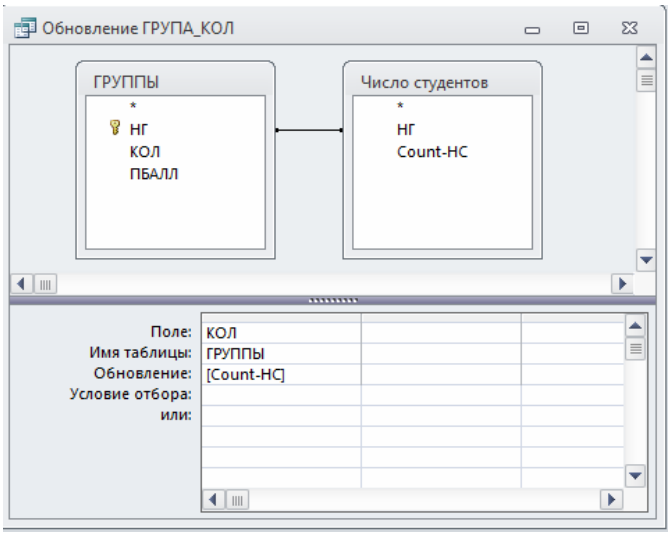


Рис. 7.19. Запрос на обновление таблицы ГРУППА

Заполним бланк запроса. Перетащим обновляемое поле КОЛ из списка таблицы ГРУППА. В строке **Обновление** введем имя поля "Count_НС" (таблицы ЧИСЛО СТУДЕНТОВ), из которого выбираются значения для обновления. Имя поля вводится в квадратных скобках.

Запрос можно выполнить, не выходя из режима конструктора. Содержимое обновляемого поля КОЛ можно просмотреть в режиме таблицы до и после

выполнения запроса. Для последующего использования подготовленного запроса сохраним его под именем "Обновление ГРУППА_КОЛ".

Упражнение

1. Произведите обновление поля ПБАЛЛ – средний проходной балл в таблице ГРУППА значениями из ранее созданной таблицы СРБАЛЛ.
2. Произведите обновление поля СРБАЛЛ-ГР – средняя оценка в группе по предмету в таблице ИЗУЧЕНИЕ. Для выполнения задания:
   - создайте запрос к таблице УСПЕВАЕМОСТЬ для расчета средней оценки в группе по предмету и сохраните результат в таблице, для чего группировку произведите по двум полям: НГ – номер группы и КП – код предмета:
   - обновите поле СРБАЛЛ-ГР в таблице ИЗУЧЕНИЕ, используя сохраненный результат.

Использование выражений в запросе на обновление

Рассмотрим формирование запроса на обновление с использованием выражения на примере заполнения поля ЧАСЫ для лекционных занятий в таблице ИЗУЧЕНИЕ. Пусть поле ЧАСЫ должно обновляться данными, вычисляемыми на основе полей ЛЕК (часы лекций) и ЧС (число семестров) из таблицы ПРЕДМЕТ. Расчетное число часов по лекциям определим по формуле ЛЕК/ЧС.

В соответствии с задачей в записях лекционных занятий таблицы ИЗУЧЕНИЕ необходимо обновить поле ЧАСЫ расчетным числом часов. Записи о лекционных занятиях можно выбрать по значению поля ВИДЗ этой таблицы, т. к. в нем указан вид занятия. Данные для расчета среднего числа часов содержатся в таблице ПРЕДМЕТ. Таким образом запрос должен строиться на основе таблиц ИЗУЧЕНИЕ и ПРЕДМЕТ.

Создадим сначала запрос на выборку на основе таблиц ИЗУЧЕНИЕ и ПРЕДМЕТ. Затем преобразуем его в запрос на обновление, нажав соответствующую кнопку панели инструментов.

Включим в бланк запроса обновляемое поле ЧАСЫ таблицы ИЗУЧЕНИЕ. В строке **Обновление** для этого поля введем выражение [ЛЕК]/[ЧС]. Для отбора в таблице ИЗУЧЕНИЕ обновляемых записей о лекционных занятиях в бланк запроса включим поле ВИДЗ и укажем в поле **Условия отбора** значение "лек".

Окончательно сформированный запрос показан на рис.7.20.

Выполним запрос, нажав кнопку **Выполнить.** В диалоговом окне появится сообщение о числе обновляемых записей.

Чтобы видеть результаты обновления в таблице ИЗУЧЕНИЕ, откройте ее одновременно с запросом.

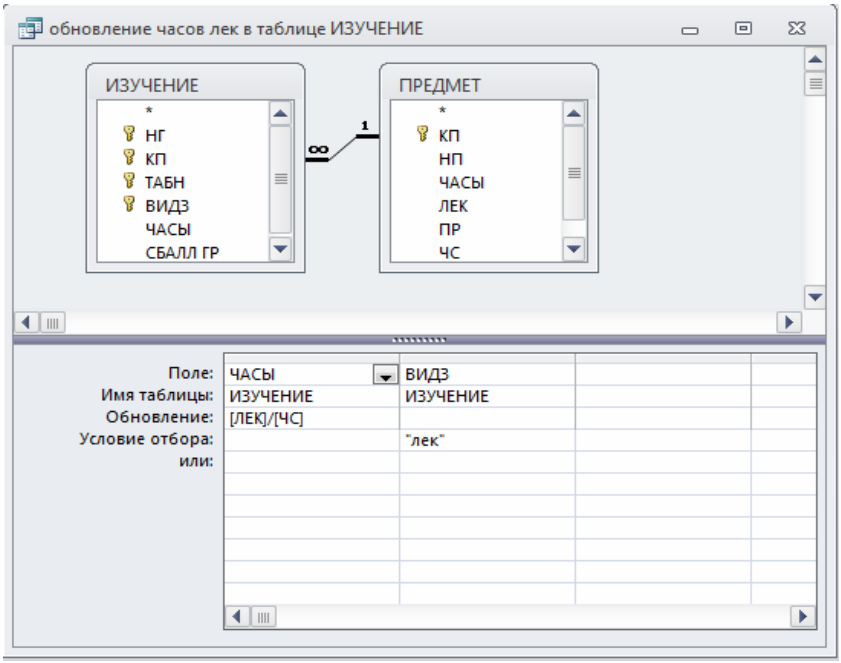


Рис.7.20. Запрос на обновление поля значениями, вычисляемыми по данным из другой таблицы

Конструирование перекрестного запроса

Создание перекрестного запроса, который позволяет получить данные в форме, подобной электронной таблице, с помощью мастера было рассмотрено выше. Однако такой запрос несложно построить полностью в режиме конструктора.

Построение запроса начинается как обычно, например, выбрать вкладку **Создание/ Конструктор запросов** в окне базы данных. В окне конструктора начинается создание запроса на выборку. В любой момент строящийся запрос на выборку может быть преобразован в перекрестный запрос. Для этого надо выбрать тип запроса **Перекрестный** на панели.

Ниже рассмотрим действия пользователя при работе с перекрестным запросом в режиме конструктора на примере.

Воспользуемся в качестве примера перекрестным запросом **Изучение предметов группами**, полученным мастером перекрестных запросов. Для большей информативности полученной перекрестной таблицы заменим в ней коды предметов их наименованиями. Полю, содержащему результат суммирования по строкам, дадим пользовательское имя "Всего часов".

Для выполнения перечисленных преобразований откроем перекрестный запрос **Изучение предметов группами** в режиме конструктора.

Поле НП (наименование предмета) размещено в таблице ПРЕДМЕТ, поэтому ее нужно добавить к разрабатываемому запросу. Для этого, находясь в окне конструктора, нажмем кнопку **Отобразить таблицу**.

Теперь схема данных запроса состоит из таблиц ПРЕДМЕТ и ИЗУЧЕНИЕ, связанных по полю КП (код предмета) отношением один-ко-многим (рис. 7.21).

Заменим в бланке запроса поле КП на поле НП таблицы ПРЕДМЕТ. Для этого щелкнем правой кнопкой мыши на области отображения полей таблиц и выберем **Имена таблиц**, чтобы получить в бланке информацию о принадлежности поля к таблице. Далее в поле КП в строке **Имя таблицы** нажмем кнопку списка и выберем поле ПРЕДМЕТ, а в строке **Поле** - поле НП.

Для изменения подписи поля ИТОГОВОЕ ЗНАЧЕНИЕ, содержащего сумму по строкам, щелкнем правой кнопкой мыши, находясь в зоне этого поля. В открывшемся контекстно-зависимом меню выберем пункт **Свойства**. В окне **Свойства** введем в строку **Подпись** "Всего часов". Окончательно сформированный перекрестный запрос приведен на рис. 7.21.

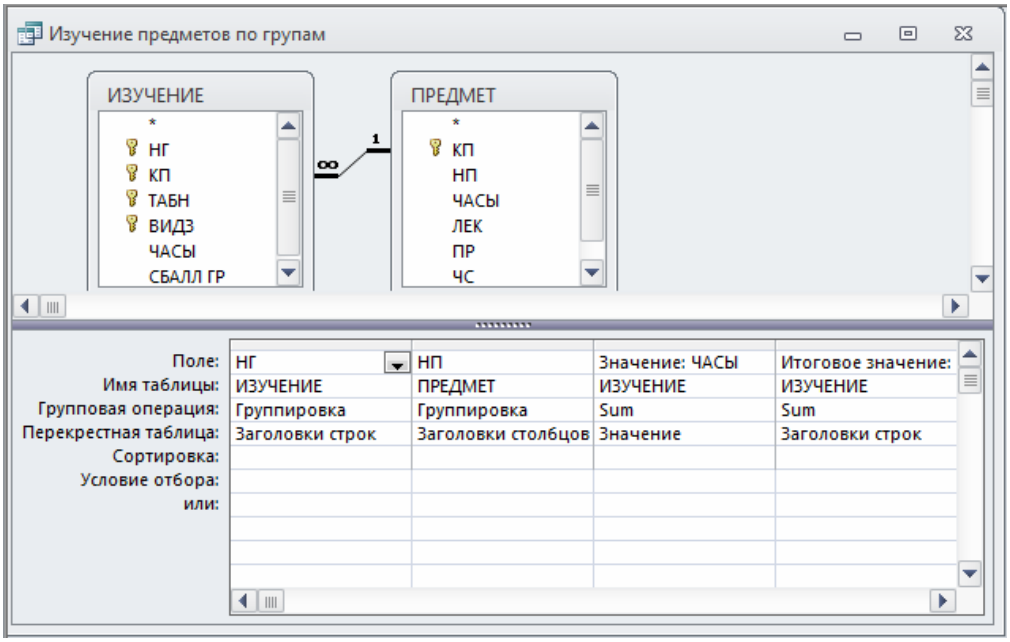


Рис. 7.21. Перекрестный запрос в режиме конструктора

Результат выполнения полученного перекрестного запроса приведен на рис. 7.22.

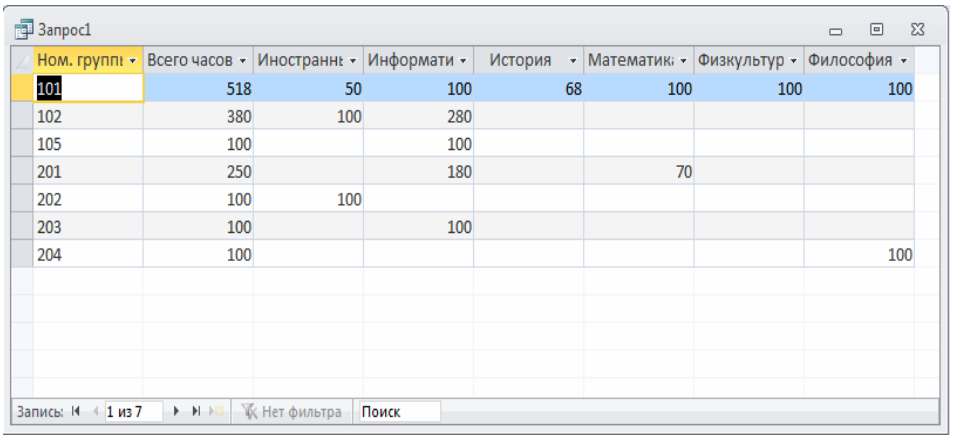


Рис. 7.22. Результат выполнения перекрестного запроса

Решение задач на основе нескольких запросов

Выше рассматривались примеры простых задач, решение которых осуществлялось выполнением одного запроса. В этом случае запросом реализуется весь алгоритм формирования результата на основе входных данных из таблиц базы и параметров задачи.

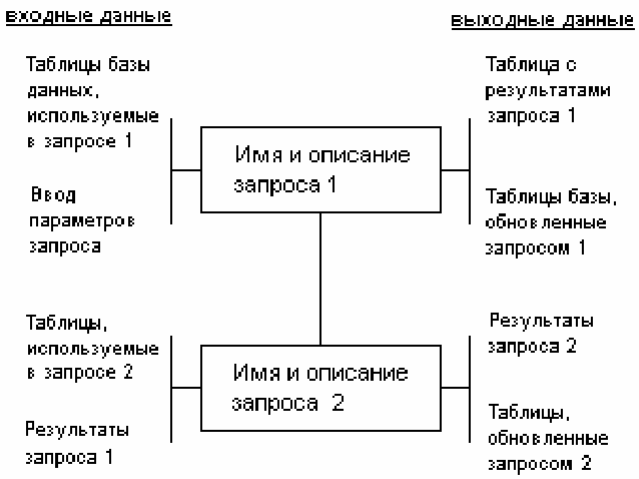


Рис. 7.23. Функционально-технологическая схема задачи из двух последовательно выполняемых запросов

Для описания алгоритма задач, реализуемых одним запросом, обычно достаточно словесного описания действий. Целесообразно также использовать функционально-технологическую схему, на которой указываются входные и выходные таблицы данных.

Более сложные задачи требуют последовательного выполнения нескольких запросов. Каждый из запросов имеет свои входные и выходные данные. В простейшем случае выходные данные предшествующего запроса являются входными для следующего построенного на нем запроса, и, только выполнив

последний запрос в цепочке запросов построенных друг на друге, вы инициируете последовательное выполнение всех запросов цепочки и полное решение задачи. Функционально-технологическая схема задачи, решаемой с помощью двух последовательно выполняемых запросов, в общем виде приведена на рис. 7.23.

Запросы в Access являются мощным средством решения различных задач. При этом возможно построение сложных запросов, в том числе построенных на других запросах. Ниже рассматривается построение таких запросов, а также реализация задачи последовательно выполняющимися запросами.

Построение запроса на основе другого запроса

Выполним анализ оценок, полученных студентами по различным предметам. Например, подсчитаем число оценок (2,3,4,5) по каждому из предметов.

Создадим сначала многотабличный запрос на выборку на основе таблиц СТУДЕНТ, УСПЕВАЕМОСТЬ, ПРЕДМЕТ, ПРЕПОДАВАТЕЛЬ, формирующий сведения об оценках, полученных студентами по различным предметам. Для этого в режиме конструктора создадим схему данных запроса и бланк, как показано на рис.7.24. Сохраним этот запрос с именем "оценки".

В результате выполнения этого запроса будет получена таблица, источником записей которой является таблица УСПЕВАЕМОСТЬ, а расшифровывающие данные выбираются из таблиц: ПРЕДМЕТ, СТУДЕНТ, ПРЕПОДАВАТЕЛЬ. Таким образом, каждая строка результата будет содержать информацию об одной оценке, полученной студентом по указанному в строке предмету. Число строк в таблице запроса будет равно числу строк в таблице УСПЕВАЕМОСТЬ.

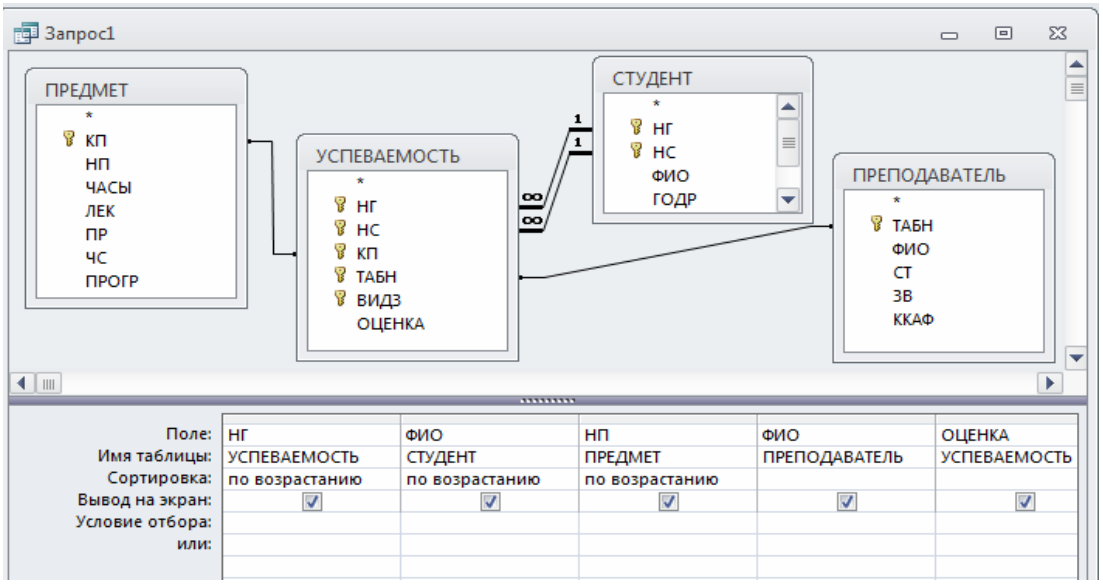


Рис.7.24. Многотабличный запрос об оценках студентов

Для подсчета числа различных оценок (2,3,4,5) по каждому из предметов на основе этого запроса создадим новый запрос - **Число оценок**. При создании нового запроса в окне **Отразить таблицу/Добавление таблицы** на вкладке **Запросы** выберем из списка запрос **оценки**. Заполним бланк запроса, как показано на рис.7.25.

Результат выполнения запроса **Число оценок** приведен на рис. 7.26, где в столбце **Выражение 1** отображено количество оценок, полученных по каждому предмету.

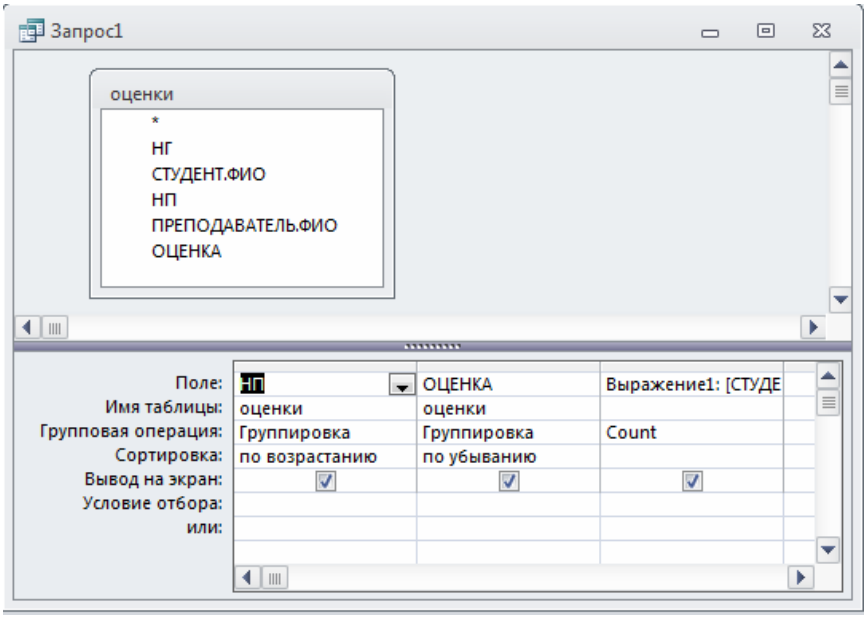


Рис.7.25. Построение запроса на основе запроса **оценки**

Замечание. Нет необходимости предварительно выполнять запрос (**оценки**), на основе которого выполняется другой запрос (**Число оценок**). Выполнение вложенного запроса инициируется системой при выполнении запроса, построенного на нем.

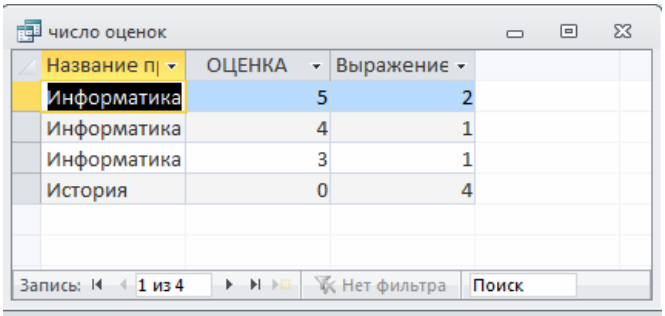


Рис. 7.26. Результаты запроса, построенного на другом запросе

Упражнения

- Создайте на основе запроса оценки, приведенного на рис.7.24, запрос для анализа оценок, выставленных каждым из преподавателей. Результат запроса должен содержать количество оценок (2,3,4,5), выставленных каждым преподавателем. Подпись столбца с результатами выполнения групповой операции Count Выражение1 замените на Количество оценок
- Создайте на основе запроса оценки запрос для определения числа студентов, получивших 2,3,4 или 5 по предмету, задаваемому в диалоге с пользователем
- Создайте на основе запроса оценки запрос для подсчета числа студентов в группе, получивших 2 (или другую заданную оценку) по каждому предмету. Предусмотрите ввод номера группы и оценки в диалоге с пользователем
- Создайте на основе запроса оценки запрос для подсчета средней оценки в группе по каждому предмету; средней величины оценок, выставленных преподавателем; средней успеваемости по каждому предмету

Решение задачи, требующей выполнения нескольких запросов и сохранения промежуточных результатов

Пусть необходимо определить среднюю нагрузку преподавателя кафедры в текущем семестре. Для этого необходимо подсчитать число преподавателей кафедры, затем общее количество часов занятий, проводимых кафедрой, и завершить решение задачи расчетом средней нагрузки преподавателя.

Подготовим и последовательно выполним соответствующие запросы.

Первый запрос. Создадим первый запрос на выборку, в котором по таблице ПРЕПОДАВАТЕЛЬ с помощью функции Count подсчитаем число преподавателей по кафедрам. Сохраним запрос под именем "Число преподавателей кафедры".

Второй запрос. Подготовим второй запрос на выборку для подсчета общего числа часов занятий, проводимых каждой кафедрой.

Этот запрос построим на базе таблиц ПРЕПОДАВАТЕЛЬ, ИЗУЧЕНИЕ, КАФЕДРА и запроса **Число преподавателей кафедры** (рис.7.25). Таблицы ПРЕПОДАВАТЕЛЬ и ИЗУЧЕНИЕ нужны для суммирования числа часов занятий, проводимых преподавателями каждой кафедры. Таблица КАФЕДРА необходима для включения в результат наименования кафедры, а запрос **Число преподавателей кафедры** - для включения в результат числа преподавателей на кафедре.

Записи этого запроса формируются на основе записей таблицы ИЗУЧЕНИЕ, причем число записей до проведения группировки равно числу записей в этой таблице. В результате запроса к каждой записи добавляется наименование кафедры НКАФ, по которому и производится группировка. Число преподавателей кафедры Count_ТАБН никак не нарушает требуемого объединения записей в группы, посеольку для каждой кафедры является единственным. Число записей в таблице результата запроса равно числу кафедр.

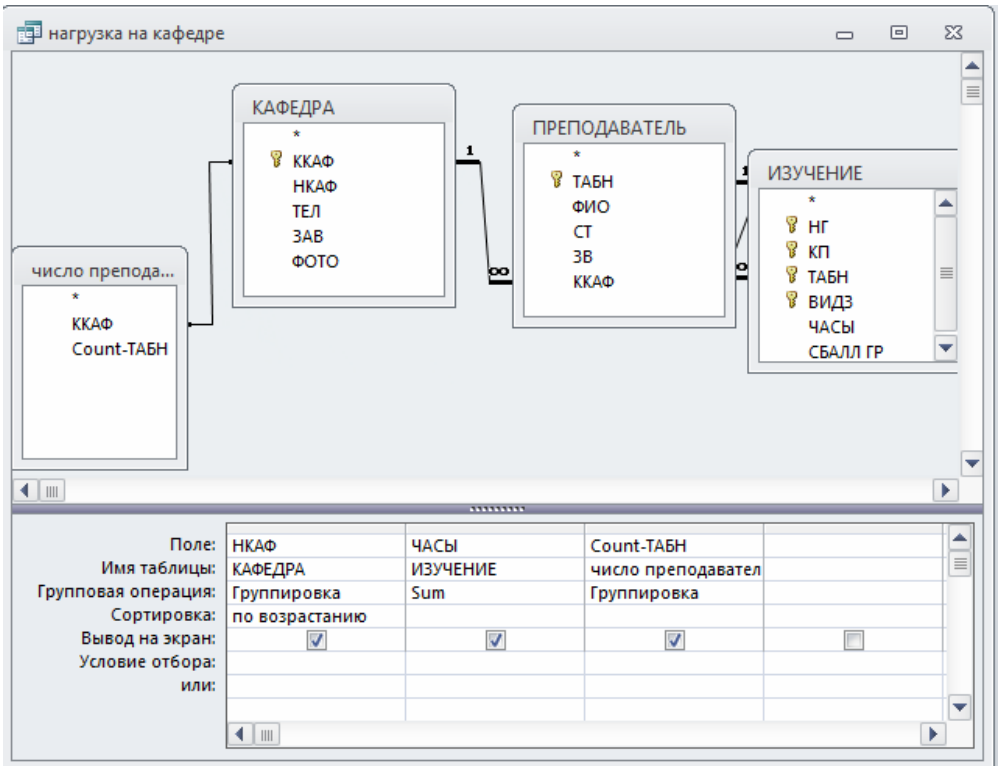


Рис. 7.27. Запрос для подсчета общего числа часов занятий, проводимых каждой кафедрой

В рамках данного запроса нельзя сразу вычислить среднюю нагрузку преподавателя, поскольку число преподавателей кафедры является результатом выполнения групповой операции. Использование результатов выполнения запроса с групповой операцией не допускается в вычисляемых полях. Поэтому необходимо сохранить результаты выполнения второго запроса в новой таблице и на ее основе построить следующий запрос, в котором будет произведен расчет средней нагрузки преподавателя.

Чтобы сохранить полученные результаты, преобразуем второй запрос на выборку в запрос на создание таблицы. Таблице, которая будет создана запросом, присвоим имя "Нагрузка". Запрос сохраним под именем " Нагрузка на кафедре" (см. рис.7.27).

Второй запрос в режиме таблицы приведен на рис. 7.28. Таблица НАГРУЗКА, сохраняемая в базе данных, содержащая результат решения этого запроса, приведена на рис. 7.29.

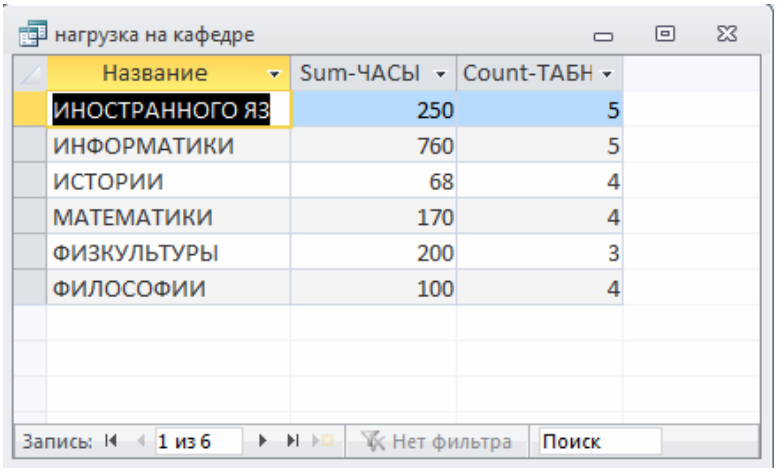


Рис. 7.28. Запрос **Нагрузка на кафедре** в режиме таблицы

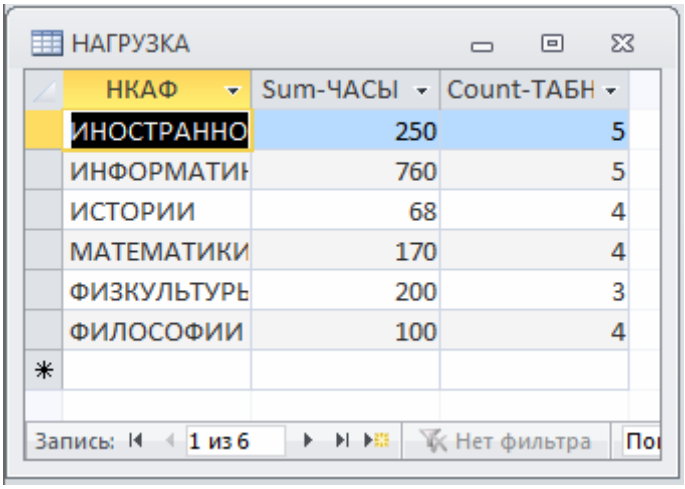


Рис. 7.29. Сохраненная таблица НАГРУЗКА с результатами второго запроса

Третий запрос. Для окончательного решения задачи расчета средней нагрузки преподавателя кафедры подготовимна базе таблицы НАГРУЗКА третий запрос на выборку с вычисляемым полем. Для создания вычисляемого поля, расчитывающего среднюю нагрузку преподавателя, в строку **Поле** пустого столбца введем выражение [Sum_ЧАСЫ]/[Count_ТАБН]. Третий запрос на выборку с вычисляемым полем представлен на рис. 7.30.

В таблице результата следует изменить заголовок столбца **Выражение1**, формируемый по умолчанию для вычисляемого поля, и его формат (для получения результата с округлением до целого). Для этого вызовем свойства поля с помощью контекстно-зависимого меню. Зададим в качестве подписи поля значение "Средняя нагрузка преподавателя", формат поля определим как фиксированный, а параметру **Число десятичных знаков** присвоим значение "0"

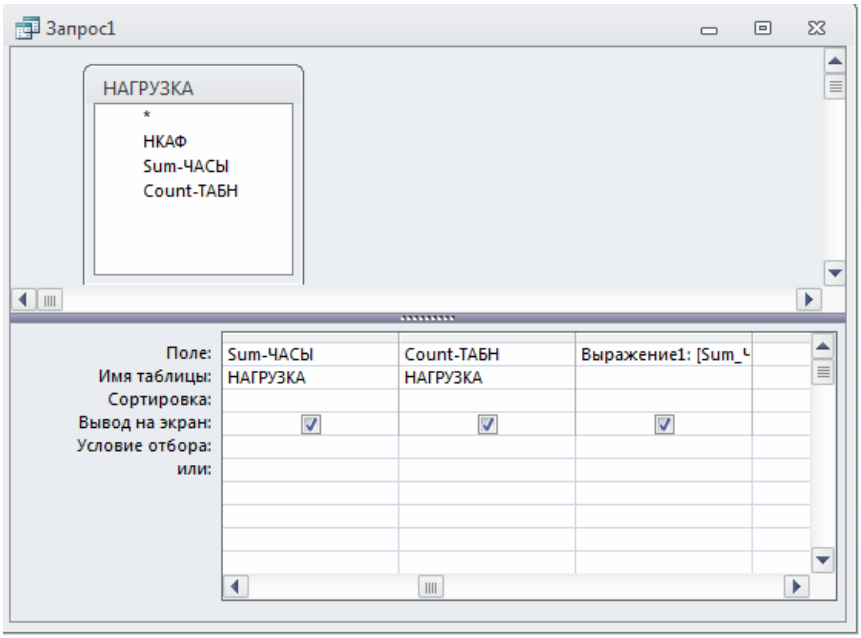


Рис. 7.30. Запрос, завершающий задачу расчета средней нагрузки

Результаты выполнения запроса после изменения подписей полей в свойствах приведены на рис. 7.31.

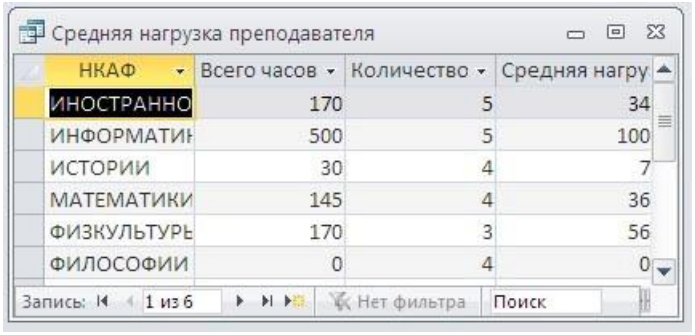


Рис. 7.31. Результаты расчета средней нагрузки преподавателей по кафедрам

Процесс решения задачи. Для решения поставленной задачи необходимо последовательно выполнить второй запрос - **Нагрузка на кафедре** (см. рис.7.25), а затем третий запрос - (см. рис. 7.30). Первый запрос **Число препо-**

давателей кафедры выполнится автоматически при выполнении запроса **Нагрузка на кафедре**. Для того чтобы автоматизировать решение задачи, требующей выполнения нескольких запросов, нужно использовать средства разработки приложения пользователя, например, написать макрос. Технология подготовки макроса, который позволяет выполнить последовательность запросов, рассмотрена ниже.

Урок 8. Отчет по одной таблице

Рассмотрим технологию создания однотабличного отчета на примере получения списков студентов по группам.

Пусть в результате проектирования макета отчета СПИСКИ СТУДЕНТОВ определены перечисленные ниже требования. Макет формируемого отчета должен иметь вид в соответствии с рис.8.1. На макете показано оформление списка студентов для одной группы. В отчете должны последовательно выводиться со своими заголовками списки студентов для каждой группы. При формировании отчета необходимо рассчитать средний проходной балл для каждой группы и отобразить его в отчете. Записи списка группы должны выводиться в порядке возрастания номера студента в группе. Название отчета должно выводиться на каждой странице отчета.

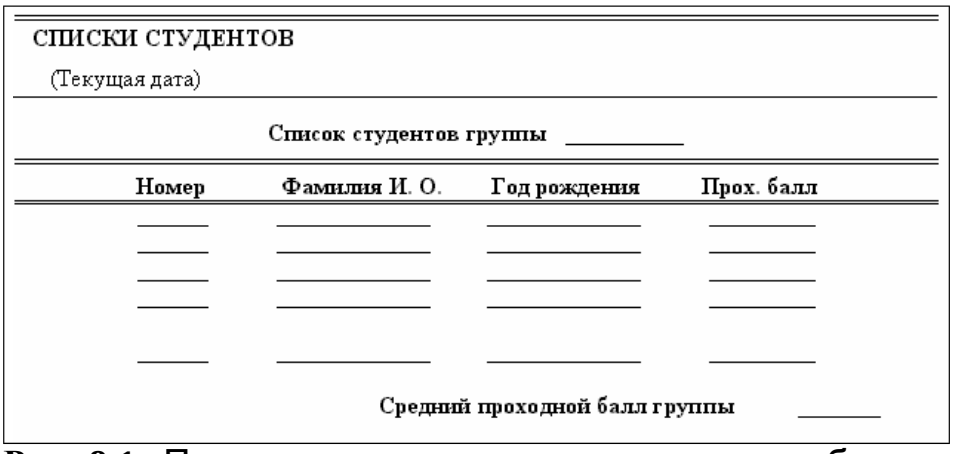


Рис. 8.1. Проект макета для создания однотабличного отчета со списками студентов по группам

Создание однотабличного отчета в режиме конструктора

В пункте меню **Создание** нажимаем кнопку **Конструктор отчетов**. В области данных вызываем контекстное меню, выбираем пункт **Свойства**. В открывшемся окне выбираем вкладку **Данные** и заполняем соответствующее поле именем таблицы СТУДЕНТ, которая будет служить источником записей для нашего отчета. В пункте меню Конструктор нажимаем кнопку **Добавить поля**.

Если в открывшемся окне конструктора отсутствует раздел Заголовок отчета, то вызываем контекстное меню и выбираем пункт **Заголовок/ примечание отчета**.

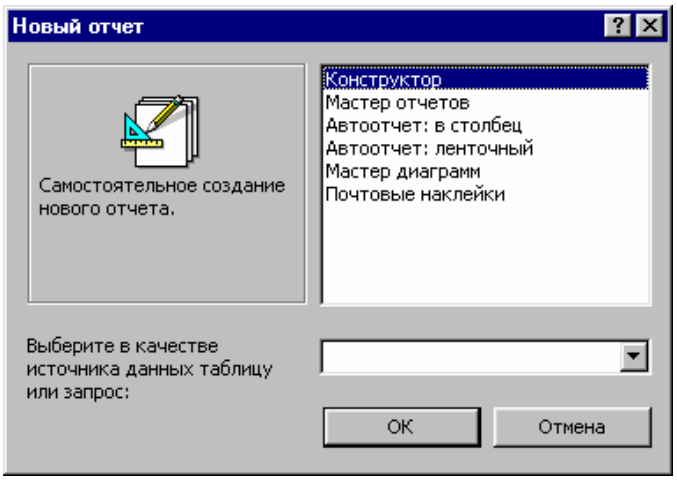


Рис.8.2. Окно выбора варианта создания отчета и таблицы-источника данных отчета

Группировка и сортировка данных отчета

Для выполнения требования к группировке и сортировке данных, отображаемых в отчете, нажмем кнопку **Группировка** на панели **Группировка и итоги** конструктора и зададим необходимые параметры в открывшемся диалоговом окне **Группировка, сортировка и итоги** (рис.8.3).

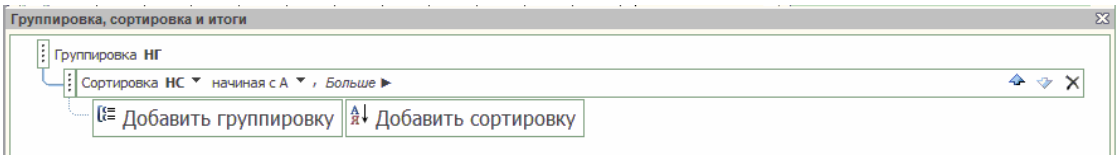


Рис.8.3. Окно определения групп записей по полю НГ и сортировке по полю НС

Группировка по полю. Поскольку общий список студентов в соответствии с проектом отчета должен быть разбит по группам, выберем в окне **Сортировка и группировка** (Sorting and grouping) из списка поле номера группы НГ, зададим группировку по этому полю. Для этого в области **Свойства группы** строках **Заголовок группы** (НГ Header) и **Примечание группы** (НГ Footer) надо выбрать значения **с разделом заголовка, с разделом примечания**. Сортировка для поля устанавливается автоматически.

Сортировка по полю. Для вывода отсортированного списка студентов в каждой группе, зададим сортировку по полю номера студента НС. Для этого в диалоговом окне выберем наряду с полем НГ поле НС. В области **Свойства группы** этого поля в строках **Заголовок группы** (НГ Header) и **Примечание группы** (НГ Footer) надо выбрать значения **без раздела заголовка, без раздела примечания**, что и определяет сортировку только по этому полю.

После определения группировки в окне конструктора отчетов (рис.8.4) появляются дополнительные разделы **Заголовок группы НГ** (НГ Header), **Примечание группы НГ** (НГ Footer).

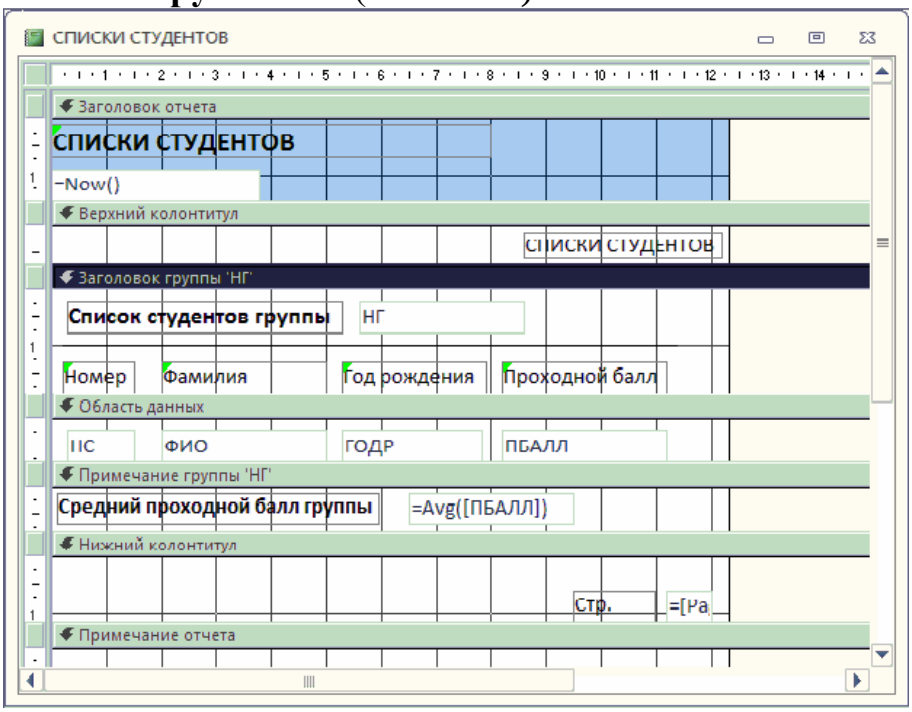


Рис.8.4. Окно конструктора отчетов при разработке отчета для вывода данных из одной таблицы СТУДЕНТ

Размещение данных в разделах отчета

Данные в отчете должны быть размещены, как показано на проекте макета отчета (см. рис. 8.1).

Размещение полей из таблиц

Размещение поля группировки. Значение номера группы должно быть представлено один раз в заголовке группы. Для этого разместим поле НГ в разделе **Заголовок группы НГ** (НГ Header).

Нажмем кнопку панели инструментов конструктора отчетов **Добавить поля** и перетащим поле НГ в раздел заголовка НГ. Откорректируем подпись поля, изменив ее на “Список студентов группы” (рис.8.4). Установим нужный шрифт в элементах. Для установки размеров рамки по размеру текста подписи

выполним команду контекстного меню **Размер|по размеру данных** (Size|Size To Fit) или соответствующую кнопку панели инструментов.

Форматирование табличной части отчета. Последовательно разместим поля НС, ФИО, ДАТАР, ПБАЛЛ в области данных, которая определяет содержимое строк табличной части. Поле размещается вместе с подписью, которую система берет из свойств полей таблицы СТУДЕНТ. Подписи полей надо перенести в область заголовка путем вырезания и вставки. Если они не совпадают с названиями столбцов в проекте макета, их надо откорректировать. Заметим, что подписи также можно создать заново, воспользовавшись кнопкой панели элементов **Надпись** (Lable).

Включение вычисляемого поля в отчет

Для включения расчетного реквизита **Средний проходной балл группы** нажмем кнопку **Поле** (Text Box) на панели элементов и разместим элементы **Свободный** (Unbound) в раздел **Примечание группы НГ** (НГ Footer) (см. рис. 8.4). Определим в свойствах этого элемента выражение для расчета среднего значения. Для этого запишем на вкладке **Данные** (Data) в строку **Данные** (Control Source) функцию =Avg ([ПБАЛЛ]), в строку **Число десятичных знаков** (Decimal Places) – “2”, на вкладке **Макет** (Format) в строку **Формат поля** (Format) поместим значение “Фиксированный”(Fixed). Отредактируем подпись поля. Для этого выделим подпись и вызовем ее свойства. В свойствах на вкладке **Макет** (Format) в строке **Подпись** (Caption) запишем: “Средний проходной балл группы”. Такие действия, как изменение подписи или ввод выражения в поле можно выполнить, и не обращаясь к свойствам элементов.

Добавление текущей даты к странице

Для добавления в отчет *текущей даты* воспользуемся встроенной функцией Now(). Для этого создадим в заголовке отчета свободный элемент, нажав кнопку **Поле** (Text Box), и зададим в окне его свойств на вкладке **Данные** (Data) в строке **Данные** (Control Source) выражение =Now(). На вкладке **Макет** (Format) в строке **Формат поля** (Format) выберем значение **Полный формат даты** (General Date). Подпись этого поля выделим и удалим.

Для добавления *номера страницы* в нижний колонтитул создадим свободный элемент и заполним в его свойствах на вкладке **Данные** (Data) строку **Данные** (Control Source) выражением =[Page]. Отредактируем подпись этого поля, за-

писав в его свойствах на вкладке **Макет** (Format) в строке **Подпись** (Caption) значение “Стр”.

Рассмотрим другие способы формирования поля даты и номера страницы. Поле текущей даты и времени можно добавить в отчет, выполнив в режиме конструктора команду **Дата и время** (Date and Time…). Установка в диалоговом окне **Дата и время** (Date and Time) флажков **Формат даты** (Include Data) и /или **Формат времени** (Include Time) позволяет вставить текущую дату и/или текущее время и выбрать нужный формат (рис.8.5).

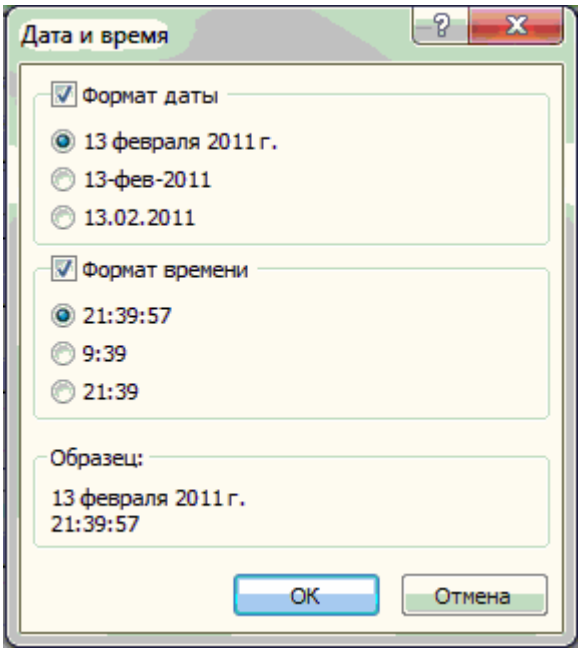


Рис.8.5. Окно выбора формата даты и/или времени

В отчет будет добавлено поле, в свойствах которого на вкладке **Данные** (Data) в строке **Данные** (Control Source) будет записано соответствующее выражение. Если в отчете имеется раздел заголовка, поле добавляется в этот раздел. В противном случае поле вносится в раздел данных. В качестве выражения записывается функция Format, которая формирует значение на основе заданных ей аргументов – функции Date(), возвращающей текущую системную дату, и формата, в котором должна выводится дата. Например, при выборе параметров, отображенных на рис.8.5, функция примет вид =Format(Date();”Long Date”).

Поле нумерации страниц можно добавить в отчет, выполнив в режиме конструктора команду **Номера страниц** (Insert|Page Number). В окне диалога **Номера страниц** (Page Numbers) выбираются параметры, определяющие формат, расположение и выраынивание номеров страниц (рис.8.6). Для печати номера

страницы на первой странице устанавливается флажок **Отображать номер на первой странице** (Show Number on First Page).

Замечание. Выражение, определяющее вывод номеров страниц, записывается в свойствах поля на вкладке **Данные** (Data) в строке **Данные**(Control Source). Выражение может иметь вид: =”Страница”&[Page]

или:
=”Страница” & [Page] & ”из” & [Pages],
что соответствует выбору **Страница N** (Page N) или **Страница N из M** (Page N of M).

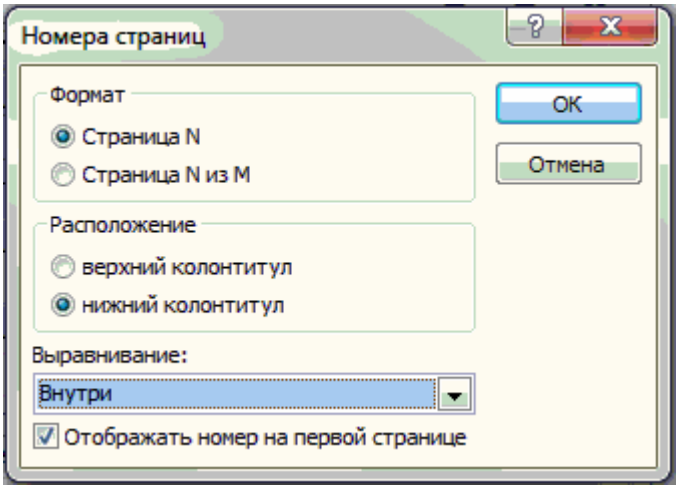


Рис.8.6. Окно выбора параметров номеров страниц.

Завершение оформления отчета

Для окончательного оформления введем в раздел **Заголовок отчета** (Report Header), вставить раздел **Верхний колонтитул** (Page Header) и выбрать нужный шрифт. Далее надо указать в свойствах отчета на вкладке **Макет** (Format) в строке **Верхний колонтитул** (Page Header):”Без заголовка” (Not with Rpt Hdr). Свойства отчета могут быть вызваны при установке курсора на пересечении линеек.

Создадим линии в соответствии с макетом, воспользовавшись кнопкой панели элементов **Линия** (Line).

Просмотр и печать отчета

Переход из режима конструкторов в режим предварительного просмотра осуществим, нажав кнопку Предварительный просмотр (Report View)

(см.рис.6.2). Для просмотра ранее созданного отчета нужно выбрать его в окне базы данных на вкладке Отчеты (Reports) и нажать кнопку Просмотр (Preview). Отчет при просмотре отобразится на экране таким, каким он будет напечатан. В режиме предварительного просмотра имеется своя панель инструментов (рис.8.7).

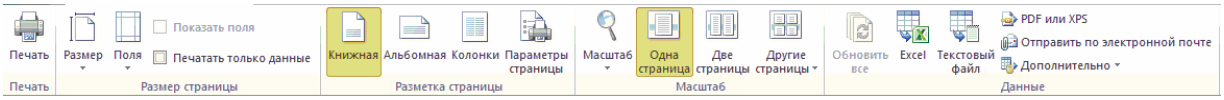


Рис.8.7. Панель инструментов предварительного просмотра отчета

Для просмотра нужных страниц отчета можно использовать стандартное *поле номера страницы* **В нижнем левом углу окна отчета.**
Кнопка **Печать** (Print) панели инструментов режима предварительного просмотра позволяет вывести отчет на печать.

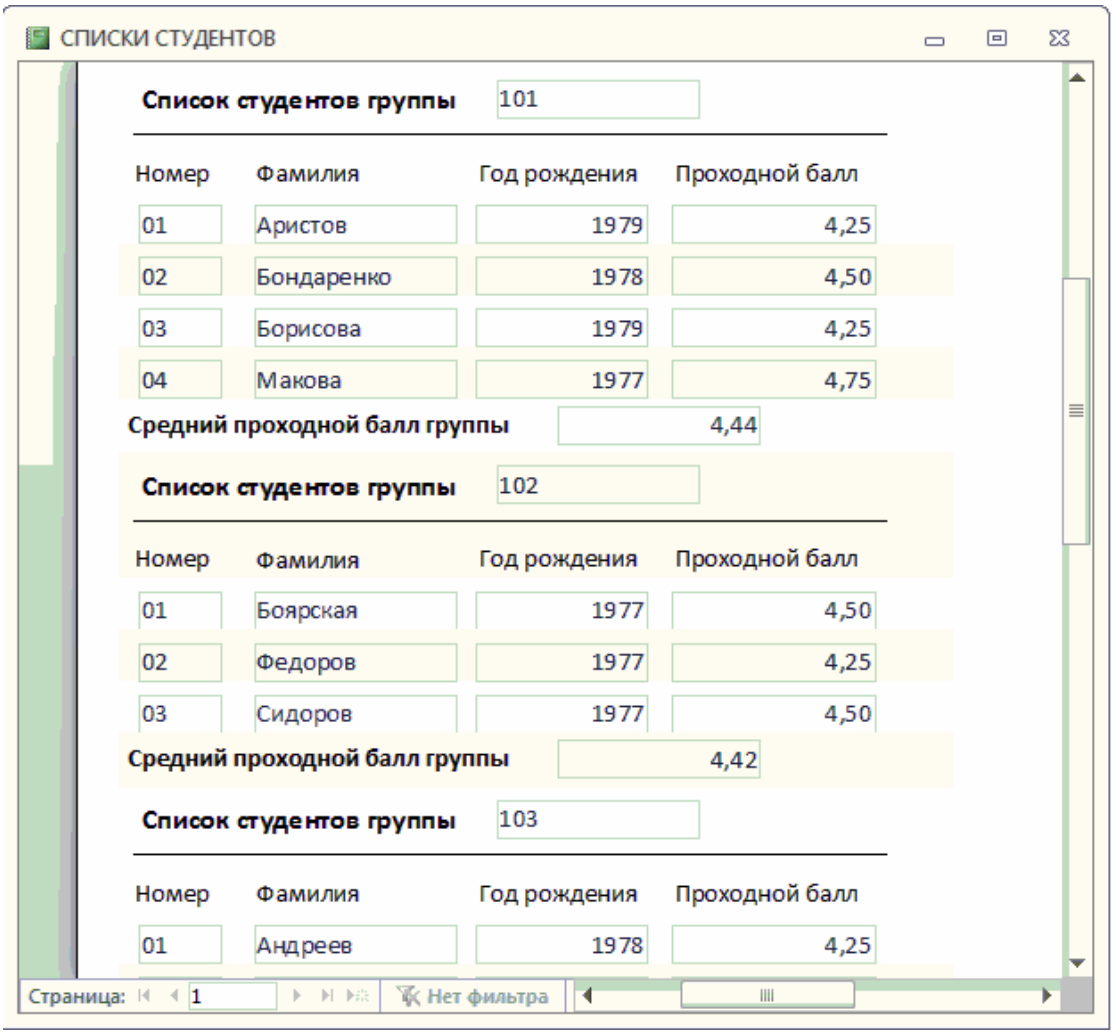


Рис.8.8. Первая страница отчета **Списки студентов** с данными из таблиц СТУДЕНТ в режиме предварительного просмотра.

С помощью команды **Файл**|**Параметры страниц** (File|Page Setup…) можно выбрать принтер, задать формат бумаги, размер полей, расстояние между строками, ориентацию (книжная, альбомная) и т.д. Команда **Файл**|**Печать** (File|Print…) позволяет выбрать для печати отдельные страницы отчета или выделенные записи, распечатать заданное число копий, вывести отчет в файл, который должен распечатываться в другое время. На рис.8.8 приведена первая страница разработанного отчета **Списки студентов** в режиме предварительного просмотра.

Урок 9. Отчеты по двум таблицам

Наряду с однотабличными отчетами Access позволяет создавать более сложные отчеты, обеспечивающие вывод данных из нескольких взаимосвязанных данных в многотабличный отчет автоматически используются связи, установленные в схеме данных БД.

Многотабличные отчеты могут содержать основную часть и включаемую часть, т.е. подчиненный. Для каждой из этих частей в качестве источника данных выбирается своя таблица или несколько таблиц и при этом не иметь подчиненных отчетов.

Рассмотрим далее построение отчета для двух таблиц, находящихся в одномногозначных отношениях. В этих отношениях одна таблица является главной, а другая – подчиненной.

Построение такого отчета имеет свои особенности в зависимости от выбора главной или подчиненной таблицы в качестве подчиненного отчета.

Многотабличный отчет с основной частью на базе главной таблицы

Рассмотрим технологию создания многотабличного отчета на основе таблиц КАФЕДРА и ПРЕПОДАВАТЕЛЬ. Пусть необходимо подготовить отчет, содержащий сведения о кафедрах и включающий списки преподавателей по кафедрам.

Проект макета отчета.

Предположим, что макет для отчета со списками преподавателей по кафедрам должен иметь вид, показанный на рис. 9.1. В соответствии с проектом макета в отчет предполагается выводить данные по каждой кафедре, включая название, код и телефон, а также фамилию и фотография заведующего. Эти данные содержатся в таблице КАФЕДРА.

В табличной части по каждой кафедре необходимо вывести данные о преподавателях кафедре, которые содержатся в таблице ПРЕПОДАВАТЕЛЬ.

Проект макета дает основание выбрать в качестве основной таблицы отчета таблицу КАФЕДРА, а таблицу ПРЕПОДАВАТЕЛЬ – в качестве источника данных для подчиненного отчета со списком преподавателей.

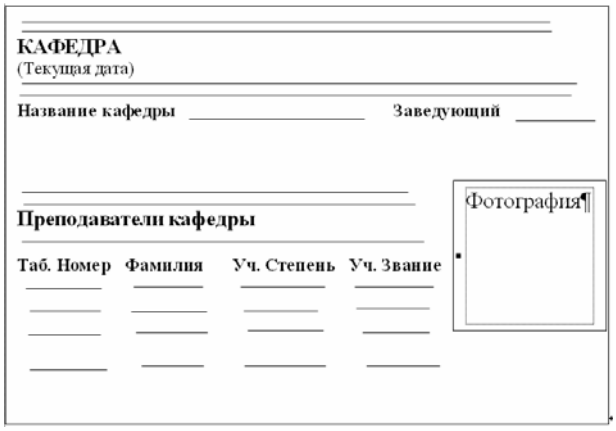


Рис. 9.1. Макет для отчета со списками преподавателей по кафедрам

Создание основной части отчета

В окне базы данных **Все объекты Access** выберем таблицу КАФЕДРА, которая будет источником данных для основной части отчета. Далее выберем вкладку **Создание** и нажмем на кнопку **Отчет**. Щелкаем правой кнопкой мыши по появившемуся отчету КАФЕДРА и выбираем **Конструктор** для построения отчета в режиме конструктора. В область данных перетащим поля таблицы КАФЕДРА: НКАФ, ККАФ, ТЕЛ, ЗАВ и ФОТО из верхнего колонтитула. Разместим поля и подписи, отредактируем их в соответствии с проектом макета на рис. 9.1. Сохраним отчет под именем «Кафедры» (см. рис. 9.2).

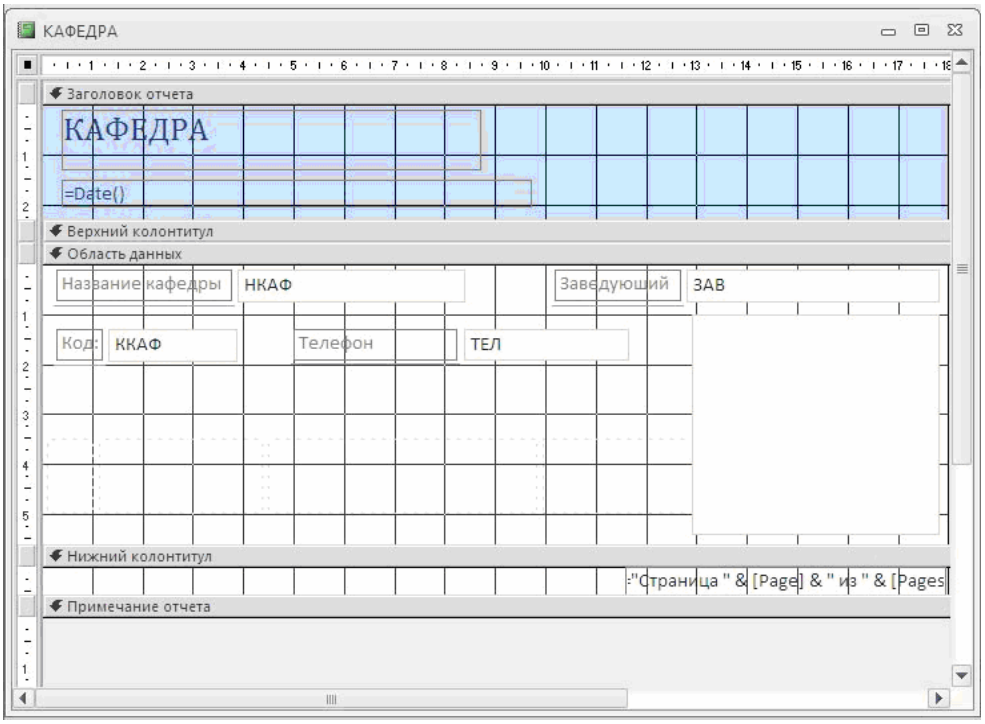


Рис. 9.2. Заполнение основной части отчета о кафедрах в режиме конструктора

Разработка подчиненного отчета

Для вывода в отчет **Кафедры** списка преподавателей из таблицы ПРЕПОДАВАТЕЛЬ подготовим отдельный отчет, который будет включен в основную часть отчета в качестве подчиненного.

Создание автоотчета

Подчиненный отчет создается как обычный однотабличный отчет. Для автоматического создания отчета во вкладке **Создание/Отчет** выберем опцию **Мастер отчетов.** В качестве источника данных этого отчета выберем таблицу ПРЕПОДАВАТЕЛЬ. Создание автоотчета завершается выводом на экран отчета, в котором в качестве заголовка фигурирует имя таблицы (рис. 9.3). В отчет включены все поля таблицы, а заголовками столбцов являются подписи этих полей, заданные в свойствах таблицы.

Доработка подчиненного отчета в режиме конструктора

Для того чтобы подчиненный отчет можно было включить в основную часть отчета в нужном виде в соответствии с проектом макета, доработаем его в режиме конструктора.

Верхний колонтитул, в котором после работы мастера оказались размещенными записи полей, и нижний колонктитул, где размещаются дата и номер страницы, не отобразятся при встраивании отчета как подчиненного. Для отображения в полном отчете заголовков столбцов табличной части перенесем название столбцов из верхнего колонтитула в заголовок подчиненного отчета.

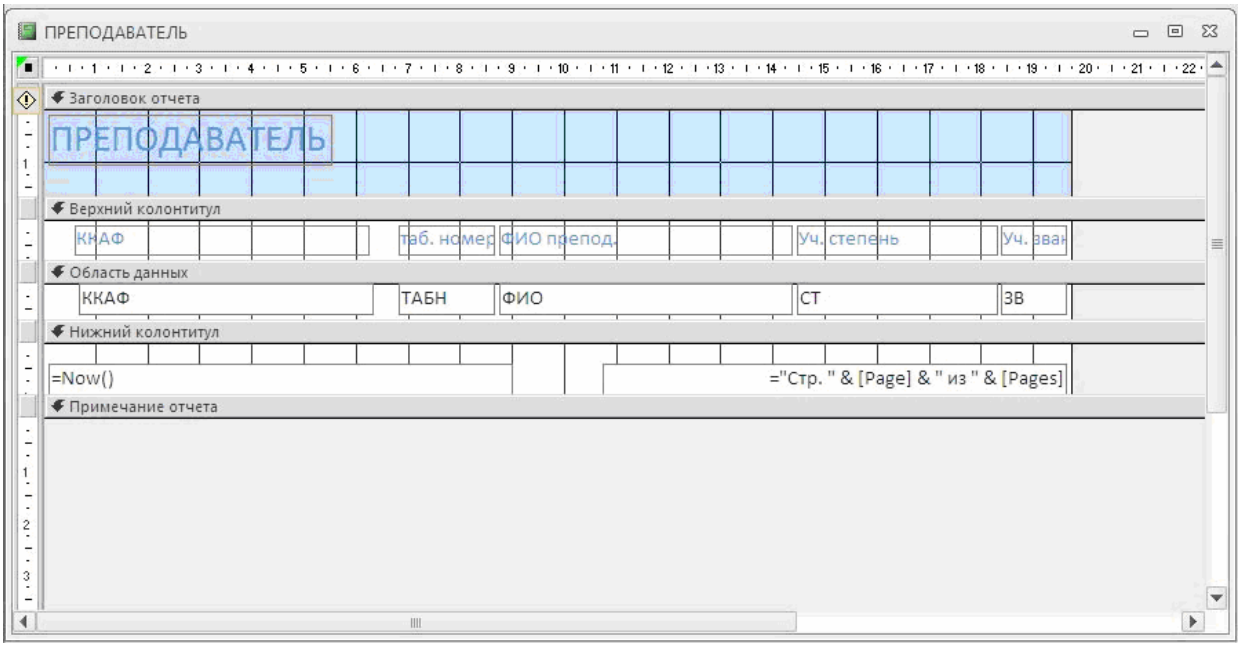


Рис. 9.3. Отчет, созданный в режиме **Автоотчет: ленточный**

Для этого расширим раздел заголовка отчета, выделим все подписи полей в верхнем колонтитуле и перетащим их. Поскольку отчет будет использоваться только как подчиненный, можно удалить оба колонтитула. Для этого нажмем правой кнопкой мыши на **Верхний** или **Нижний Колонтитул** и выберем **Колонтитулы страницы.** После этого в открывшемся окне нажимаем ДА. После этого оба колонтитула удалятся.

Удалим элемент поля ККАФ и его надпись «Код кафедры», т.к. в подчиненном отчете значения этого поля будут повторяться во всех строках о преподавателях, а однократное отображение кода кафедры предусмотрено в основной части отчета.

Окончательно сконструированный подчиненный отчет приведен на рис. 9.4. Сохраним подчиненный отчет под именем «Преподаватели».

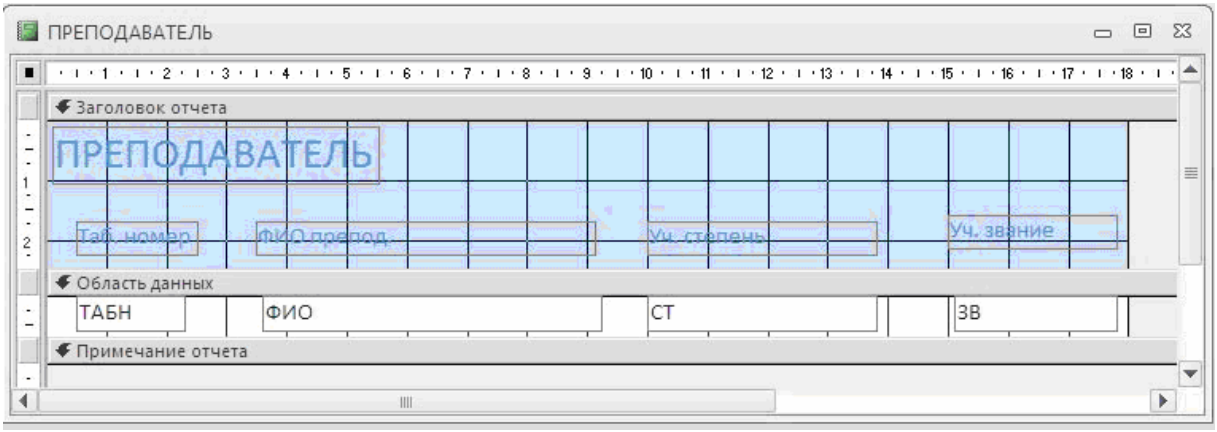


Рис. 9.4. Подчиненный отчет для включения в отчет **Кафедры**

Включение подчиненного отчета

Воспользуемся самым простым способом включения подчиненного отчета в основной.

Разместим на экране окно базы данных рядом с окном отчета **Кафедры** в режиме конструктора, который представлен выше на рис. 9.1.

Перейдем в окно базы данных. Выберем из перечня имен отчетов подчиненный отчет **Преподаватели** и перетащим его в область данных отчета **Кафедры.**

Удалим элемент надписи подчиненного отчета, выделив его и нажав <Del>. Отчет **Кафедры** после внедрения подчиненного отчета показан на рис. 9.5.

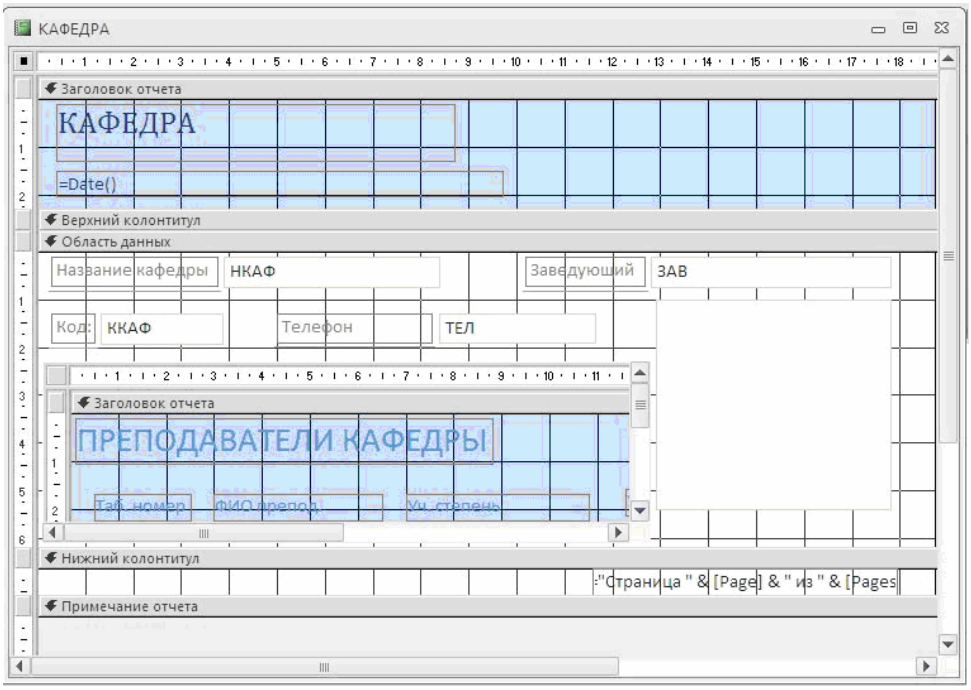


Рис. 9.5. Отчет на основе таблиц КАФЕДРА и РЕПОДАВАТЕЛЬ в режиме конструктора со встроенным подчиненным отчетом

На рис. 9.5 подчиненный отчет отображен внутри отчета **Кафедры,** где доступен для редактирования, как и основной отчет. Для того чтобы подчиненный отчет при отображении не был взят в рамку, выделим его, как видно на этом рисунке, откроем его свойства и на вкладке **Макет** в строке **Тип границы** выберем значение О**тсутствует**.Просмотрим содержимое отчета **Кафедры** (см. рис. 9.6.), нажав на него 2 раза левой кнопкой мыши, и убедимся, что его вид соответствует проекту макета отчета на рис. 9.1.

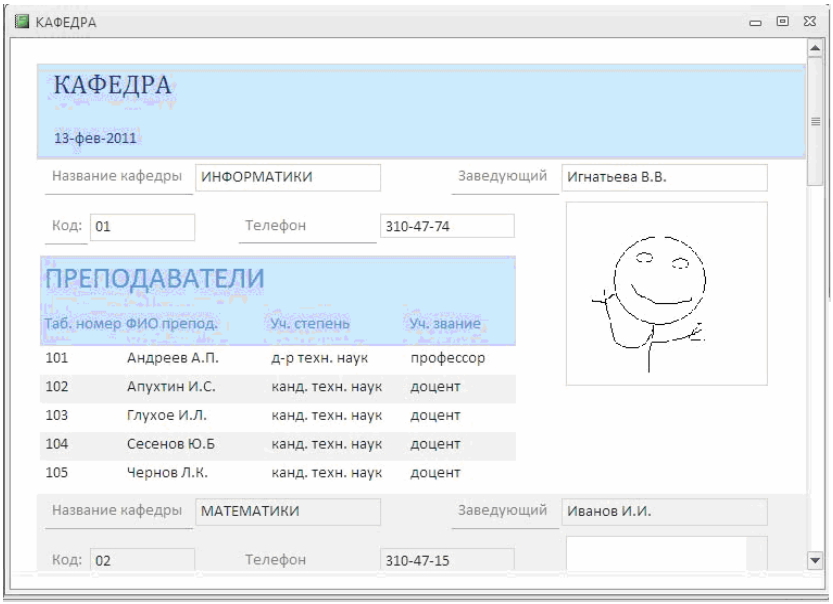


Рис. 9.6. Первая страница многотабличного отчета

Сортировка записей. Для вывода записей отчета в нужном порядке задается критерий сортировки. Для этого в режиме конструктора надо нажать кнопку **Итоги**. При необходимости сортировки и группировки записей подчиненного отчета необходимо выполнить для него такие же действия.

Урок 10. Многотабличные отчеты

Рассмотрим технологию разработки отчета, основным источником которого является подчиненная таблица, когда данные главных таблиц относительно этого источника тоже включаются в ответ. Мастер отчетов позволяет построить многотабличный отчет для взаимосвязанных таблиц, выбрать из них нужные поля в заданной последовательности и указать, какая таблица из участвующих в отчете, будет записеобразующей, т.е. основным источником данных. Кроме того, мастер предоставляет возможность определить группировку и сортировку записей отчета по различным полям, подсчитать итоговые значения.

Пусть необходимо получить отчет, в котором выводятся в виде списка данные о занятиях, проводимых в каждой группе. Строки отчета должны быть упорядочены по коду предмета. При выводе данных в отчете должны также формироваться расчетные суммарные часы по занятиям для каждой группы.

Проект макета отчета, который должен быть создан для вывода данных о занятиях, проводимых в каждой группе, приведен на рис. 10.1.

Рис. 10.1. Проект макета отчета о занятиях, проводимых в группах

Выбор таблиц для отчета и варианта его создания.

Основные сведения о занятиях, проводимых в группах, содержит подчиненная таблица ИЗУЧЕНИЕ. Из таблицы ИЗУЧЕНИЕ можно получить перечень

идентификаторов занятий, проводимых в каждой группе, с указанием часов по занятию. Эта таблица является основным источником записей для создаваемого отчета. Наименования предметов содержит таблица ПРЕДМЕТ, а фамилии преподавателей – таблица ПРЕПОДАВАТЕЛЬ. Обе эти таблицы являются главными по отношению к таблице ИЗУЧЕНИЕ. Одно-многозначные связи между этими таблицами и подчиненной таблицей ИЗУЧЕНИЕ установлены в схеме данных базы по соответствующим идентификаторам занятия: коду предмета и табельному номеру преподавателя.

Начиная создание отчета во вкладке **Создание,** выберем **Мастер отчетов.**

В окне **Создание отчетов** (см. рис. 10.2.) выберем из таблицы ИЗУЧЕНИЕ поля, включаемые в отчет: НГ, ВИДЗ, ЧАСЫ.

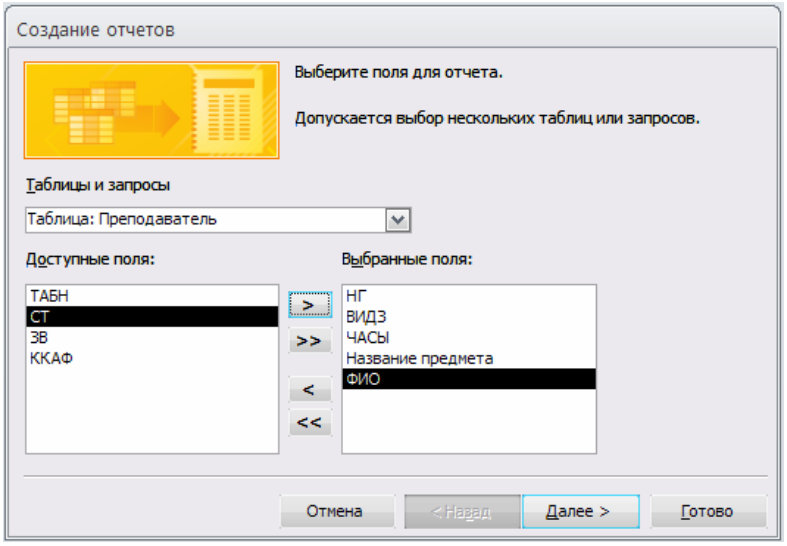


Рис. 10.2. Окно для формирования пользователем списка полей, включаемых в отчет

Из таблицы ПРЕДМЕТ выберем поле с наименованием предмета НП, а из таблицы ПРЕПОДАВАТЕЛЬ поле ФИО.

Замечание. Для того чтобы включать поля в нужной последовательности, следует иметь в виду, что поле вставляется в след за выделенным в списке полей уже включенных в ответ. Например, для включения поля НП сразу за полем НГ последнее должно быть предварительно выделено.

На рис. 10.2. представлено окно Создание отчета после выбора всех полей, используемых в отчете.

В следующем сеансе окна мастера в строке **Выберите тип представления данных**. Далее задаются уровни группировки, которые позволяют вывести

записи, объединенные по разным полям. Зададим один уровень группировки по полю НГ. При этом в окне отображается общая структура формируемого макета отчета.

В следующем сеансе диалогового окна мастера **Создание отчета** задается порядок сортировки записей (рис. 10.3).

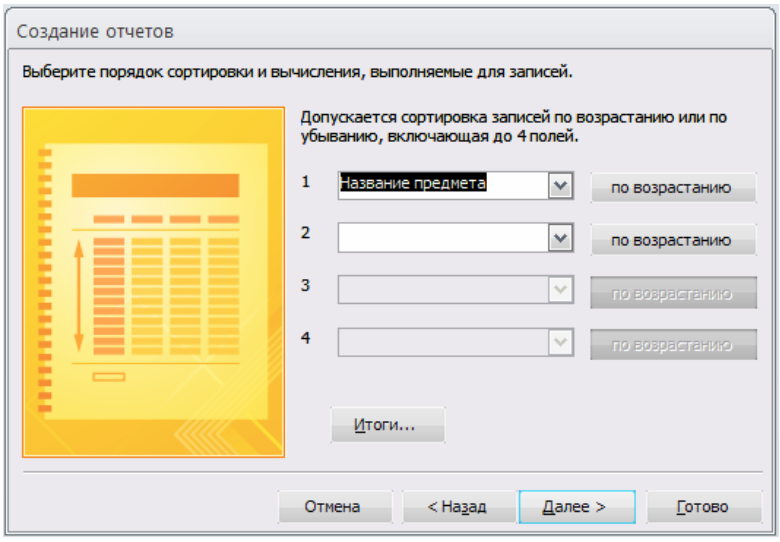


Рис. 10.3. Окно мастера отчетов при выборе порядка сортировки строк

Мастер позволяет производить сортировку по четырем полям в порядке возрастания или убывания значений поля. Выберем поле НП (Название предмета), по которому нужно произвести сортировку по возрастанию. Чтобы произвести подсчет итоговых значений, нажмем кнопку **Итоги.** Для числовых полей в открывшемся окне **Итоги** будут выведены строки, в которых можно выбрать статистическую функцию (Sum, Avg, Min, Max) для подсчета значений в итоговой строке группы. В соответствии с проектом макета отчета (см. рис. 10.1) должно быть выведено суммарное число часов в поле отчета **Итого по группе.** Поэтому выберем для поля ЧАСЫ функцию Sum.

Замечание. Если необходимо подсчитать долю суммарных часов группы от общих часов всех групп, нужно отметить флажок **Вычислить проценты**.

Далее в следующем сеансе окна мастера (рис. 10.4) выберем из шести предлагаемых видов макета отчета **Ступенчатый** с **Книжной** ориентацией. Отметим флажок **Настроить ширину полей для размещения на одной странице.**

Потом зададим имя отчета – «Изучение предметов в группах», которое также отобразиться в заголовке отчета. Под этим именем мастер автоматически сохраняет отчет в базе данных. Для того чтобы сразу доработать отчет, выберем дальнейшие действия – **Изменить макет отчета.** Отчет отобразиться на экране в режиме конструктора (рис. 10.5).

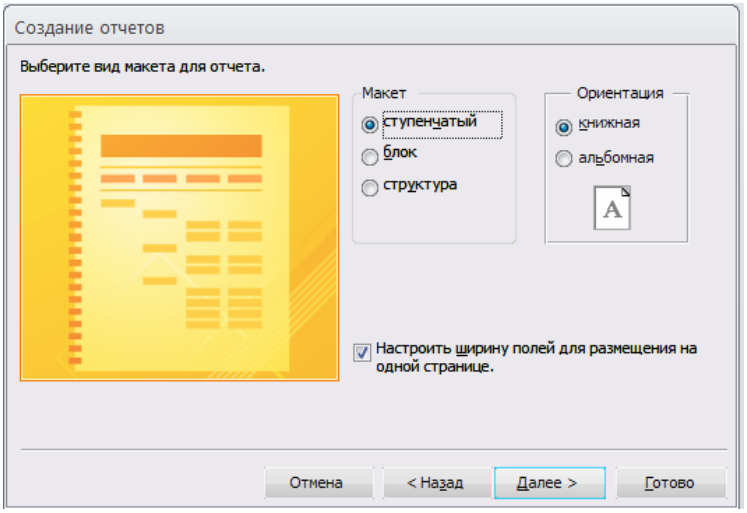


Рис. 10.4. Окно мастера отчетов при задании вида макета отчета

Этот отчет по основным параметрам соответствует проекту макета отчета (см. рис. 10.1). в него включены поля из трех взаимосвязанных таблиц. Причем пользователь не потребовалось задавать связи между таблицами и включать в отчет поля КП (код предмета) и ТАБН (номер преподавателя), являющиеся полями связи.

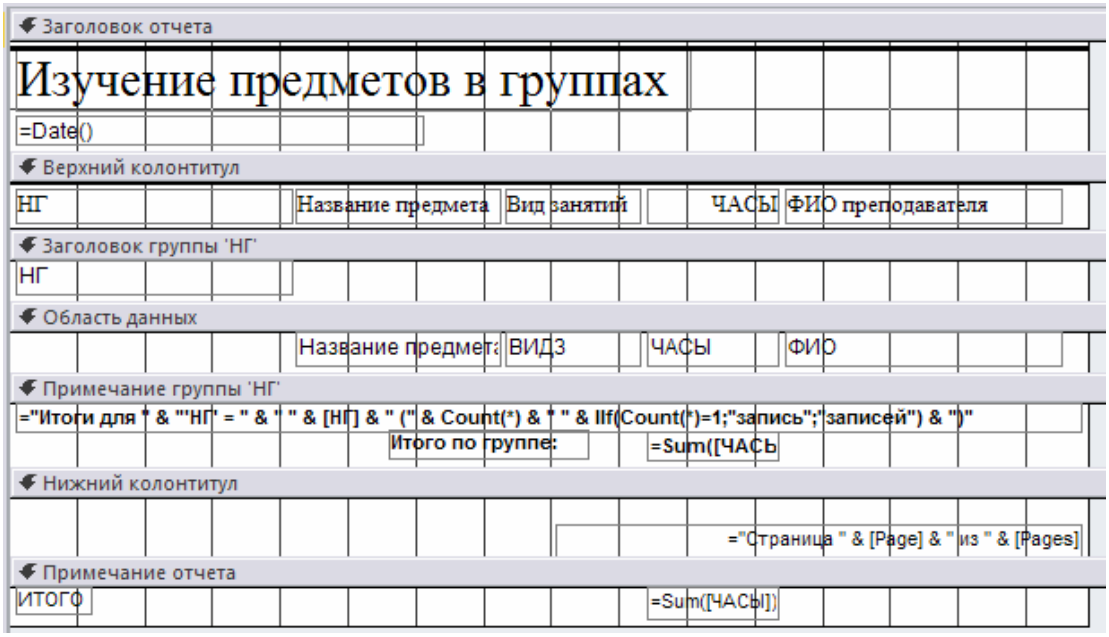


Рис. 10.5. Макет отчета о занятиях, проводимых в группах, построенный мастером и отображенный в режиме конструктора

Отметим, что на основе информации, сохраняемой в схеме данных, и заданных пользователем полей при создании макета отчета, мастер сам строит необходимый запрос. По этому запросу формируются записи из полей нескольких взаимосвязанных таблиц. В свойствах отчета (рис. 10.6) в качестве источ-

ника записей мастер записывает инструкцию SQL, реализующую запрос и определяющую выборку заданных полей из различных таблиц.

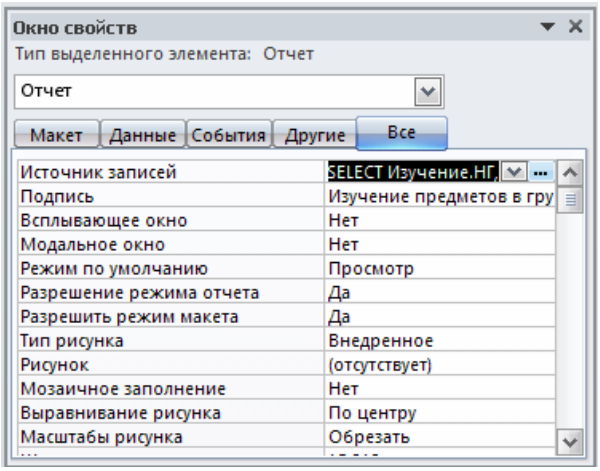


Рис. 10.6. Окно свойств многотабличного отчета **Изучение предметов в группах**

Запрос, созданный мастером при подготовке макета отчета, можно просмотреть и при необходимости откорректировать. Чтобы отобразить запрос на экране, нужно в диалоговом окне свойств отчета в строке **Источник записей** нажать кнопку **Построитель,** которая вызовет построитель запросов. Открывающееся окно построителя запросов представлено на рис. 10.7.

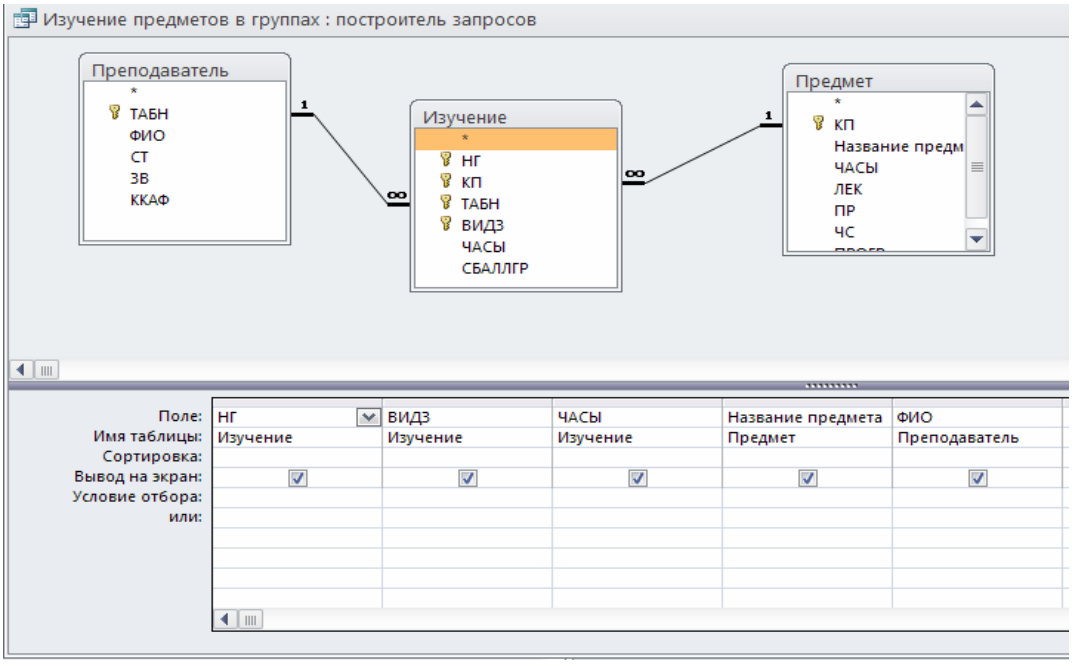


Рис. 10.7. Запрос, подготовленный мастером отчетов при создании макета отчета

Редактирование макета отчета в режиме конструктора. В отчет, построенный мастером (см. рис. 10.5), в заголовок группы автоматически было вклю-

чено поле номера группы НГ. В примечание группы мастером включены итоги по группе: поле для подсчета числа записей в группе и поле для подсчета суммы часов по группам. Отредактируем надписи этих полей.

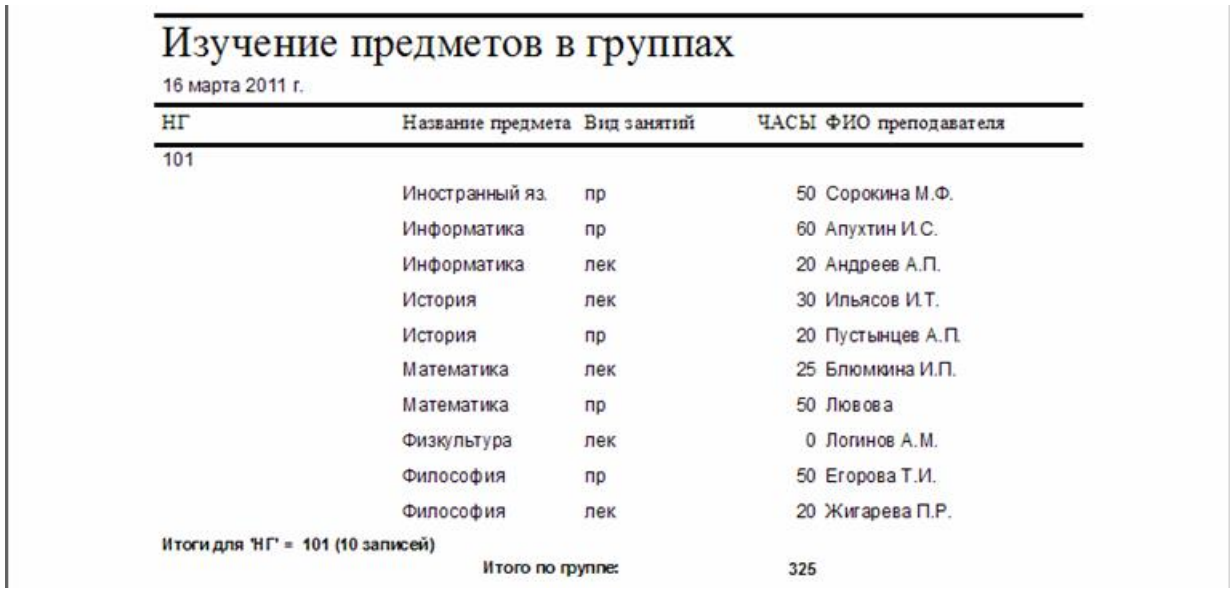
Поместим поле с выражением =Date(), определяющим текущую дату, в заголовок отчета.

Элементы заголовков столбцов размещены в верхнем колонтитуле отчета и, следовательно, будут печататься на каждой странице отчета. Чтобы заголовки печатались в каждой группе, как предусмотрено в проекте макета отчета на рис. 10.1, переместим их в заголовок группы.

Если необходимо, чтобы сведения о каждой группе печатались на отдельной странице, вставьте в примечание группы разрыв страницы. Для этого воспользуйтесь кнопкой **Разрыв страницы** на панели Конструктора отчетов.

Отредактируем текст заголовков в соответствии с проектом макета. Отчет в режиме просмотра после доработки представлен на рис. 6.25.

Замечание. В отчетах можно вывести значения некоторого поля записи или итогового поля группировки нарастающим итогом. Например, можно накапливать сумму часов от группы к группе. Так, если в первой группе суммарное число часов равно 262, во второй – 150, а в третьей – 130, то задав свойство поля **Сумма с накоплением = Для всего**, можно получить значения: для первой группы 262, для второй 412, для третьей 542. Чтобы накапливать сумму значений поля для записей в группе, нужно установить свойство **Сумма с накоплением = Для группы.** Это свойство размещено на вкладке **Данные.**



Изучение предметов в группах

16 марта 2011 г.

НГ	Название предмета	Вид занятий	ЧАСЫ	ФИО преподавателя
101				
	Иностранный яз.	пр	50	Сорокина М.Ф.
	Информатика	пр	60	Апухтин И.С.
	Информатика	лек	20	Андреев А.П.
	История	лек	30	Ильясов И.Т.
	История	пр	20	Пустынцев А.П.
	Математика	лек	25	Блюмкина И.П.
	Математика	пр	50	Любова
	Физкультура	лек	0	Логинов А.М.
	Философия	пр	50	Егорова Т.И.
	Философия	лек	20	Жигарева П.Р.
Итоги для 'НГ' = 101 (10 записей)				
	Итого по группе:		325	

Рис. 6.25. Многотабличный отчет (начало) об изучении предметов в группах

Урок 11. Разработка отчета на основе запроса.

Запрос является мощным и удобным средством выборки взаимосвязанных данных. Поэтому с помощью запроса можно подготовить данные для сложного отчета.

Рассмотрим технологию создания сложного отчета с использованием запроса на примере формирования бланка “Экзаменационная ведомость” для группы студентов по заданному предмету. Бланк должен иметь форму документа, используемого для ввода данных после внесения оценок преподавателем в этот бланк.

Проект макета отчета. Бланк “ Экзаменационная ведомость”, которая должна выводиться из базы данных в режиме просмотра и печати отчета.

Экзаменационная ведомость

Название предмета Группа

Преподаватель

Вид сдачи 29-май-03

Nп/п	Фамилия И.О.	Отметка о сдаче	Подпись преподавателя

Итого: Отлично_____

Хорошо_____

Удовлетворительно_____

Неудовлетворительно_____

Неявка_____

Подпись преподавателя _____

Рис. 11.1. Проект макета отчета **Экзаменационная ведомость.**

Создание запроса для подготовки данных в отчет

Для вывода в отчет реквизитов, указанных в проекте макета нужно определить таблицы – источники и их взаимосвязи.

Таблицы источники данных. Для вывода в шапку бланка экзаменационной ведомости значений реквизитов **Вид сдачи**, **Группа** необходимы данные из полей ВИДЗ, НГ таблицы ИЗУЧЕНИЕ. Для вывода значений реквизитов с общей надписью **Преподаватель (фамилия, ученое звание)** необходимы

данные из полей ФИО, ЗВ таблицы ПРЕПОДАВАТЕЛЬ. Для вывода реквизита **Название предмета** необходимы данные из поля НП таблицы ПРЕДМЕТ.

Для вывода в табличную часть бланка экзаменационной ведомости значений реквизитов **N п/п, Фамилия И.О.** необходимы данные из полей НС, ФИО таблицы СТУДЕНТ.

Таким образом, запрос для выборки этих взаимосвязанных данных должен быть построен на основе таблицы ИЗУЧЕНИЕ, ПРЕПОДАВАТЕЛЬ, ПРЕДМЕТ, СТУДЕНТ.

Связи между таблицами запроса. При создании запроса связи между таблицами установятся автоматически. Связи таблиц ИЗУЧЕНИЕ, ПРЕДМЕТ, ПРЕПОДАВАТЕЛЬ определяются в соответствии со схемой данных БД.
При создании запроса между таблицами СТУДЕНТ и ИЗУЧЕНИЕ автоматически установится также связь по одноименному полю НГ (номер группы). Эта связь является *связью – объединением*, которой нет в схеме данных базы. Заметим, что эти таблицы находятся в отношениях многие-ко-многим, поскольку один студент изучает много предметов, и один предмет изучается многими студентами. Связь, установленная между таблицами СТУДЕНТ и ИЗУЧЕНИЕ, определяет операцию симметричного объединения. При этом записи из этих таблиц объединяются и добавляются в результат только в том случае, если связанные поля содержат одинаковые значения.

Конструирование запроса для подготовки макета отчета.

Процесс включения в запрос необходимых таблиц и полей из них является достаточно простым и выполняется по технологии, подробно рассмотренной в главе 5. В строке бланка запроса **Условие отбора** определим параметры запроса |Номер группы| и |Наименование предмета| для аналогового ввода их значений при выполнении запроса. Это позволяет получить данные для конкретной ведомости.

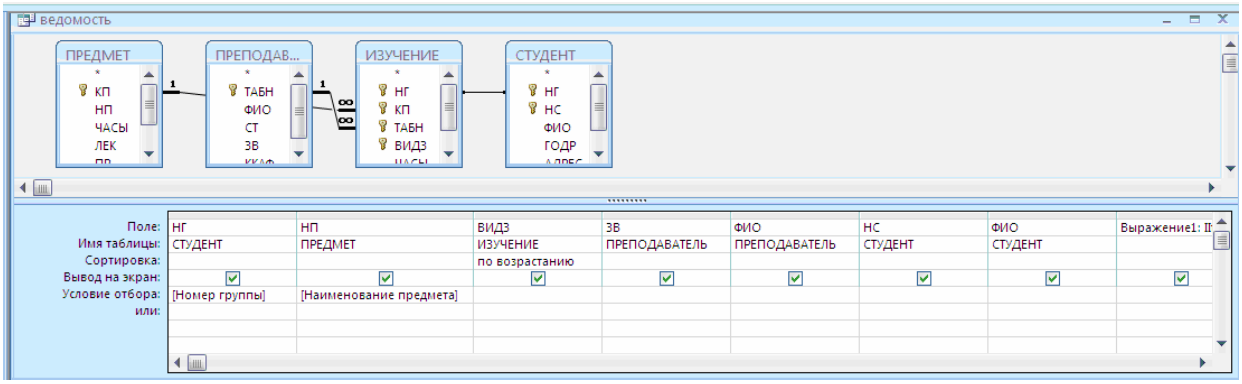


Рис. 11.2. Запрос на выборку в режиме конструктора для формирования отчета-бланка экзаменационной ведомости

В отчете необходимо выводить значения реквизита **Вид сдачи: Экзамен, Зачет,** которых нет непосредственно в таблицах БД, но они могут быть получены на основе значений поля ВИДЗ таблицы ИЗУЧЕНИЕ. Два значения реквизита **Вид сдачи: ”**Экзамен” и ”Зачет” соответствуют двум возможным значениям поля ВИДЗ: ”лек” и ”пр”. Фактически нужно вместо значения ”лек” формировать слово ”Экзамен”, а вместо ”пр” – слово “Зачет”. Для этого надо в запрос ввести новое поле, которое формируется как вычисляемое. Это поле должно содержать встроенную функцию управления.

Выражение 1 : **Iif** ([ВИДЗ] = «лек»; «Экзамен»; «Зачет»)

Именно этого вычисляемого поля по умолчанию является ”Выражение 1:”
Функция Iif (”immediate if” – мгновенное условие) аналогична инструкции if… Then… Else и имеет следующий формат:
Iif (условие; если Истина; если Ложь)
В соответствии с форматом этой функции, если выполнится условие (ВИД)= «лек» (т.е. в поле ВИДЗ находится значение «лек»), ТО РЕЗУЛЬТАТОМ ФУНКЦИИ БУДЕТ «Экзамен». В противном случае, т.е. если (ВИДЗ) = «пр» результатом функции будет слово «Зачет».
В предыдущем рисунке приведен в окончательном виде запрос для подготовки данных отчета-бланка экзаменационной ведомости. Этот запрос сохранен под именем «Ведомость».

Формирование записей результатов в запросе.

В процессе формирования записей результата из таблицы СТУДЕНТ последовательно выбираются записи с *заданным номером группы* (НГ).
С каждой записью таблицы СТУДЕНТ объединяются связанные по номеру группы НГ записи из таблицы ИЗУЧЕНИЕ. Из таблицы ИЗУЧЕНИЕ в этом объединении участвуют только записи с *заданным наименованием предмета.*
Дополнеие формируемых запросом записей полями из таблицы ПРЕДМЕТ и ПРЕПОДАВАТЕЛЬ не добавляет новых записей. Эти таблицы представляют сторону *один* в отношении *один-ко-многим* и служат лишь для расшифровки кодов КП и ТАБН из таблицы ИЗУЧЕНИЕ.
Если по заданному предмету проводится не одно, а два вида занятий: лекции и практика, то число формируемых записей удваивается. В этом случае должны быть сформированы две ведомостит – одна для сдачи экзамена, другая для сдачи зачета.
Заметим, что таблица результатов запроса содержит ненормализованные данные, что проявляется в повторяемости значений в столбцах ГРУППА, НАИМЕНОВАНИЕ ПРЕДМЕТА, ФИО ПРЕПОД., УЧ. ЗВАНИЕ, ВИД ЗАНЯТИЙ и ВЫРАЖЕНИЕ1. Поэтому данная таблица не соответствует проекту макета

отчета Экзамена ведомость не может непосредственно использоваться как выходной документ.
Использование средств Access для подготовки отчета обеспечивает преобразование получаемой таблицы результатов запроса в выходной документ нужной формы в соответствии с проектом макета отчета.

ведомость

Группа	Название пј	вид занятий	уч.звание	ФИО препо	Номер студ	ФИО	Выражение1
101	информатика	лек	ассистент	Новиков П.Н.	04	Макова Н.В.	Экзамен
101	информатика	лек	ассистент	Новиков П.Н.	03	Борисова Е.И.	Экзамен
101	информатика	лек	ассистент	Новиков П.Н.	02	Бондаренко С	Экзамен
101	информатика	лек	ассистент	Новиков П.Н.	01	Аристов Р.П.	Экзамен
101	информатика	пр	доцент	Апухтин И.С.	04	Макова Н.В.	Зачет
101	информатика	пр	доцент	Апухтин И.С.	03	Борисова Е.И.	Зачет
101	информатика	пр	доцент	Апухтин И.С.	02	Бондаренко С	Зачет
101	информатика	пр	доцент	Апухтин И.С.	01	Аристов Р.П.	Зачет

Рис. 11.3. Результат выполнения запрса с данными для экзаменационной ведомости

Конструирование отчета на основе запроса

Рассмотрим технологию отчета на основе запроса **Ведомость** подготовленного выше. Для конструирования отчета во вкладке **Создание** в группе **Отчеты** нажмем кнопку **Конструктор Отчетов** . В окне **Новый отчет** выберем запрос **Ведомость**, который будет источником данных для отчета.

Размещение данных в разделах отчета

Результаты запроса содержат много полей с повторяющимися значениями НГ, НП, ФИО преподавателя, ЗВ, ВИДЗ, Выражение 1: Данные в отчете должны быть размещены, как показано на макете отчета **Экзаменационная ведомость**. Значение каждого из этих полей должно быть представлено в отчете в заголовке один раз.
Заметим, что результаты для отчета получены по заданной группе и предмету. Поскольку по предмету может быть два вида занятий и для каждого вида занятий предполагается вывод отдельной ведомости, необходимо выполнить группировку по полю ВИДЗ. Для группировки по виду занятия нажмем кнопку **Сортировка и группировка** на панели инструментов конструктора отчетов и заполним поля открывшегося окна.

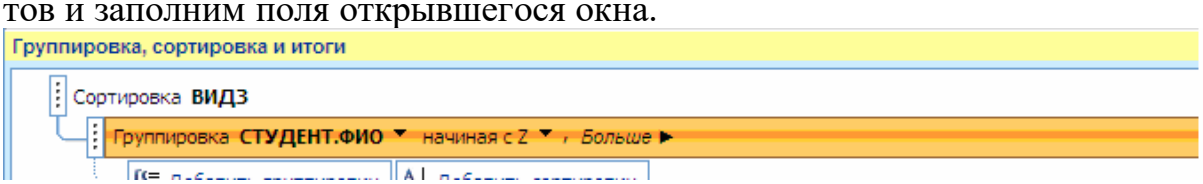


Рис. 11.4. Окно определения групп записей по полю ВИДЗ и сортировки по полю НС.

После определения группировки в окне конструктора отчета появятся разделы **Заголовок группы «ВИДЗ»** и **Примечание группы «ВИДЗ»** .
Чтобы каждая группировка могла быть оформлена как самостоятельная экзаменационная ведомость, будем формировать шапку макета ведомости в заголовке группы **ВИДЗ**, а строки о подведении итогов сдачи экзамена (или зачета) в примечании группы **ВИДЗ**.

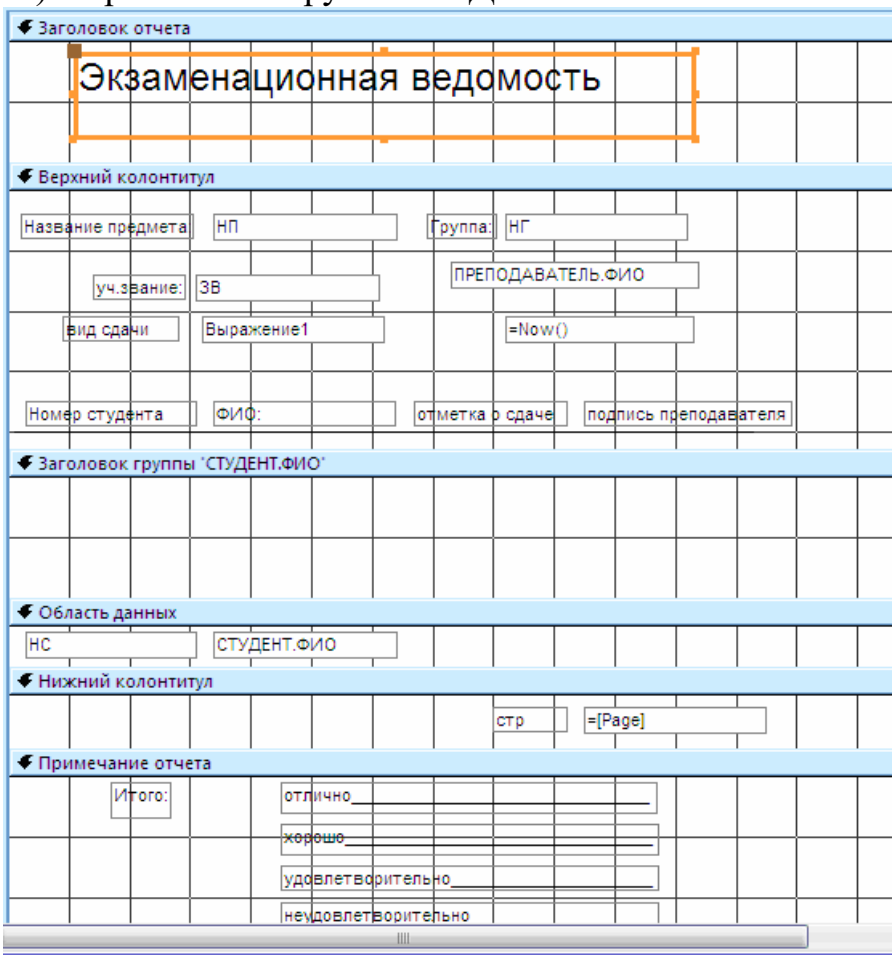


Рис. 11.5.Офрмление шапки и табличной части отчета.

Создание текстовых элементов отчета.

Для создания в заголовке и примечании группы ВИДЗ текстовых элементов за исключением названий реквизитов выводимых полей, используем кнопку **Надпись**. Для оформления текста можно воспользоваться кнопками панели форматирования. Кроме того, можно задавать параметры надписей в окне свойств этих элементов. Окно свойств элемента открывается кнопками панели инструментов **Свойства**. Элемент предварительно должен быть выделен.

При размещении элементов может понадобиться расширить раздел отчета. Для этого установить курсор мыши на его нижнюю границу так, чтобы появилась двунаправленная стрелка, и переместить эту стрелку вниз.

Оформление табличной части отчета.

Для размещения в отчете полей вызовем окно списка полей запроса **Ведомость**, нажав на панели конструктора отчетов кнопку **Добавить поля**

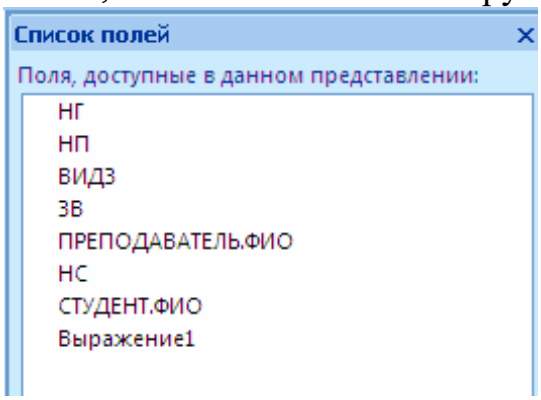


Рис. 11.6. Список полей запроса **Ведомость**

В область данных разместим поля НС и ФИО, на основе содержимого которых будут формироваться строки табличной части отчета в соответствии с проектом макета отчета. Для этого из списка полей перетащим поля НС (номер студента) и СТУДЕНТ. ФИО в область данных.

Каждое поле представляется двумя элементами: собственно полем и его подписью. Например, поле ФИО таблицы СТУДЕНТ представляется элементом СТУДЕНТ. ФИО, отображающим значение поля, и элементом, содержащим подпись этого поля из таблицы базы данных – Фамилия И.О. Установим нужный шрифт в элементах. Используем команду меню **Формат |Размер| по размеру данных** для установки размеров элемента по размеру текста подписи.

Замети, что подпись поля из табличной части должна быть перемещена для формирования заголовка столбцов в раздел **Заголовок группы «ВИДЗ»**. Для этого выделим подпись поля. Затем вырежем его. Активизируем раздел **Заголовок группы «ВИДЗ»** и вставим подпись в этот раздел.

Оформим элементы в соответствии с требованиями проекта макета отчета. В частности, заменим подпись поля «Номер студента на N п/п». Для того чтобы в бланке экзаменационной ведомости были пустые столбцы для оценок и подписи (заполняемые преподавателем вручную на экзамене), добавим в строку с заголовками столбцов новые текстовые элементы «Отметка о сдаче» и «Подпись преподавателя».

Размещение полей в заголовке

Значения полей НГ, НП, ФИО преподавателя, ЗВ, Выражение 1 (экзамен, зачет) повторяются во всех записях группировки. Поэтому они должны размещаться в заголовке группы. Перетащим эти поля из списка полей в раздел **Заголовок группы «ВИДЗ»**.

Добавление текущей даты и номера страницы

Для добавления в отчет текущей даты воспользуемся встроенной функцией Now (). Для этого создадим несвязанный элемент, нажав кнопку **Поле** на панели элементов и разместим его в нужном месте. Зададим в окне его свойств на вкладке **Данные** в строке **Данные** выражение = Now (), а на вкладке **Макет** в строке **Формат поля** выберем значение – **Средний формат даты.**

Для добавления номера страницы в раздел **Нижний колонтитул** создадим несвязанный элемент и заполним в его свойствах строку **Данные** выражением = «Страница» @ [Page].

Создадим горизонтальные и вертикальные линии в соответствии с макетом, воспользовавшись кнопкой **Линии**. Установим нужную толщину линий в свойствах этого элемента.

Сохраним отчет под именем «Экзаменационная ведомость».

Предварительный просмотр отчета.

При подготовке к просмотру отчета, построенного на базе запроса с параметрами, Acceess предварительно выполняет запрос и выводит диалоговые окна ввода параметров отчета.

Просмотр отчета. Для отображения отчета **Экзаменационная ведомость** на экране в том виде, в котором он будет напечатан, выйдем из режима конструктора отчетов, нажав кнопку **Вид**. Последовательно появляются диалоговые окна для ввода параметров запроса, которые являются в то же время параметрами отчета.

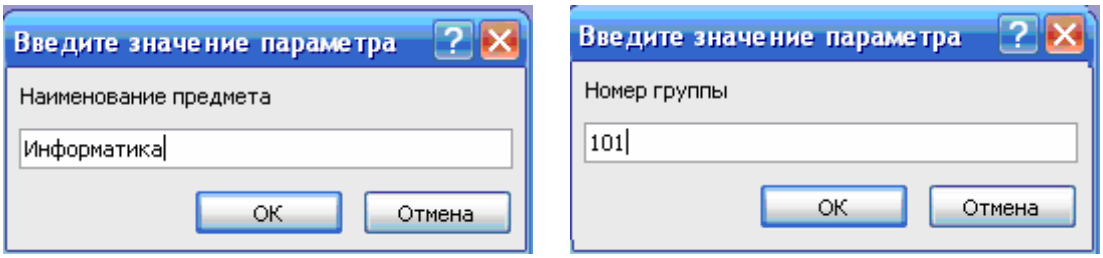
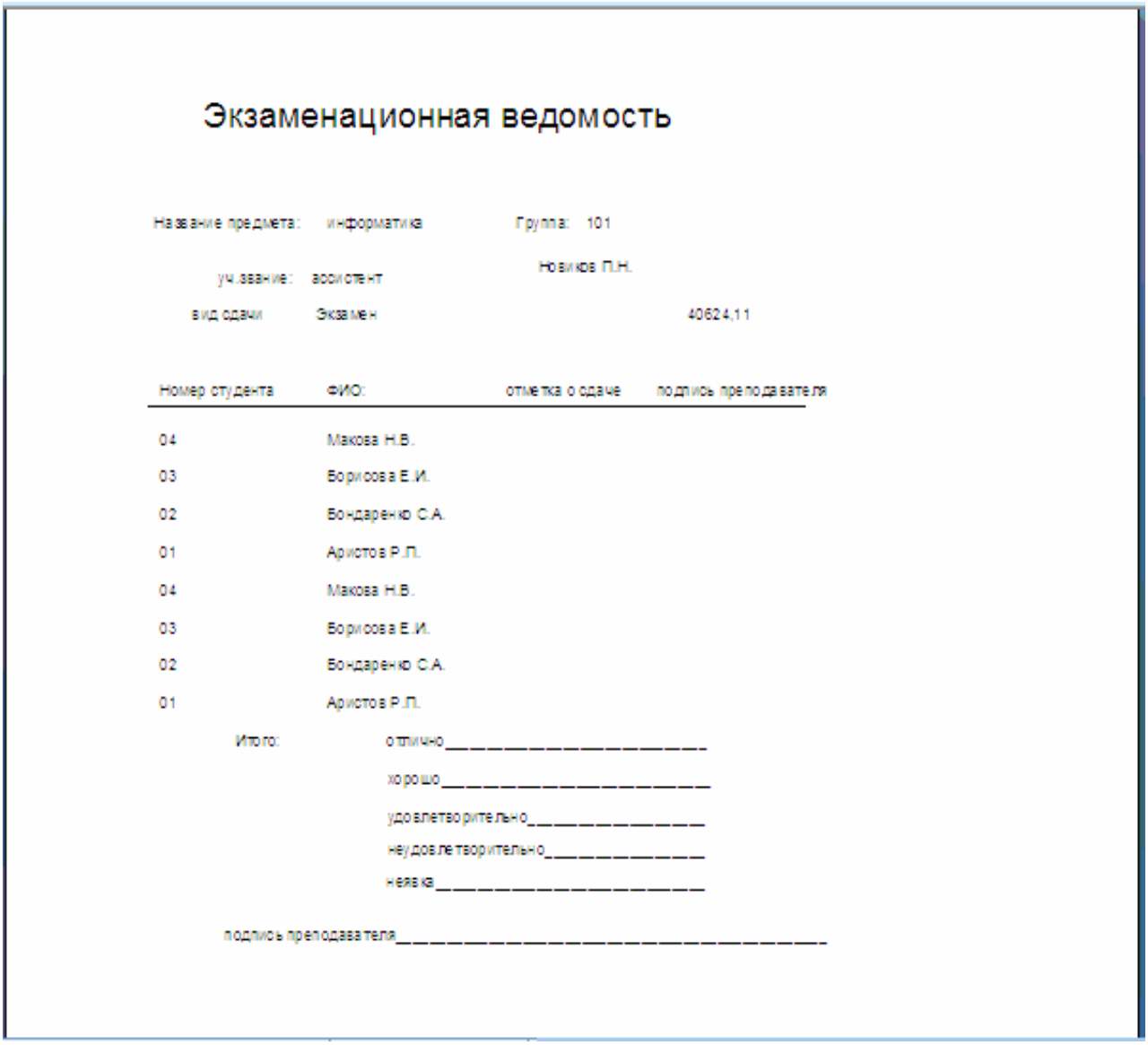


Рис. 11.7. Диалоговые окна ввода параметров отчета

Введем значение параметра «Номер группы» – «101» и значение параметра «Наименование предмета» – «Информатика». Для данных значений параметров отчет будет состоять из двух страниц, на первой из которых будет представлена экзаменационная ведомость для экзамена по информатике, на второй – для сдачи зачета. Первая страница отчета **Экзаменационная ведомость** приведена ниже.



Экзаменационная ведомость

Название предмета: информатика Группа: 101

Новиков П.Н.

уч.звание: ассистент

вид сдачи Экзамен 40624,11

Номер студента	ФИО:	отметка о сдаче	подпись преподавателя
04	Макова Н.В.		
03	Борисова Е.И.		
02	Бондаренко С.А.		
01	Аристов Р.П.		
04	Макова Н.В.		
03	Борисова Е.И.		
02	Бондаренко С.А.		
01	Аристов Р.П.		

Итого: отлично_____

хорошо_____

удовлетворительно_____

неудовлетворительно_____

неявка_____

подпись преподавателя_____

рис. 11.8. Вывод бланка экзаменационной ведомости для лекций на первой странице отсчета

Урок 12. Управление приложением пользователя

В предыдущих разделах была рассмотрена технология разработки объектов базы данных Access: таблиц, форм, запросов, отчетов, макросов, модулей как средств решения задач и разработки интерфейса приложения пользователя. При этом большое количество объектов, не сгруппированных по функциям приложения, затрудняет выполнение пользователем задач обработки данных в автоматизируемой предметной области.

Для организации эффективной работы пользователя нужно создать целостное приложение данной предметной области, все компоненты которого должны быть сгруппированы по функциональному назначению. При этом необходимо обеспечить удобный *графический интерфейс пользователя.*

Особую роль при создании приложения играют формы, так как они являются основным диалоговым средством работы пользователя.

Формы построены таким образом, что любое действие пользователя вызывает реакцию системы, т.е. воспринимается как событие, в зависимости от которого могут выполняться необходимые действия. Для выполнения этих действий используются макросы или процедуры обработки события, созданные пользователем на языке VBA, которые связаны с событием. VBA позволяет вызвать внешние процедуры из библиотечных баз данных MS Access (.mda) и библиотек динамической компоновки (.dll). Именно при работе в форме возникает основной ряд событий, обработка которых позволяет связывать различные объекты в задачах обработки данных. Таким образом, ходом управления приложения можно управлять, обрабатывая возникающие в формах события. Приложения, создаваемые средствами Access, используют расширенную обработку событий, при которой программы приложения выполняются в ответ на различные события.

Для объединения объектов в едином диалоговом приложении могут быть созданы так называемые *кнопочные формы*. Кнопочная форма является панелью управления приложением. Кнопки такой формы обеспечивают вызов других кнопочных форм, а также отдельных объектов: Отчетов, форм, макросов, модулей, с которых начинается решение задачи. Сюда же помещаются и кнопки для возврата к кнопочным формам предыдущих уровней выхода из Access. Обычно также предусматривается кнопка для изменений самой кнопочной формы.

Пользователь, нажимая кнопку на панели управления, инициирует событие *нажатия кнопки*. К этому событию могут привязываться действия по открытию других кнопочных форм или конкретные действия по обработке данных, реализующие функции приложения.

Вызов главной кнопочной формы – панели управления приложением при открытии базы данных позволяет пользователю сразу начать работу в среде приложения и приступить к выполнению задач.

В Access имеется средство автоматизированной разработки формы по управлению приложением, называемое *диспетчером кнопочных форм.*

Создание кнопочной формы с помощью диспетчера

Пользователь имеет возможность с помощью диспетчера кнопочных форм создавать кнопочную форму стандартного вида. Создадим главную кнопочную форму, которая будет использоваться как панель управления приложения «Учебный процесс».

Кнопка **Диспетчер кнопочных форм** может быть получена следующими действиями:

На вкладке **Файл** выбираем **Параметры**. Откроется окно **Параметры Access** (см. рис. 12.1). Далее выбираем **Настройка ленты**. В списке справа выбираем **Работа с базами данных** и нажимаем на кнопку **Создать группу**. В левом списке выбираем **Диспетчер кнопочных форм** и нажимаем кнопку **Добавить.**

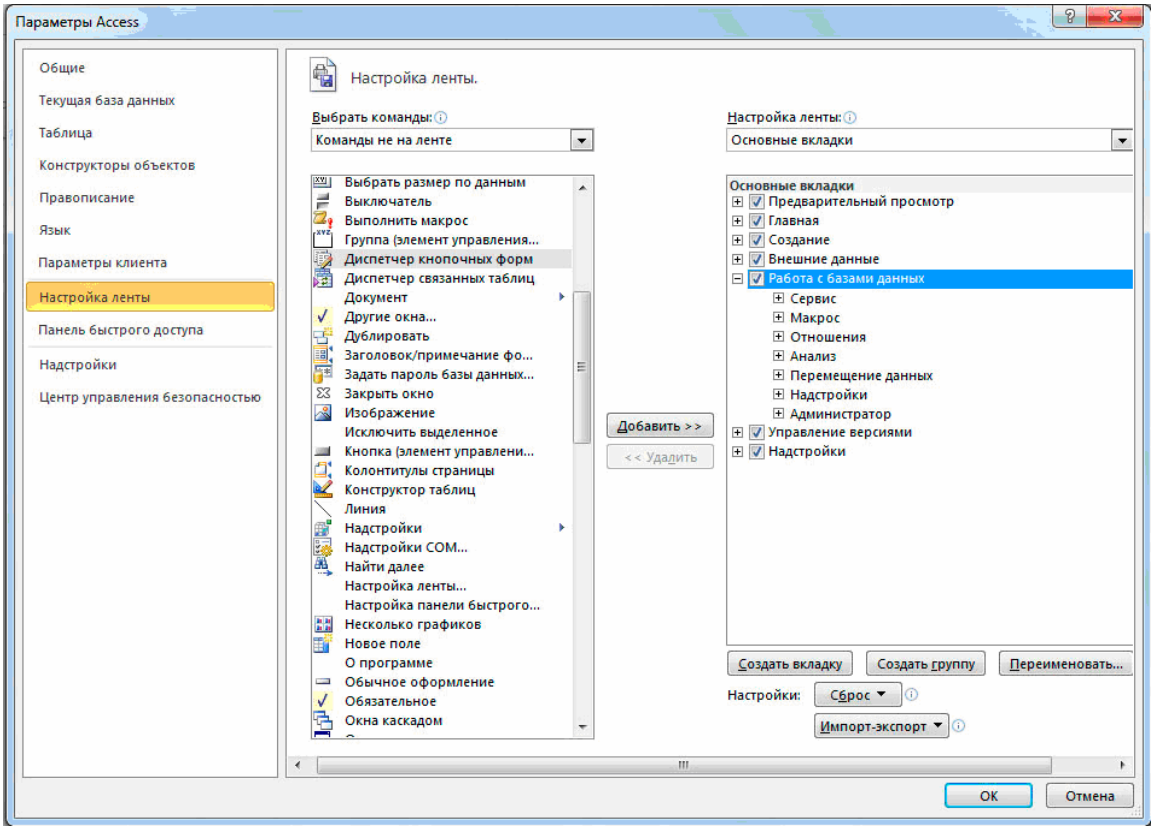


Рис. 12.1. Работа с параметрами.

В результате окно диалога должно измениться так, как показано на рис. 12.2.

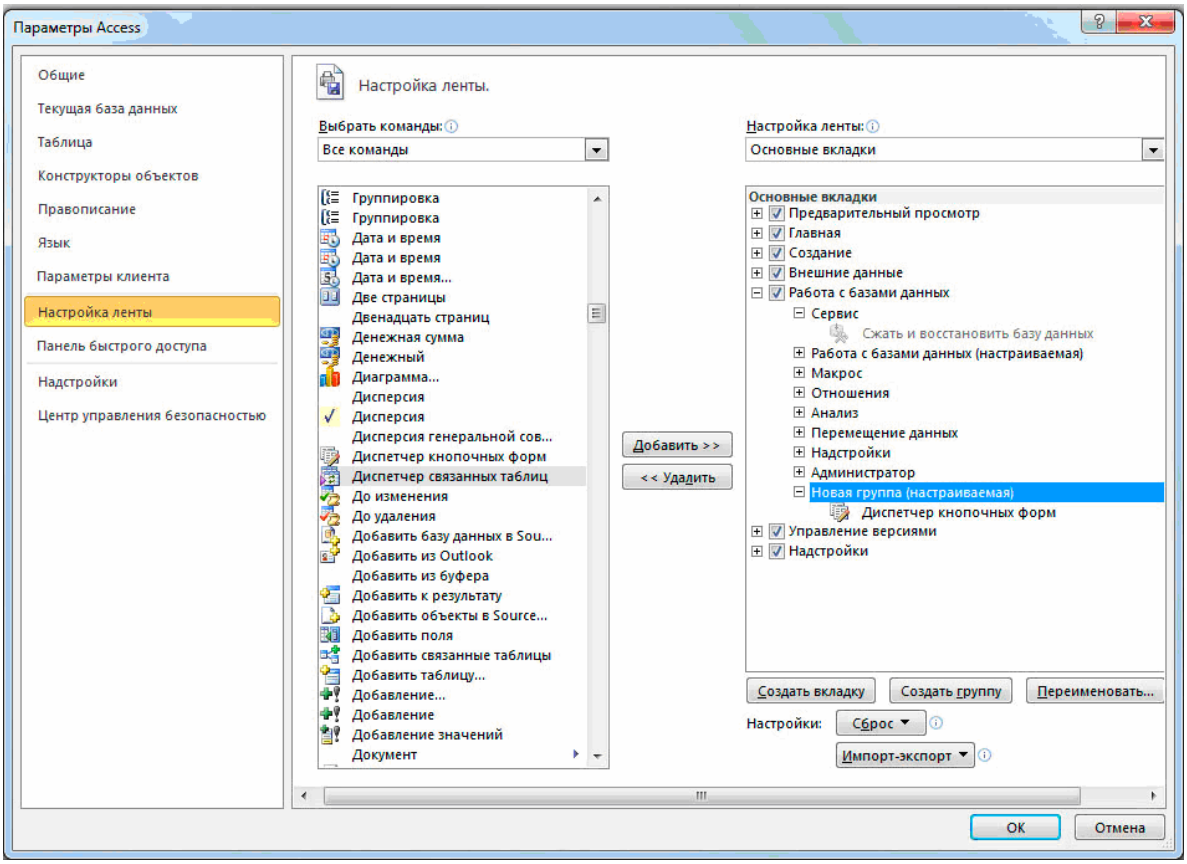


Рис. 12.2. Работа с параметрами (продолжение).

Теперь на вкладке **Работа с базами данных** в группе **Работа с базами данных** нажмите кнопку **Диспетчер кнопочных форм** (см. рис. 12.3).

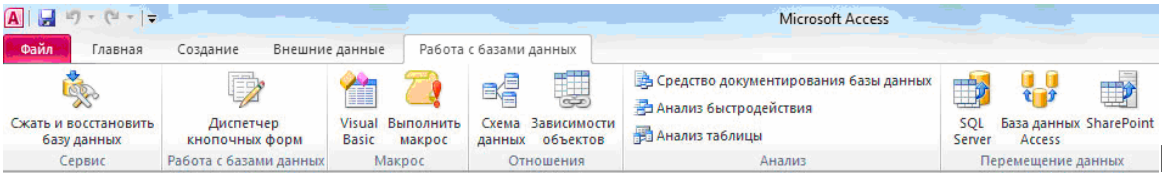


Рис. 12.3. Вкладка **Работа с базами данных**, группа **Работа с базами данных**.

В окне **Диспетчер кнопочных форм** подтвердим создание кнопочной формы (рис.12.4).

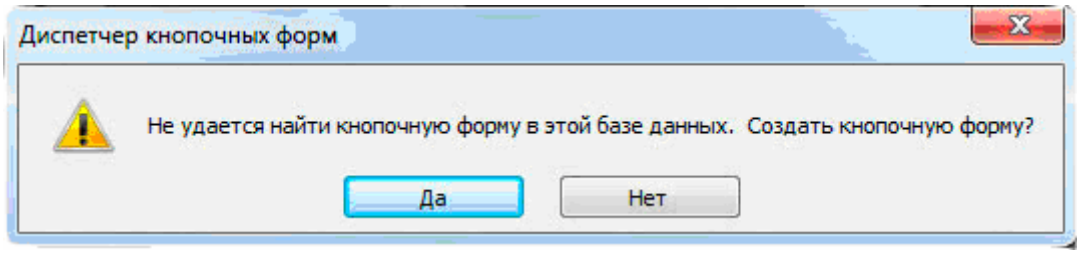


Рис. 12.4. Окно подтверждения создания кнопочной формы приложения

Появится окно со строкой **Главная кнопочная форма** (рис.12.5), в котором формируется список кнопочных форм разных уровней.

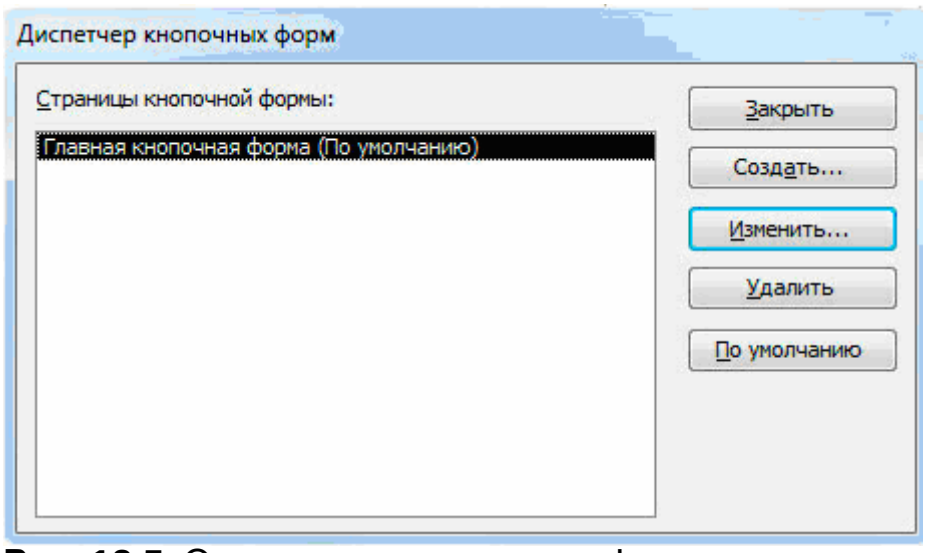


Рис. 12.5. Окно списка кнопочных форм приложения

Создание подчиненных кнопочных форм приложения

Создадим три кнопочные формы для комплексов задач деканата, кафедр и учебного отдела.

Для создания кнопочной формы в окне диспетчера кнопочных форм нажмем кнопку **Создать**. В окне **Создание** (рис.12.6) в поле **Имя страницы кнопочной формы** введем имя первой кнопочной формы «Кнопочная форма Деканат».

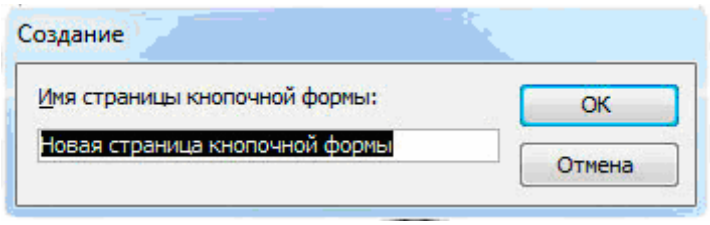


Рис.12.6. Окно создания пустой кнопочной формы

После нажатия кнопки **ОК** в окне диспетчера кнопочных форм появится строка «Кнопочная форма Деканат».

Аналогичными действиями создадим еще две кнопочные формы: **Кнопочная форма Кафедра** и Кнопочная форма **Учебный отдел** (рис.12.7).

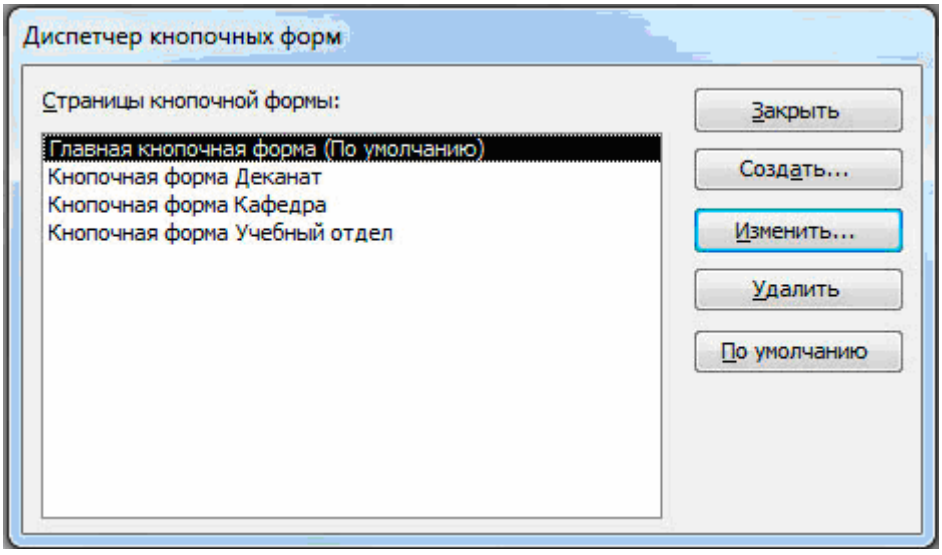


Рис. 12.7. Список кнопочных форм приложения «Учебный процесс»

Любую из подготовленных кнопочных форм можно сделать стартовой, которая будет открываться по умолчанию при открытии базы данных. Для этого следует выбрать ее имя в окне диспетчера кнопочных форм и нажать кнопку **По умолчанию**. Оставим главную кнопочную форму стартовой кнопочной формой. Эта кнопочная форма будет находиться на верхнем уровне иерархии взаимосвязанных кнопочных форм в соответствии со структурой создаваемого приложения.

Редактирование имени и формирование элементов кнопочной формы

Элементами кнопочных форм являются кнопки с подписями. Подпись задается пользователем и должна по возможности кратко и точно называть действия, которые будут выполняться при нажатии кнопки. Для формирования элементов главной кнопочной формы в окне **Диспетчер кнопочных форм** выделим соответствующую ей строку и нажмем кнопку **Изменить**.

В окне диалога **Изменение страницы кнопочной формы** (рис.12.8) главную кнопочную форму переименуем в «Приложение «Учебный процесс»», введя это имя в поле **Название кнопочной формы**.

Элементы, создаваемые в кнопочной форме, могут быть подразделены на два основных типа:

□ Элементы, предназначенные для организации вызова других кнопочных форм
□ Элементы, обеспечивающие открытие формы, отчета, запуск макроса программы, выход из приложения с закрытием базы данных или перехода в режим продолжения разработки кнопочной формы диспетчером кнопочных форм.

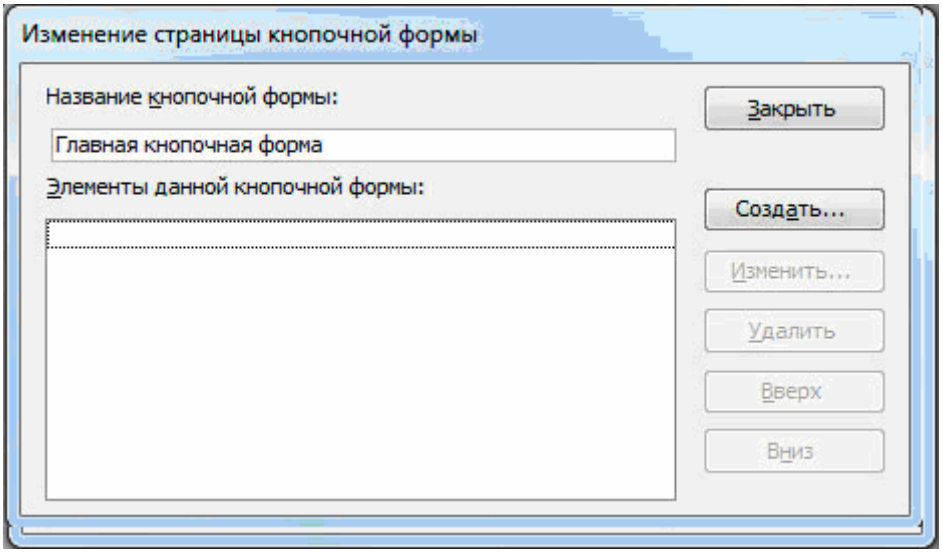


Рис. 12.8. Окно редактирования имени кнопочной формы и создания ее кнопок

Формирование элементов для вызова кнопочных форм. Для формирования в главной кнопочной форме **Приложение Учебный процесс** кнопки вызова подчиненной кнопочной формы нажмем кнопку **Создать**.

В окне **Изменение элемента кнопочной формы** строке **Команда** выберем **Переход к кнопочной форме** (рис.12.9)

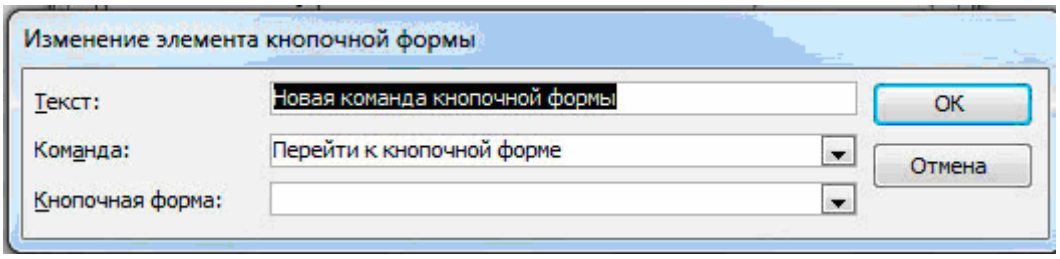


Рис. 12.9. Окно создания элемента кнопочной формы

В строке **Кнопочная форма** выбирается форма, переход к которой должна обеспечивать создаваемая кнопка текущей формы. Откроем список созданных для приложения кнопочных форм и выберем элемент **Кнопочная форма Деканат.**

В строке **Текст** введем подпись для этой кнопки: «Задачи деканата» (рис.12.10).

Для завершения формирования элемента нажмем **ОК**. Элемент отобразиться в списке **Элементы данной кнопочной формы**.

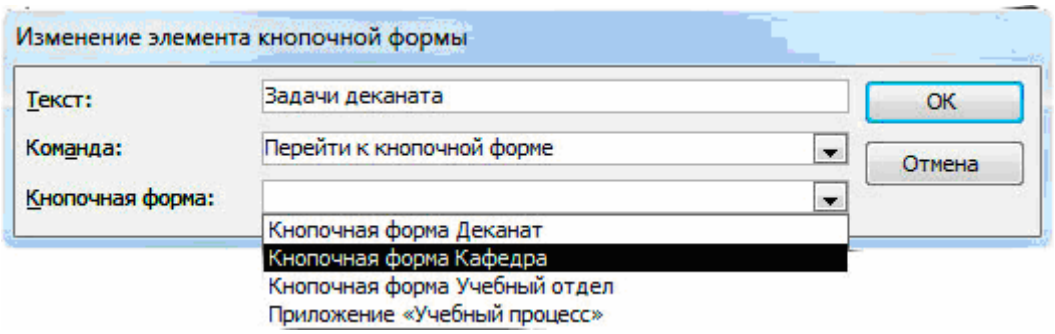


Рис. 12.10. Выбор подчиненной кнопочной формы

Аналогичным образом создадим кнопки вызова других подчиненных форм и зададим для них подписи «Задачи кафедры» и «Задачи учебного отдела».

Чтобы изменить или удалить какую – либо из созданных кнопок, надо выбрать имя этой кнопки в списке **Элементы данной кнопочной формы** и нажать кнопку **Изменить** (Edit) или **Удалить**. При необходимости изменить порядок кнопок в списке надо выбрать элемент и нажать кнопку Вверх или Вниз.

Для перехода к редактированию созданных кнопочных форм создадим в главной кнопочной форме кнопку **Изменение кнопочной формы,** выбрав для нее команду **Конструктор приложения**. В дальнейшем это позволит в любой момент вызвать диспетчер кнопочных форм и с его помощью внести необходимые изменения.

Для завершения работы с приложением создадим кнопку, выбрав для него команду **Выйти из приложения**, и присвоим ей это же имя.

Страница главной кнопочной формы **Приложение Учебный процесс** после создания всех ее элементов представлена на рис. 12.11.

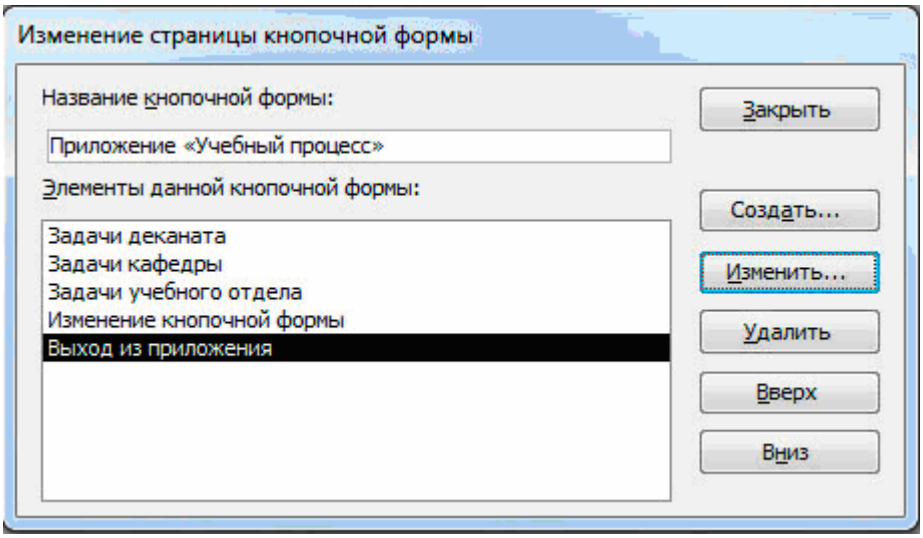


Рис. 12.11. Страница кнопочной формы Приложение Учебный процесс

Закончив создание страницы кнопочной формы, нажмем кнопку **Закрыть**. Это позволит вернуться к списку всех кнопочных форм приложения.

В подчиненных кнопочных формах создадим кнопку для возврата к главной кнопочной форме, заполнив поля в окне **Изменение элемента кнопочной формы**, как показано на рис.12.12.

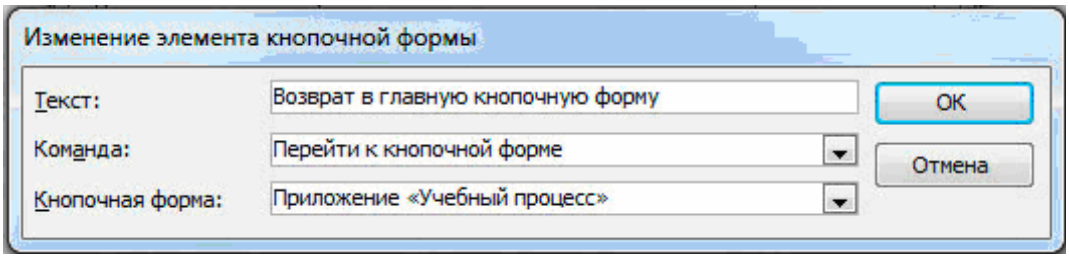


Рис.12.12. Создание кнопки для возврата в главную форму

Таким образом, могут быть установлены все необходимые связи между кнопочными формами, представленными в списке окна диспетчера кнопочных форм. Практически на эти связи и число уровней не накладывается ограничения и могут быть созданы любые удобные пользователю переходы по кнопочным формам.

Формирование элементов для открытия объектов приложения. Для формирования кнопок, обеспечивающих выполнение задач пользователя, необходимо, чтобы они были связаны с одним из объектов приложения. Диспетчер кнопочных форм обеспечивает связь с формами, отчетами, макросами и модулями. Для связи создаваемой кнопки с нужным объектом надо выбрать в строке **Команда** (рис.12.13) одну из команд:

- Открыть форму для добавления
- Открыть форму для изменения
- Открыть отчет

- Выполнить макрос
- Выполнить программу

В третьей строке окна **Изменение элемента кнопочной формы**, которая будет соответствовать команде, выбранной во второй строке, выберем конкретный объект.

Например, если выбрать в строке **Команда** команду – **Открыть отчет**, появится третья строка – **Отчет**, где можно будет выбрать имя открываемого отчета, Если выбрать в строке Команда– команду **Открыть форму для изменения**, появится строка **Форма**.

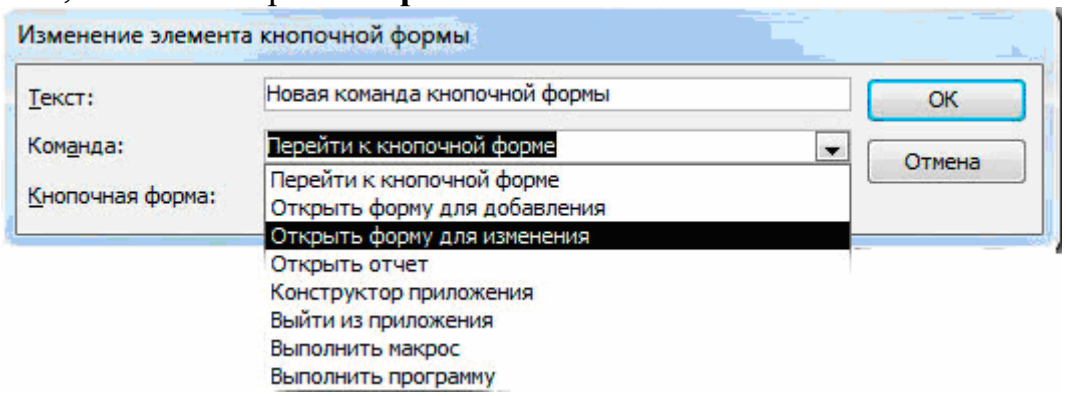


Рис. 12.13. Выбор команды для элемента кнопочной формы

Создадим в кнопочной форме Деканат кнопку для работы с формой УСПЕВАЕМОСТЬ, через которую производится ввод данных из экзаменационной ведомости и с которой связано выполнение макроса, решающего задачу выдачи списков двоечников и отличников (см. выше).

В окне диспетчера кнопочных форм выберем **Кнопочная форма Деканат**, Для создания элементов в этой форме нажмем кнопку **Изменить**. В открывшемся окне **Изменение страницы кнопочной формы** нажмем кнопку **Создать**.

В окне **Изменение элемента кнопочной формы** выберем команду **Открыть форму для изменения** (рис.12.13). В появившейся строке Форма откроем список доступных форм и выберем форму УСПЕВАЕМОСТЬ. В строку **Текст** введем подписи кнопки «Списки успеваемости».

Аналогичным образом создадим кнопку для выполнения макроса «Расчет числа студентов в группах» (см. выше). Этот макрос решает задачу путем запуска на выполнение двух запросов. Введем для этой кнопки подпись «Расчет числа студентов в группах».

Аналогичным образом могут быть созданы все необходимые в кнопочных формах кнопки для вызова форм, отчетов, макросов и программ пользователя.

Главная кнопочная форма **Приложение Учебный процесс** представлена на рис. 12.14.

Фрагмент структуры приложения «Учебный процесс» иллюстрирует рис. 12.15.

После завершения работы с диспетчером кнопочных форм в списке форм базы данных появляется форма с именем **Кнопочная форма**.

Главная кнопочная форма может запускаться при открытии базы данных.

Для этого нажмите кнопку **Файл**, а затем выберите команду **Параметры.** В диалоговом окне **Параметры Access** щелкните элемент **Текущая база данных.**

Замечание.
Эта команда доступна при открытой базе данных. Параметры запуска определяют вид окна приложения при его открытии. Эти параметры позволяют изменить заголовок окна приложения, выбрать собственное меню, контекстное меню, используемое по умолчанию в формах и отчетах, панели инструментов, а также указать форму, открываемую в базе данных по умолчанию.

При помощи параметров запуска можно запретить пользователю работу в окне базы данных, и тогда он сможет выполнять только те работы, которые предусмотрены в кнопочных формах приложения.

Диспетчер кнопочных форм Access создает таблицу **Элементы кнопочной формы**, содержащую описания кнопок формы и выполняемых ими действий.

Замечание.
Попытка изменить кнопочную форму в режиме конструктора формы может привести к тому, что приложение перестанет работать. Однако вставка в кнопочную форму рисунков не приводит к таким последствиям.

Рис. 12.14. Главная кнопочная форма **Приложение Учебный процесс**

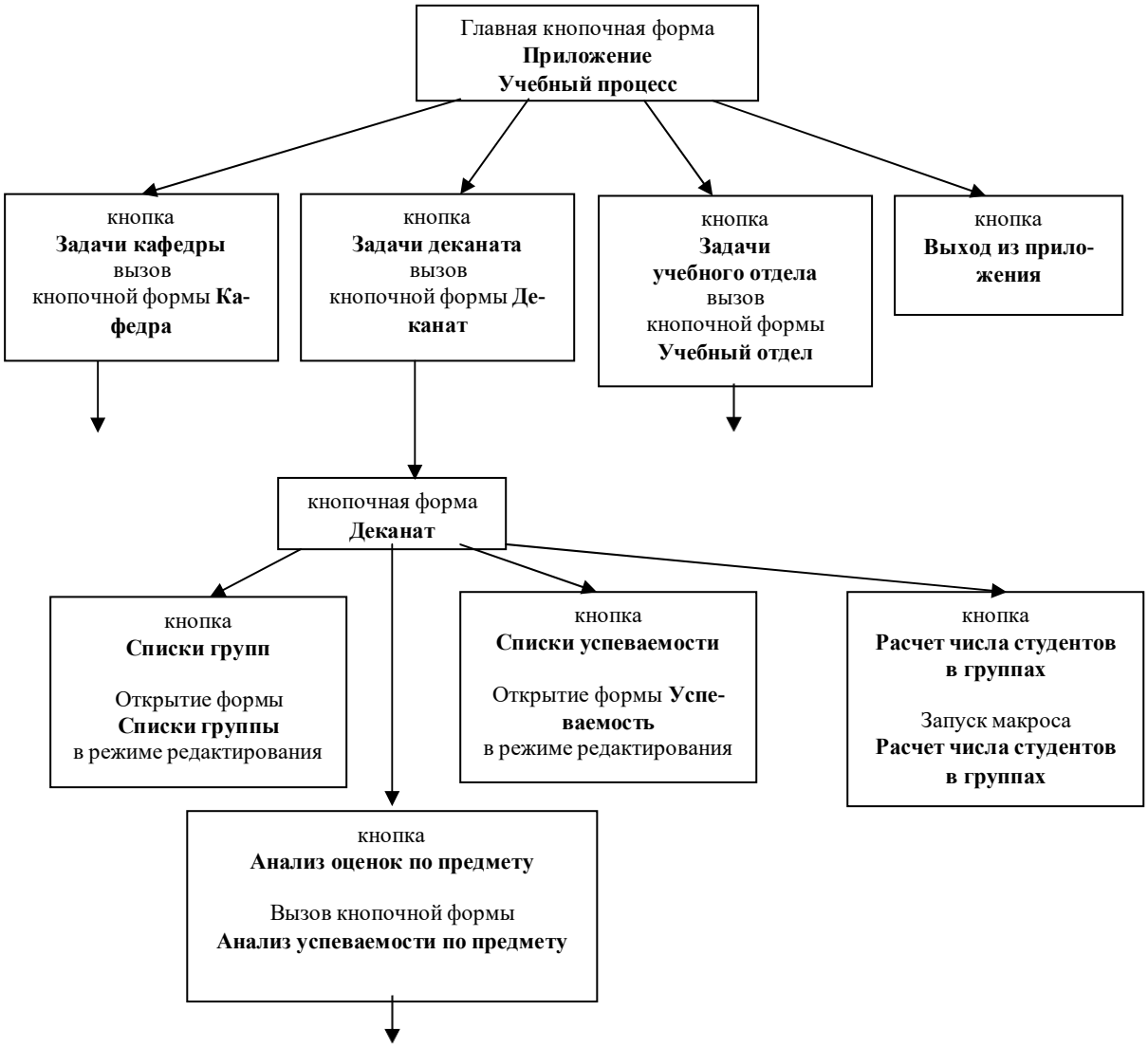


Рис. 12.15. Фрагмент структуры приложения «Учебный процесс»

Добавление рисунка в кнопочную форму

В стандартную кнопочную форму можно вставить рисунок, например, созданную в Microsoft Paint эмблему фирмы. В левой части этой формы уже размещен элемент управления **Рисунок**, в котором собственно рисунок отсутствует, но создана рамка для его вывода на экран. Заметим, что такой элемент может быть создан в любой форме с помощью кнопки панели элементов **Рисунок**.

Воспользовавшись кнопкой , переключим кнопочную форму режим конструктора. Щелкнем кнопкой мыши на области формы слева и нажмем правую кнопку мыши. В открывшемся окне свойств на вкладке **Макет** в строке **Рисунок** укажем путь и имя файла с расширением одного из графических

приложений, форматы которых поддерживаются установленными в системе фильтрами.

Если путь или имя файла неизвестны, можно выбрать рисунок с помощью построителя рисунков, который вызывается нажатием кнопки построителя в конце строки **Рисунок**. В открывшемся окне **Выбор рисунка** откроем папку, в которой хранится файл с нужным рисунком, и выберем этот файл. Предоставляемая построителем возможность просматривать рисунки значительно облегчает процесс поиска. По умолчанию тип рисунка определен как – **Внедренный**. Если изменить это свойство на **Связанный**, то рисунок в форме будет соответствовать содержимому связанного файла.

Конструирование кнопочной формы

Главная и подчиненные ей кнопочные формы могут быть созданы пользователем самостоятельно в режиме конструктора. В этом режиме может быть создана панель управления приложением нестандартного вида.

Для создания кнопочной формы необходимо в окне базы данных на вкладе **Создание** нажать кнопку **Конструктор форм**. При этом не должен указываться источник данных. Открывается форма в режиме конструктора. В этой форме можно создать кнопки для вызова кнопочных форм или выполнения любых объектов приложения. Кнопочная форма может быть сохранена под любым именем и в любой момент отредактирована в режиме конструктора. Таким образам могут быть созданы все необходимые кнопочные формы приложения. Каждая созданная кнопочная форма будет являться отдельным объектом и, в отличие от форм, созданных диспетчером кнопочных форм, будет представлена в списке форм в окне базы данных.

Пользователь имеет возможность создать кнопку самостоятельно или с помощью мастера. Мастера значительно ускоряет процесс построения кнопки и связывания ее с нужными действиями, автоматически выполняя всю необходимую работу. Мастер ведет диалог с пользователем и на основе его ответов создает кнопку. Технология создания кнопок описана в главе 4.

Мастер позволяет создавать кнопки более 30 типов. Например, кнопки для открытия формы, страницы, выполнения запроса, макроса, выхода из приложения или выполнения других действий. Открытие кнопочной формы ничем не отличается от открытия обычной формы.

Мастер кнопок Access создает процедуру обработки событий на языке Visual Basic. Эта процедура связывается с событием «Нажатие кнопки» и выполняет выбранное действие.

Пользователь имеет возможность открыть процедуру, ознакомиться с программой и при необходимости внести в программу изменения. Для открытия

процедуры обработки события надо в режиме конструктора вызвать контекстно-зависимое меню кнопки и выбрать команду **Обработка событий** (Build Event…).

Например, если в форме создана кнопка для открытия формы DVA, то процедура, связанная с кнопкой, будет иметь вид, представленный на рис.12.16.

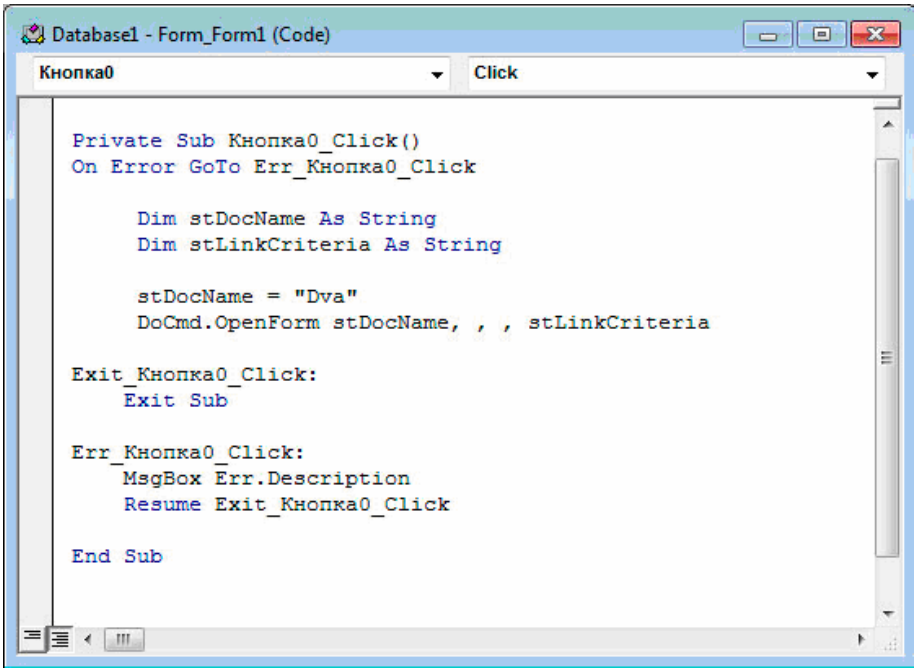


Рис.12.16. Процедура открытия формы DVA, созданная мастером кнопок.

Запуск Access с открытием приложения пользователя

Если в параметрах запуска установлена ссылка на кнопочную форму, то эта форма открывается автоматически при открытии базы данных. Если на рабочем столе Windows создать ярлык для запуска Access с автоматическим открытием базы данных, то пользователю, для того чтобы сразу попасть в среду своего приложения, достаточно будет дважды щелкнуть на этом ярлыке мышью.

Для запуска Access с автоматическим открытием базы данных создать ярлык для Access или ярлык для базы данных.

Создание ярлыка для Access. Нажмем правую кнопку мыши на кнопке **Пуск** панели задач Windows. Выберем команду **Проводник.** Откроем папку, в которой хранится Access. установим курсор на значок программы MSAccess.exe. нажмем правую кнопку мыши и в контекстном меню выберем команду **Создать ярлык**. Ярлык будет создан в той же папке, где находится программа Msaccess.exe.

Зададим в созданном ярлыке свойства, определяющие открытие базы данных. Щелкнув правой кнопкой мыши на выделенной строке ярлыка, откроем контекстное меню и выберем команду **Свойства**.

В окне свойств выберем вкладку **Ярлык**, установим курсор в поле **Объект** и в конце командной строки добавим имя открываемой базы данных, заключив его в кавычки. Напомним, что при этом указывается полный путь к файлу базы данных.

Новый ярлык можно разместить на рабочем столе или добавить в главное меню Windows, просто перетащив ярлык на рабочий стол или в главное меню. Созданный на рабочем столе ярлык для запуска Access представлен на рис. 12.17.

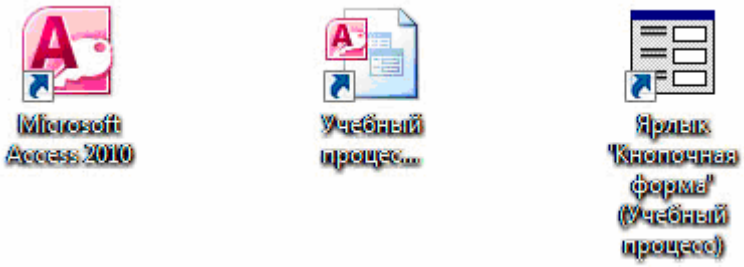


Рис. 12.17. Ярлыки запуска Access: ярлык для запуска Access, ярлык для базы данных, ярлык кнопочной формы

При запуске приложения с помощью ярлыка можно поменять заставку Access на любой точечный рисунок .bmp, если сохранить его в той же папке, где размещен файл базы данных, под тем же именем, что и база данных.

Создание ярлыка для базы данных. Другим, еще более простым способом запуска Access с открытием базы данных, является создание ярлыка для базы данных.

Откроем папку, в которой хранится файл базы данных. Установим курсор на значок файла базы данных, нажмем правую кнопку мыши и в контекстном меню выберем команду **Создать ярлык**. Перетащим созданный ярлык на рабочий стол. Созданный на рабочем столе ярлык для базы данных «Учебный процесс» представлен на рис. 12.17.

Создание ярлыка для объекта базы данных. Возможно создание ярлыка и для отдельного объекта базы данных, в том числе и для кнопочной формы, управляющей приложением. Такой ярлык позволяет запустить Access, открыть в нем базу данных и кнопочную форму.

Для создания ярлыка достаточно перетащить объект из окна базы данных на рабочий стол. Созданный на рабочем столе ярлык для кнопочной формы базы данных «Учебный процесс» представлен на рис. 12.17.

Разработчик: Калинкина Е. А - преподаватель общепрофессиональных дисциплин Областного государственного бюджетного профессионального образовательного учреждения «Рязанский колледж электроники»

Пособие подготовлено с использованием материалов книги Е.М. Карчевский, И.Е. Филиппов, И.А. Филиппова Access 2010 в примерах