

МИНИСТЕРСТВО НАУКИ И ВЫСШЕГО
ОБРАЗОВАНИЯ РОССИЙСКОЙ ФЕДЕРАЦИИ
АМУРСКИЙ ГОСУДАРСТВЕННЫЙ УНИВЕРСИТЕТ

И.Е. Еремин, В.В. Еремина, О.В. Жилиндина

Базы данных

Учебно-методическое пособие

Благовещенск
Издательство АмГУ
2021

*Печатается по решению
ученого совета университета*

Рецензенты:

Н.А. Чалкина, доцент кафедры общей математики и информатики Амурского государственного университета, к. пед. наук, доцент.

Еремин И.Е., Еремина В.В., Жилиндина О.В.

Базы данных. Учебно-методическое пособие. /И.Е. Еремин, В.В. Еремина, О.В. Жилиндина. – Благовещенск: Изд-во АмГУ, 2021. – 108 с.

Учебно-методическое пособие предназначено для студентов направления подготовки 09.03.01. – Информатика и вычислительная техника, 09.03.02. – Информационные системы и технологии, изучающих дисциплину «Базы данных». Пособие содержит теоретический материал, указания для выполнения лабораторных работ и примеры для самостоятельного выполнения.

© Амурский государственный университет, 2021

© Еремин И.Е., Еремина В.В., Жилиндина О.В., 2021

ВВЕДЕНИЕ

Современные информационные системы, основанные на концепции интеграции данных, характеризуются огромными объемами хранимых данных, сложной организацией, необходимостью удовлетворять разнообразные требования многочисленных пользователей.

Цель любой информационной системы — обработка данных об объектах реального мира. В широком смысле слова база данных — это совокупность сведений о конкретных объектах реального мира в какой-либо предметной области. Под *предметной областью* принято понимать часть реального мира, подлежащего изучению для организации управления и, в конечном счете, автоматизации, например, предприятие, ВУЗ и т.д.

Создавая базу данных, пользователь стремится упорядочить информацию по различным признакам и быстро извлекать выборку с произвольным сочетанием признаков. Сделать это возможно, только если данные структурированы.

Структурирование — это введение соглашений о способах представления данных.

Неструктурированными называют данные, записанные, например, в текстовом файле.

Пользователями базы данных могут быть различные прикладные программы, программные комплексы, а также специалисты предметной области, выступающие в роли потребителей или источников данных, называемые *конечными пользователями*.

В современной технологии баз данных предполагается, что создание базы данных, ее поддержка и обеспечение доступа пользователей к ней осуществляются централизованно с помощью специального программного инструментария — *системы управления базами данных*.

База данных (БД) — это поименованная совокупность структурированных данных, относящихся к определенной предметной области.

Система управления базами данных (СУБД) — это комплекс программных и языковых средств, необходимых

для создания баз данных, поддержания их в актуальном состоянии и организации поиска в них необходимой информации.

Централизованный характер управления данными в базе данных предполагает необходимость существования некоторого лица (группы лиц), на которое возлагаются функции администрирования данными, хранимыми в базе.

Методические указания к лабораторным работам по дисциплине «Базы данных» направлены на формирование представления о базах данных (БД) возможностях СУБД и их использовании, способствует развитию познавательной деятельности студентов, развитию у них творческого мышления, умений и навыков самостоятельной работы.

Основные функциональные возможности и технологические операции работы в СУБД в указаниях рассматриваются на основе привязки к конкретному типу программного продукта – СУБД MS Access. Знания, полученные при изучении данного материала, являются базовыми при практическом знакомстве с любым новым видом СУБД.

1. Создание таблиц базы данных

База данных организует и хранит информацию, располагая при этом всеми необходимыми средствами для пополнения и корректировки, составления запросов, выборки и отчетов.

База данных включает различные объекты для ввода, хранения и управления Вашей информацией. Рассмотрим основные объекты: таблицы, формы и отчеты.

Таблицы являются основой базы данных и состоят из столбцов и строк. Каждая строка таблицы называется записью. Описание объекта построено на выделении его характерных признаков. Для каждого признака отводится столбец. Каждый столбец в таблице содержит один тип информации и называется полем.

Форма позволяет более наглядно отобразить информацию, содержащуюся в одной записи. Формы удобно использовать как для ввода, так и для просмотра данных.

Запросы предназначены для получения информации по заданным критериям, но источником является таблицы, в которых хранятся данные.

Таблицы – основа базы данных. Именно в таблицах хранится информация, организованная определенным образом.

Рассмотрим последовательность необходимых действий при создании файла и таблиц базы данных. База данных «Учебный процесс» будет состоять из семи таблиц, свойства по лей которых приведены в приложении 1.

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

Зададим имя файла, определим папку, в которой будет размещен этот файл и щелкаем кнопкой мыши на значке **Создать** (рис 1.2).

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.

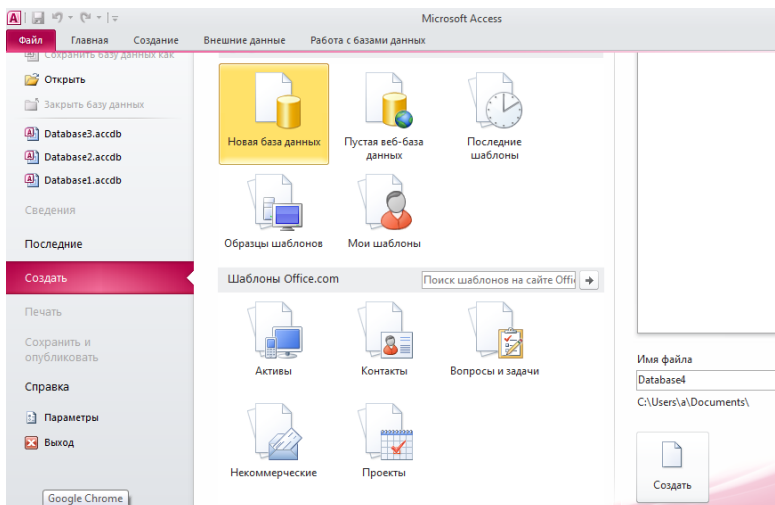


Рис. 1.1. Создание файла базы данных.

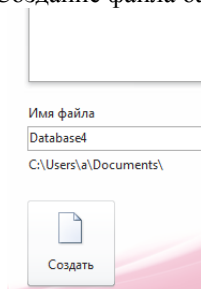


Рис. 1.2. Именованная база данных.

Таблица 1.1. Основные параметры структуры таблицы ГРУППА

Имя поля	Ключевое поле	Обязательное поле	Тип данных	Размер	Число десятичных знаков	Подпись поля	Условие на значение
НГ	Да	Да	Текстовый	3		Номер группы	
КО		Нет	Чис-	Байт		Коли-	≥ 0

Л			ло- вой			че- ство сту- ден- тов в груп- пе	And <=35 (Коли- чество сту- дентов боль- ше допу- стимо- го)
ПБ АЛ Л		Нет	Чис- ло- вой	Оди- нар- ное с пла- ваю- щей точ- кой	2	Про- ход- ной балл	>2 And <5 Or 0 (Ошиб- ка в оцен- ке)

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

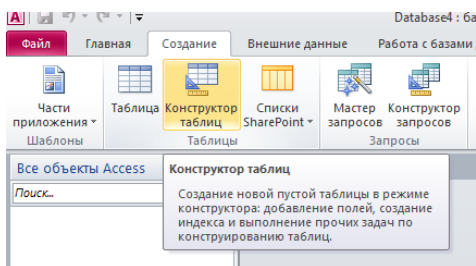


Рис 1.3. Вызов конструктора таблиц.

В соответствии с приведенными в табл. 1 проектными параметрами структуры для таблицы ГРУППА в окне конструктора **Таблица 1: 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 **Одинокое 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. 1.1

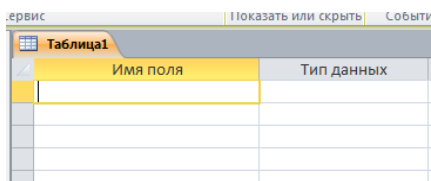


Рис 1.4. Окно конструктора таблиц.

Общие	Подстановка
Размер поля	255
Формат поля	
Маска ввода	
Подпись	
Значение по умолчанию	
Условие на значение	
Сообщение об ошибке	
Обязательное поле	Нет
Пустые строки	Да
Индексированное поле	Нет
Сжатие Юникод	Да
Режим ИМЕ	Нет контроля
Режим предложений ИМЕ	Нет
Смарт-теги	

Рис. 1.5. Вкладка Общие.

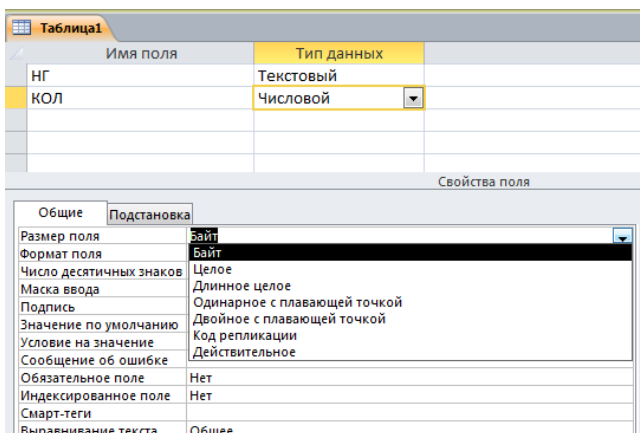


Рис. 1.6. Выбор размера для поля КОЛ (количество студентов в группе) в окне конструктора таблиц

Условие на значение, которое заносится в бланк запроса, является выражением, которое может быть сформировано с помощью построителя выражений (рис. 8). Построитель вызывается при нажатии кнопки справа от строки **Условие на значение**, в которую выражение должно быть введено (см. рис. 1.7).

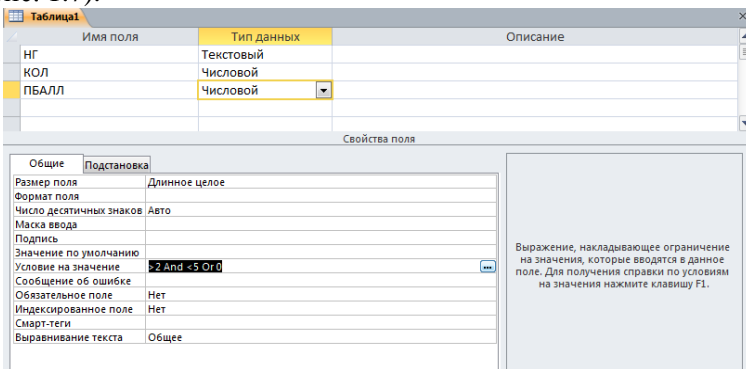


Рис. 1.7. Определение свойств поля ПБАЛЛ (средний проходной балл в группе)

Для ввода операторов <больше равно> и <меньше равно> существуют специальные знаки. Не допускайте пробел-

лов при вводе выражения. Нужные пробелы построитель введет сам.

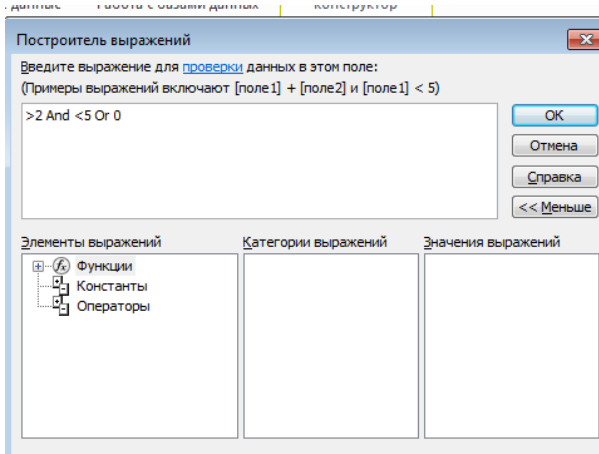


Рис. 1.8. Окно построителя выражений с выведенным списком всех операторов

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. 1.1.

Сохраним созданную структуру таблицы и присвоим имя новой таблице – «ГРУППА».

Выполним команду **Файл|Сохранить** и введем это имя в окне **Сохранения** (рис. 1.9).

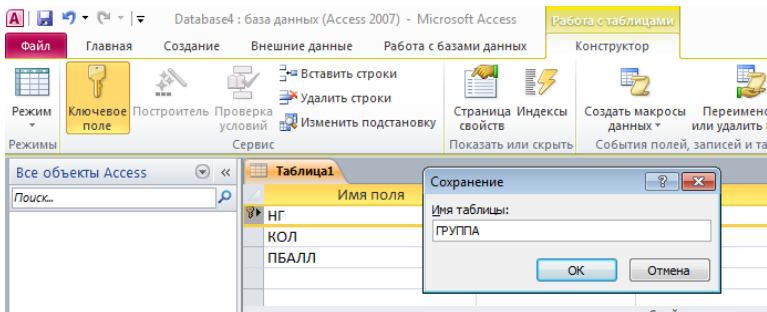


Рис. 1.9. Сохранение таблицы.

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

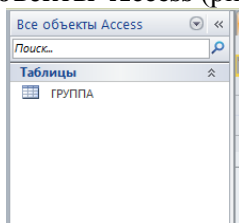


Рис. 1.10. Таблица ГРУППА.

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

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 фор-

матированные, 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. Последные 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 МЕМО

В таблице ПРЕДМЕТ предусмотрено поле ПРОГР, которое будет содержать текстовое данное большой длины – краткую программу курса. Для такого поля выбирается тип данного – **Поле МЕМО** (Мемо). 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.

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.

Задание

Создайте структуру таблиц КАФЕДРА, ПРЕДМЕТ, ПРЕПОДАВАТЕЛЬ, СТУДЕНТ, ИЗУЧЕНИЕ, УСПЕВАЕМОСТЬ. При создании таблиц используйте проектные параметры их структуры, которые представлены в таблицах 1.2-1.8.

Таблица 1.2. Описание свойств полей таблицы СТУДЕНТ

Имя поля	Ключевое 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		Нет	Текстовый	15		ФИО
ГОДР		Нет	Числовой	Целое		Год рождения
АДРЕС		Нет	Текстовый	25		Адрес проживания
ПБАЛЛ		Нет	Числовой	С плав. точкой	2	Проходной балл

Таблица 1.3. Описание свойств полей таблицы ГРУППА

Имя поля	Ключевое поле	Обязательное поле	Тип данных	Размер	Число десятичных знаков	Подпись поля	Условие на значение	Сообщение об ошибке
НГ	Да	Да	Текстовый	3		Номер группы		
КОЛ		Нет	Числовой			Количество студентов в группе	≥ 0 And ≤ 35	Количество студентов в группе от 0 до 35
ПБАЛЛ		Нет	Числовой	С плав. точкой	2	Проходной балл	> 2 And < 5 Or 0	Ошибка в оценке

Имя поля	Ключевое поле	Обязательное поле	Тип данных	Размер	Число десятичных знаков	Подпись поля	Условие на значение	Собщение об ошибке
				кой				

Таблица 1.4. Описание свойств полей таблицы КАФЕДРА

Имя поля	Ключевое поле	Обязательное поле	Тип данных	Размер	Подпись поля
ККАФ	Да	Да	Текстовый	2	Код кафедры
НКАФ		Нет	Текстовый	15	Название кафедры
ТЕЛ		Нет	Текстовый	9	Телефон кафедры
ЗАВ		Нет	Текстовый	15	ФИО заведующего кафедрой
ФОТО		Нет	Поле объекта OLE		Фотография заведующего кафедрой

Таблица 1.5. Описание свойств полей таблицы ПРЕПОДАВАТЕЛЬ

Имя поля	Ключевое поле	Обязательное поле	Тип данных	Размер	Подпись поля
ТАБН	Да	Да	Текстовый	4	Табельный номер
ФИО		Да	Текстовый	30	ФИО преподавателя
СТ		Нет	Текстовый	15	Ученая степень
ЗВ		Нет	Текстовый	10	Ученое звание
ККАФ		Да	Текстовый	2	Код кафедры

Таблица 1.6. Описание свойств полей таблицы ПРЕДМЕТ

Имя поля	Ключевое поле	Обязательное поле	Тип данных	Размер	Подпись поля	Условие на значение	Сообщение об ошибке
КП	Да	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	Число семестров		
ПРОГР		Нет	Поле MEMO		Программа		

Таблица 1.7. Описание свойств полей таблицы ИЗУЧЕНИЕ

Имя поля	Ключевое поле	Обязательное поле	Тип данных	Размер	Подпись поля
НГ	Да	Да	Текстовый	3	Номер группы
КП	Да	Да	Текстовый	2	Код предмета
ТАБН	Да	Да	Текстовый	4	Табельный номер преподавателя
ВИДЗ	Да	Да	Текстовый	3	Вид занятий
ЧАСЫ		Нет	Числовой	Целое	Средний балл по

Имя поля	Ключевое поле	Обязательное поле	Тип данных	Размер	Подпись поля
					предметам
СБАЛЛ.ГР		Нет	Числовой	С плав. точкой (2 знака)	Средний балл группы

Таблица 1.8. Описание свойств полей таблицы УСПЕВАЕМОСТЬ

Имя поля	Ключевое поле	Обязательное поле	Тип данных	Размер	Подпись поля
НГ	Да	Да	Текстовый	3	Номер группы
НС	Да	Да	Текстовый	2	Номер студента
КП	Да	Да	Текстовый	2	Код предмета
ТАБН	Да	Да	Текстовый	4	Табельный номер преподавателя
ВИДЗ	Да	Да	Текстовый	3	Вид занятий
ОЦЕНКА		Нет	Числовой	Целое	Оценка

2. Ввод данных в таблицы. Ввод записей в режиме таблицы

Заполнение таблиц данными возможно как непосредственным вводом данных, так и в результате выполнения программ и запросов.

Практически все СУБД позволяют вводить и корректировать данные в таблицах двумя способами:

- с помощью предоставляемой по умолчанию стандартной формы в виде таблицы;
- с помощью экранных форм, специально созданных для этого пользователем.

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	Телефон	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 языка	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 дру-

тому полю можно, нажав клавишу <Esc> или  на **Панели быстрого доступа**.

Если на **Панели быстрого доступа** отсутствует значок , 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** появится 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.

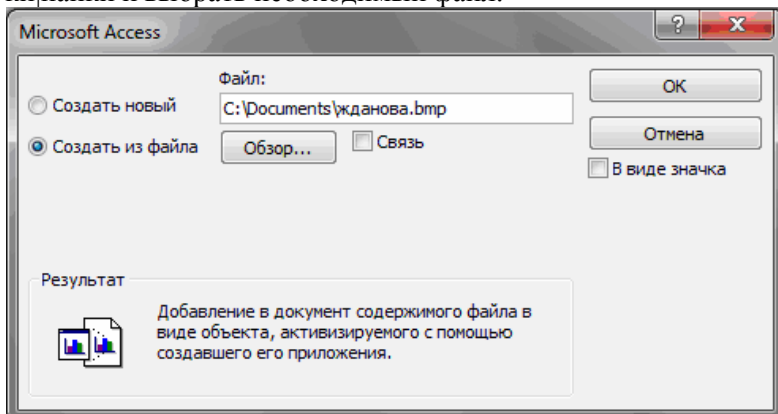


Рис. 2.1. Окно для выбора вставляемого объекта из файла

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 несоздана 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.

Таблица 2.2. Данные таблицы СТУДЕНТ

Группа	Номер студента в группе	ФИО	Год рождения	Адрес	Проходной балл
101	01	Аристов Р.П.	2003	Студенческая 15-12	4,25
101	02	Бондаренко С.А.	2004	Институтская 3-5	4,50
101	03	Борисова Е.И.	2003	Первомайская 3-9	4,25
101	04	Макова Н.В.	2002	Ленина 8-9	4,75
102	01	Боярская Н.П.	2002	Калинина 3-10	4,50
102	02	Федоров Д.К.	2001	Амурская 150-20	4,25
102	03	Сидоров И.Р.	2003	Василенко 2-15	4,50
103	01	Андреев Г.М.	2002	Институтская 10-6	4,25
103	02	Петров О.К.	2001	Воронкова 10-2	4,75
104	01	Иванов К.К.	2002	Ленина 150	4,50

Таблица 2.3. Данные таблицы ГРУППА

Номер группы	Количество студентов в группе	Проходной балл
101	30	4,5
102	32	4,5
103	29	4,8
104	35	4,4
105	35	4,8
201	35	3,9
202	30	4,0
203	28	4,7
204	25	4,0

Таблица 2.4. Данные таблицы ПРЕПОДАВАТЕЛЬ

Табельный номер	ФИО преподавателя	Уч. степень	Уч. звание	Код кафедры
101	Андреев А.П.	д-р техн.наук	профессор	01
102	Алухтин И.С.	канд.техн.наук	доцент	01
103	Глухов 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	Бломкина 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
30	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

Табель- ный но- мер	ФИО преподавателя	Уч. степень	Уч. звание	Код кафедры
304	Цветков А.И.	канд. ист.наук	доцент	03
401	Сорокина М.Ф.	канд.фил.на ук	доцент	04
402	Богомолов П.Р.	канд.фил.на ук	доцент	04
403	Лысова М.И.	канд.фил.на ук	доцент	04
404	Шаповалова М.Ф.	канд.фил.на ук	доцент	04
405	Кудряшова Г.М.		ассистент	04
501	Жигарева П.Р.	канд. пед.наук	доцент	05
502	Егорова Т.И.		ст. препода- ватель	05
503	Ермолин Е.Н.		ассистент	05
601	Логинов А.М.	канд.фил.на ук	доцент	06
602	Яковлев П.П.	канд.фил.на ук	доцент	06
603	Раков А.В.	канд.фил.на ук	доцент	06
604	Соловьев С.И.		ассистент	06

Таблица 2.5. Данные таблицы ПРЕДМЕТ

Код предмета	Название предмета	Все- го ча- сов	Лек- ции	Практи- ка	Семест- ров
01	Информати- ка	200	80	120	4
02	Математика	200	100	100	4
03	История	140	90	50	3
04	Иностран-	200	0	200	4

Код предмета	Название предмета	Всего часов	Лекции	Практика	Семестров
	ный язык				
05	Философия	100	40	60	2
06	Физкультура	100	0	100	2

Таблица 2.6. Данные таблицы ИЗУЧЕНИЕ

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

Таблица 2.7. Данные таблицы УСПЕВАЕМОСТЬ

Номер группы	Номер студента	Код предмета	Таб. преподавателя	Вид занятия	Оценка
101	01	01	101	лек	5
101	01	03	302	пр	0

Номер группы	Номер студента	Код предмета	Таб. Номер преподавателя	Вид занятия	Оценка
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

3. Логическая структура базы данных

Логическая структура базы данных «Учебный процесс» 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 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.

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 **Заккрыть**. В результате в окне **Схема данных** будут представлены все таблицы базы данных «Учебный процесс» со списками своих полей (см. рис. 3.2).

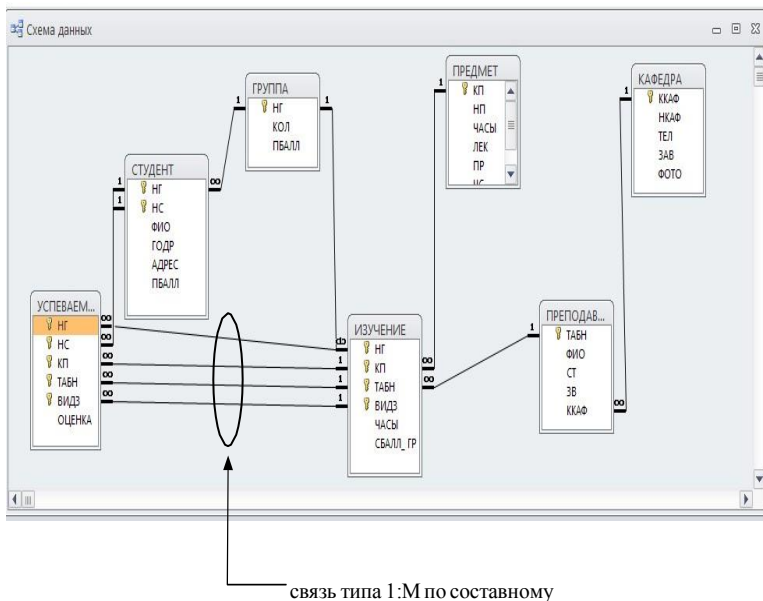


Рис. 3.1. Схема данных БД «Учебный процесс»

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 отношения** установится значение **один-ко-многим**. Отметим параметр **Обеспечение целостности данных**. Если таблицы 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. 3.2) для ключевого поля НС главной таблицы ТАБЛИЦА/ЗАПРОС выберем соответствующее поле подчиненной таблицы СВЯЗАННАЯ ТАБЛИЦА/ЗАПРОС. В этом же окне установим режимы **Обеспечение целостности данных** и другие параметры связи.

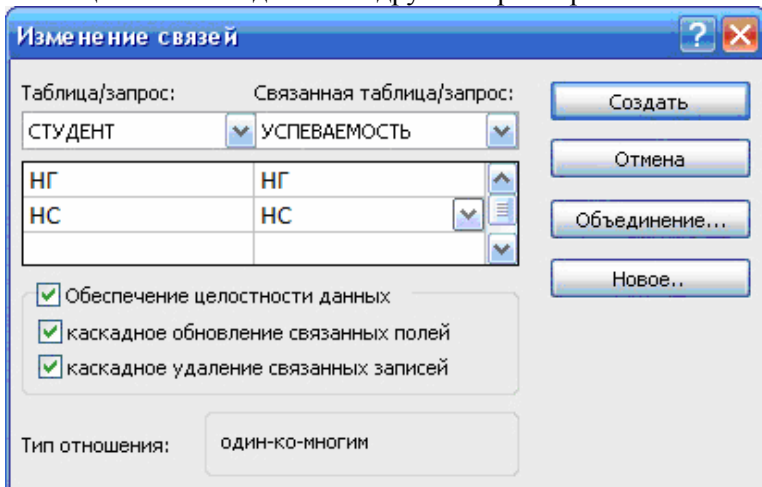


Рис. 3.2. Окно выбора параметров Связи

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. 3.1 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. 3.1 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-многочисленными 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 связанных полей** (см рис. 3.2). 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. 3.3).

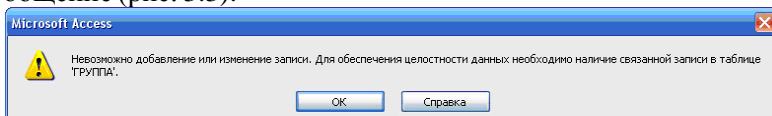


Рис.3.3. Сообщение о невозможности внесения изменений в таблицу
СТУДЕНТ

Определение технологии загрузки базы данных

Рассмотрим технологию загрузки на примере базы данных «Учебный процесс», в которой связи между таблицами соответствуют схеме данных, приведенной выше.

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

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

Загрузка таблицы УСПЕВАЕМОСТЬ может производиться только после загрузки таблиц СТУДЕНТ и ИЗУЧЕНИЕ.

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 Заголовков 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)

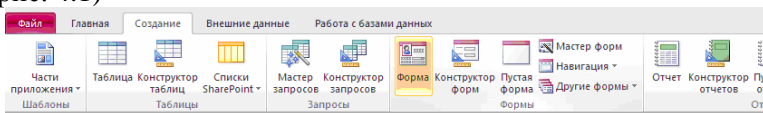


Рис 4.1. Кнопка **Форма** на вкладке **Создать** в группе **Формы**

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**.

ПРЕДМЕТ	
Код предмета	01
Название предмета	Информатика
Всего часов	200
Лекции	80
Практика	120
Семестров	4
Программа	

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

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 во вкладке **Режимы** кнопки **Режим** (рис 4.3). Кнопка **Режим**обеспечена списком, развернув который можно выбрать необходимый режим представления формы.

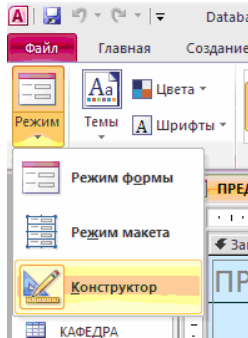


Рис. 4.3. Список режимов кнопки **Режим**

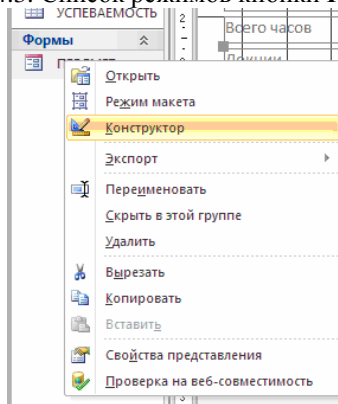


Рис. 4.4. Переход в режим конструктора

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

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

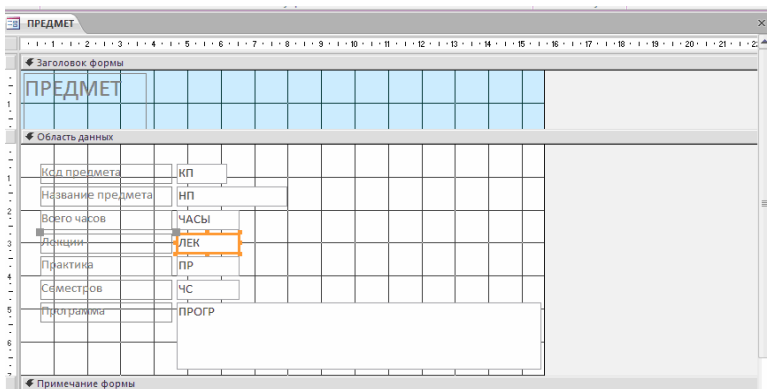


Рис. 4.5. Режим конструктора формы, полученной командой «Форма» по таблице ПРЕДМЕТ

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.

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.

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

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

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 .

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 - Закрывать** (File - Close) или

нажатием кнопки  (Закрывать) 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 нарис 4.6.

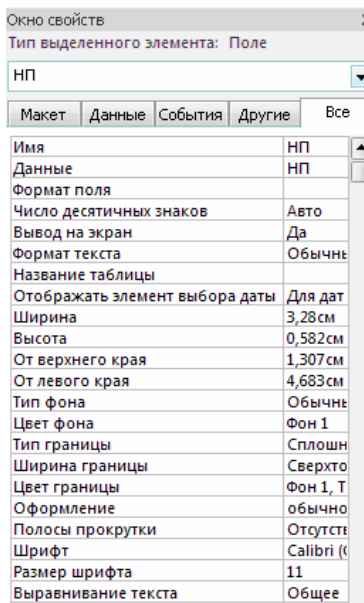


Рис 4.7. Свойства для поля НП в форме, установленные мастером при создании формы для таблицы ПРЕДМЕТ

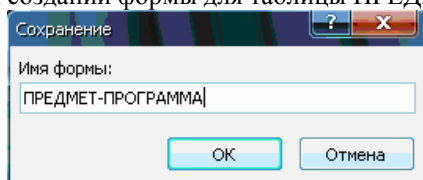


Рис. 4.8. Ввод имени формы при ее сохранении

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 во 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.

ПРЕДМЕТ1

Университет экономики и финансов

ПРЕДМЕТ И ЕГО ПРОГРАММА

Код предмета: Название предмета: Всего часов:

Лекции: Практика: Всего семестров:

Программа:

Записи: 1 из 6 | Нет фильтра | Поиск

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

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.

5. Формы для загрузки двух таблиц

В работе 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 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.

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 под схемой определим общую структуру составной формы, которую назовем СПИСОК ГРУППЫ.

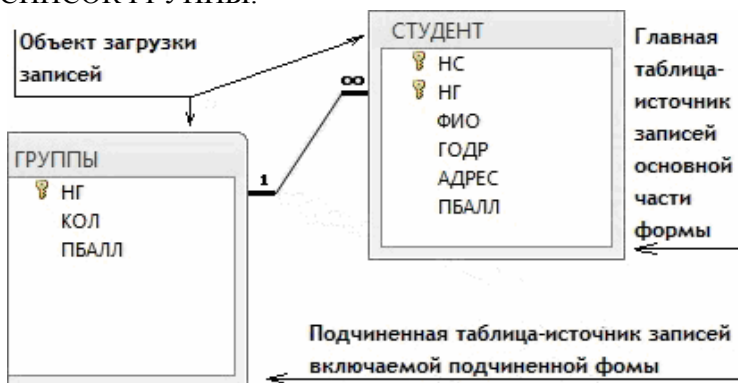


Рис.5.1. Подсхема данных для конструирования формы на базе таблиц ГРУППА → СТУДЕНТ

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».

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.

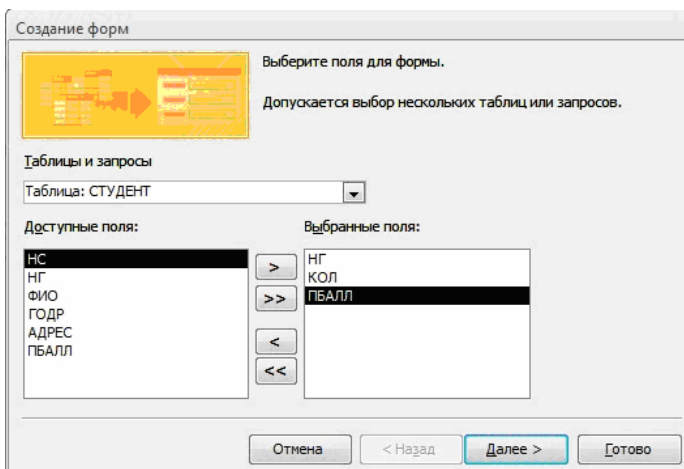


Рис. 5.2. Выбор полей для формы в окне мастера форм

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 **Доступные 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**.

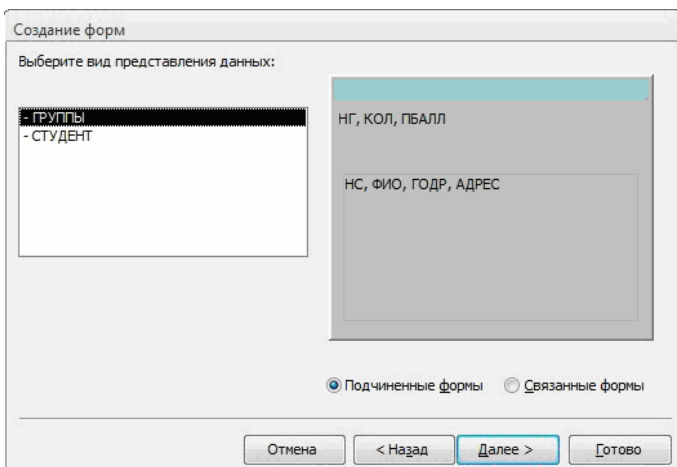


Рис. 5.3. Выбор варианта явного включения подчиненной формы

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**.

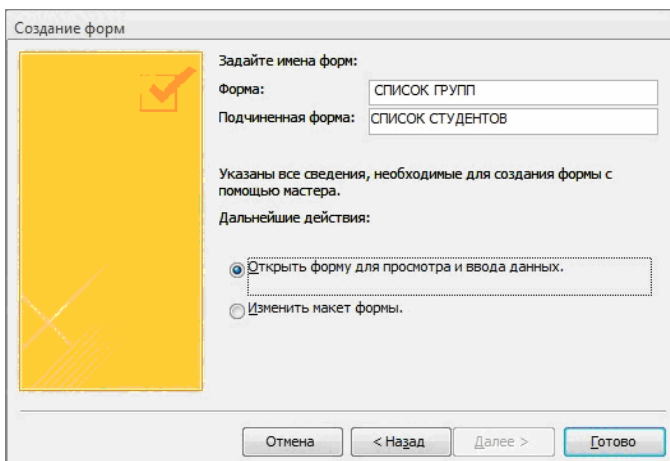


Рис. 5.4. Окно ввода имен форм и выбора дальнейших действий мастера

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

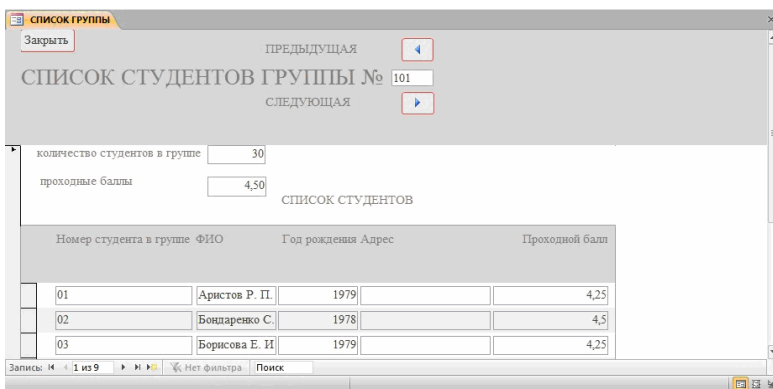


Рис. 5.5. Многотабличная форма, созданная мастером на основе таблиц ГРУППА и СТУДЕНТ

При этом в подчиненной форме выводятся те записи таблицы СТУДЕНТ, которые связаны с текущей записью таблицы ГРУППА, данные которой отображаются в основ-

ной части формы (рис. 5.5).

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.

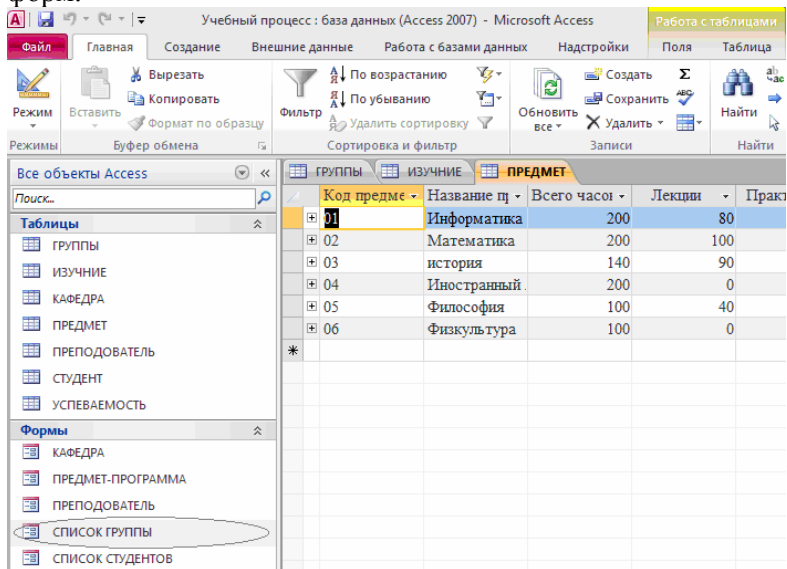


Рис. 5.6. Выбор формы в окне базы данных

В окне конструктора форм (рис. 5.7) указано имя формы, СПИСОК ГРУППЫ, представ- лены поля с надписями ос-

новой части формы, размещенные в области данных, а также поля и надписи подчиненной формы СПИСОК СТУДЕНТОВ в рамке, созданной для нее мастером.

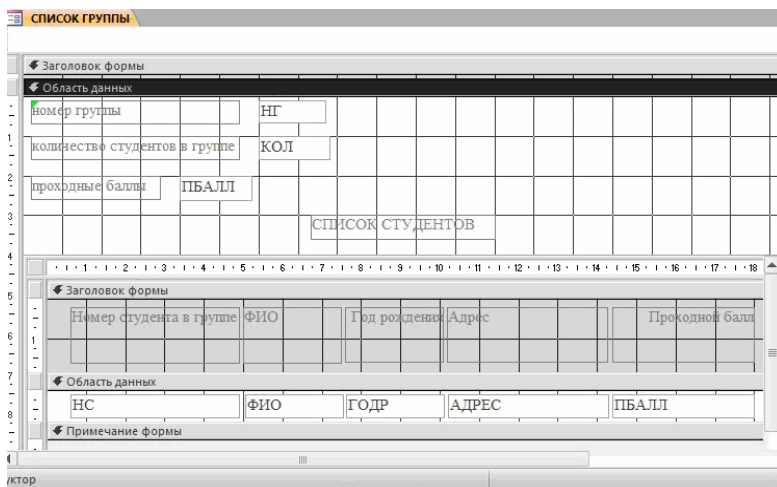


Рис. 5.7. Форма для двух таблиц ГРУППА и СТУДЕНТ в режиме конструктора

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.

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 | Выровнять | по нижнему краю**.

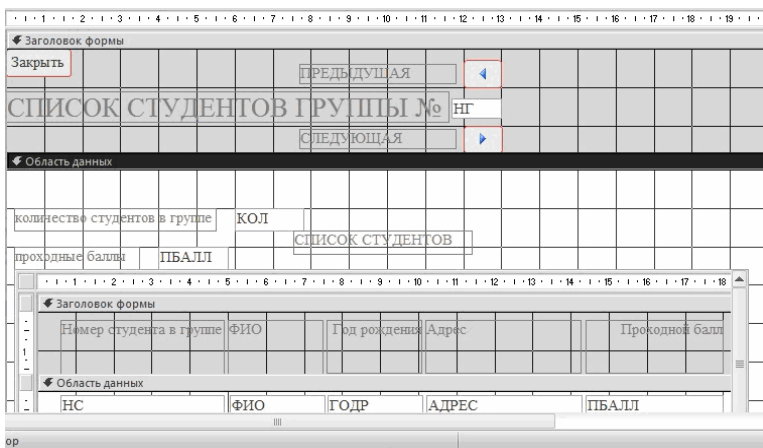


Рис. 5.8. Размещение элементов основной части формы после редактирования

Создание кнопок для перехода к другой записи

Для того чтобы в форме можно было переходить к следующей и предыдущей записям 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).

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** (синяя). 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».

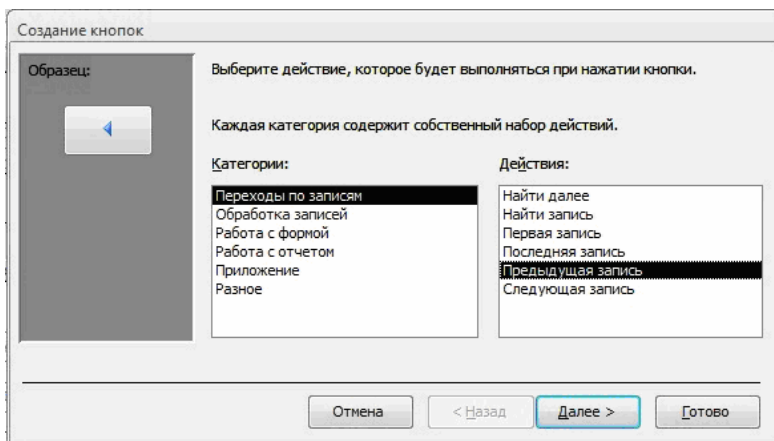


Рис.5.9. Окно мастера кнопок

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

Номер студента в группе	ФИО	Год рождения	Адрес	Проходной балл
01	Аристов Р. П.	1979		4,25
02	Бондаренко С.	1978		4,5

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 создан-


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

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  окнам Windows.

Задание.

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.

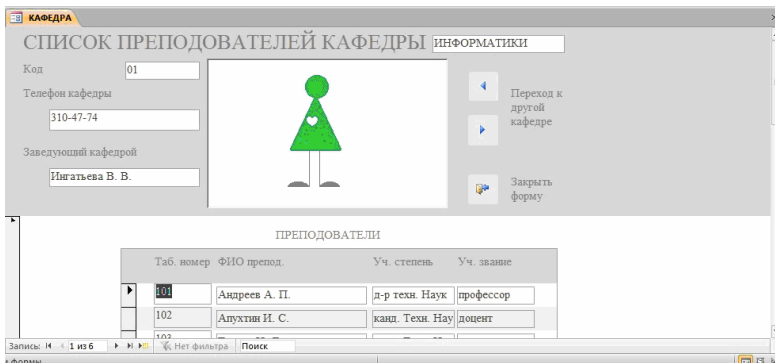


Рис. 5.12. Форма для загрузки и работы с таблицами КАФЕДРА → ПРЕПОДАВАТЕЛЬ

1. В соответствии с технологией загрузки базы данных, 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.

6. Многотабличные формы

В соответствии с этапами загрузки базы данных «Учебный процесс», определенными выше, 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 под схему данных для создания формы, определим общую структуру формы и размещение реквизитов в соответствии со структурой входного документа и под схемой 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 немашинной 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.

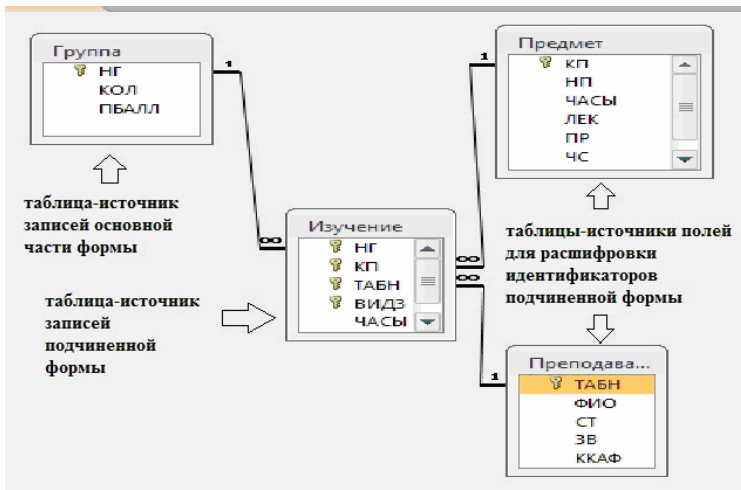


Рис. 6.1. Подсхема для формы ввода-вывода данных в таблицу ИЗУЧЕНИЕ

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 *не посредственное 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* -таблицаИЗУЧЕНИЕ

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 вво-

даться 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 группе*

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 окне **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.

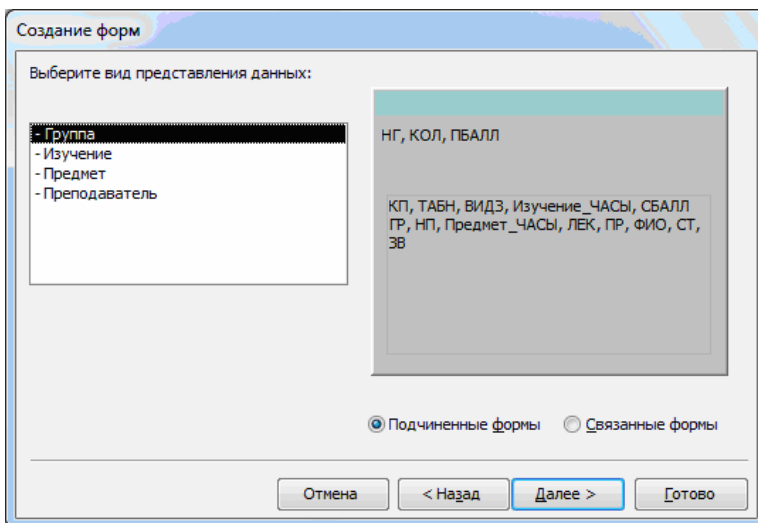


Рис. 6.2. Окно мастера форм при выборе типа включения подчиненной формы

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.

Имя группы:

Кол.ст.в.гр.:

Прох.балл:

Изучение

поля таблицы ИЗУЧЕНИЕ				поля таблиц ПРЕДМЕТ ПРЕПОДАВАТЕЛЬ					
Код предмета	Таб.н.преп.	Вид занятий	Изучение_ЧАСы	Ср. балл по предм.	Название предмета	Всего часов	Лекции	Практика	ФИО препода.
01	101	лек	20	0	Информатика	200	80	120	Андреев А. П.
01	102	пр	60	0	Информатика	200	80	120	Алутин И. С.
02	201	лек	25	0	Математика	200	100	100	Блюмкина И. П.
02	202	пр	50	0	Математика	200	100	100	Львова П. Р.
03	301	лек	30	0	История	140	90	50	Ильясев И. Т.
03	302	пр	20	0	История	140	90	50	Пустынцев А.П.
04	401	пр	50	0	Иностраный яз.	200	0	200	Сорокина М.Ф.
05	501	лек	20	0	Философия	100	40	60	Жигарева П. Р.

Запись: 1 из 10 | Нет фильтра | Поиск

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 пользова-

телю 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 линии** 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.

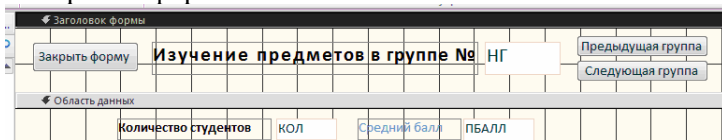


Рис. 6.4. Основная часть многотабличной формы после редактирования в режиме конструктора

Ограничение доступа к полям таблицы-источника основной части формы.

Защитим данные записей таблицы ГРУППА от произвольных изменений 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.

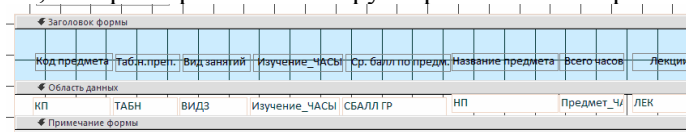


Рис. 6.5. Подчиненная форма ИЗУЧЕНИЕ в режиме конструктора

Для изменения вида подчиненной формы вызовем ее свойства. На вкладке **Макет** в окне **Окно свойств** (прав.кнопка мыши - пункт «свойства формы») (рис. 6.6) за-

меним 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.

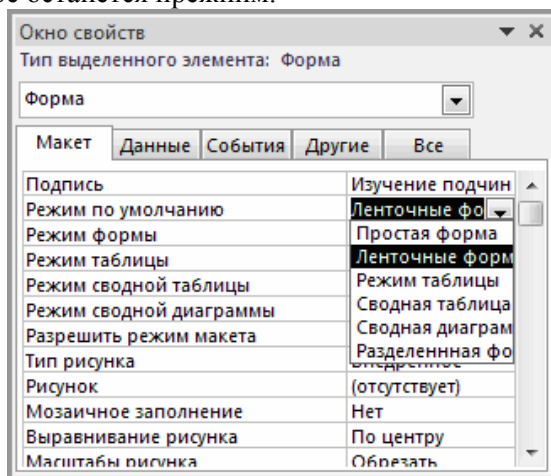


Рис. 6.6. Отображение списка режимов по умолчанию в окне свойств подчиненной формы

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 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.

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).

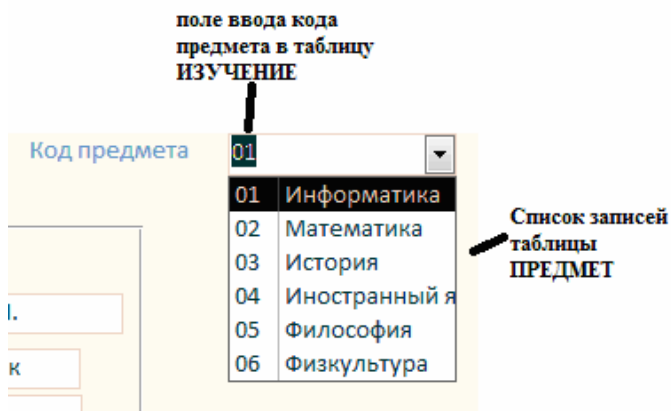


Рис. 6.8. Отображение списка

Создание поля со списком с помощью мастера

Создадим поле со списком для ввода значений кода предмета – КП в таблицу ИЗУЧЕНИЕ. Это позволит просматривать и вводить значения, которые уже имеются в главной таблице ПРЕДМЕТ, а также проверять соответствие 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, вычертим 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).

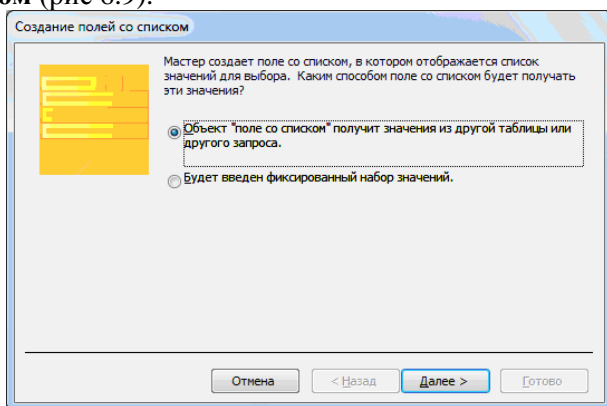


Рис. 6.9. Выбор способа форматирования списка

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

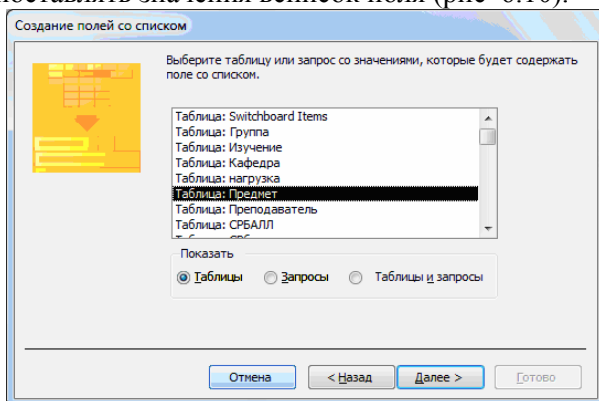


Рис. 6.10. Выбор источника данных для формирования списка

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

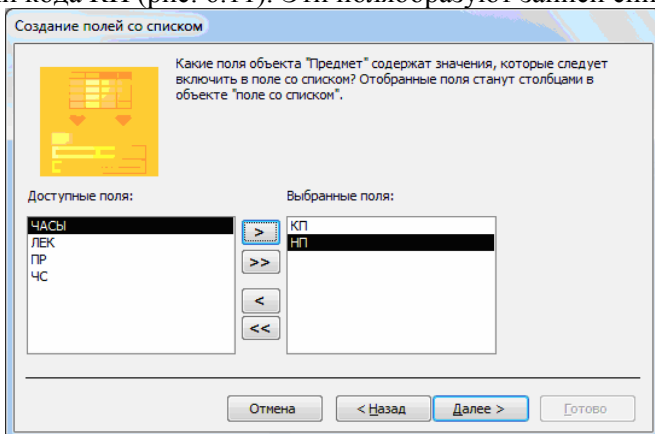


Рис. 6.11. Выбор полей для формирования записи списка

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

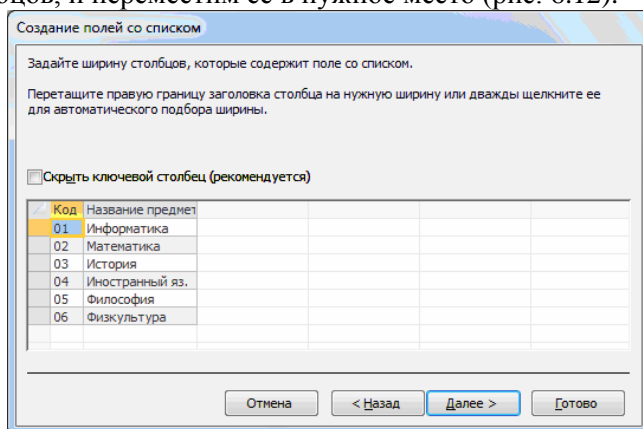


Рис. 6.12. Окно для оформления столбцов списка

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

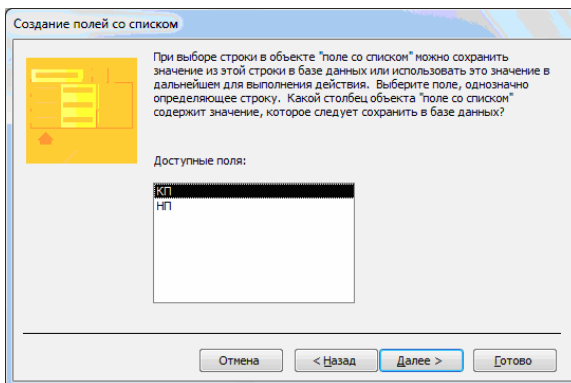


Рис. 6.13. Окно для выбора поля источника значений для поля формы

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

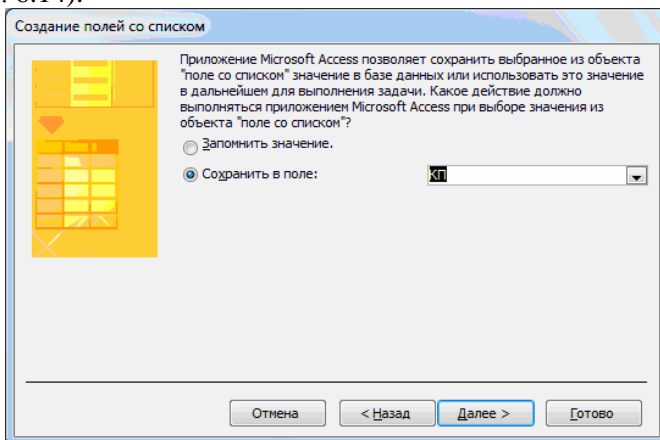


Рис. 6.14. Выбор поля формы, в которое вводится значение из списка

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

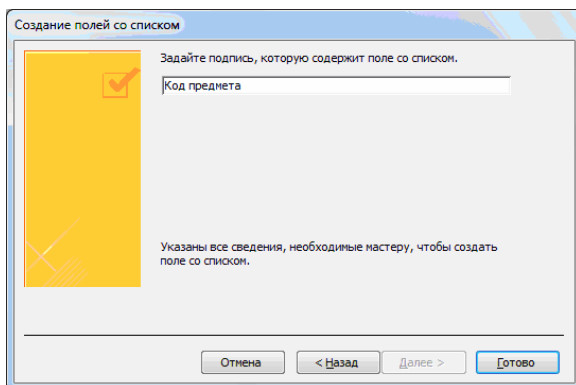


Рис. 6.15. Определение подписи поля со списком

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 значения в 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, имеющих 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 контекстно-зависимое меню. Выберем пункт **Свойства**, затем – вкладку **Данные**. В строке **Данные** выберем поле ТАБН, которое необходимо заполнять 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 .

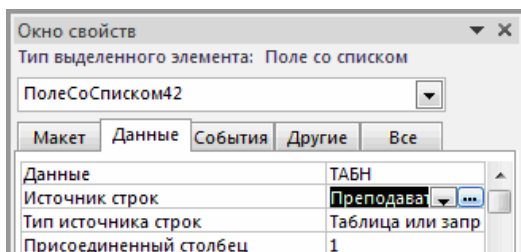


Рис. 6.17. Окно определения свойств данных поля со списком

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 списка 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

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.

Заккрыть форму **Изучение предметов в группе № 101** Предыдущая группа
Следующая группа

Количество студентов Средний балл

ВВЕДИТЕ ИДЕНТИФИКАТОР ЗАНЯТИЯ И ЧАСЫ

Код предмета Код преподавателя
Вид занятий Часы

справочные данные:

ПРЕДМЕТ		ПРЕПОДАВАТЕЛЬ	
Название предмета	<input type="text" value="Иформатика"/>	Фамилия И. О.	<input type="text" value="Андреев А. П."/>
Всего часов	<input type="text" value="200"/>	Уч. степень	<input type="text" value="д-р техн. наук"/>
Лекции	<input type="text" value="80"/>	Уч. звание	<input type="text" value="профессор"/>
Практика	<input type="text" value="120"/>		

ВНИМАНИЕ! Ввод значений в поля "Вид занятий", "Код предмета", "Код преподавателя" обязателен

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.

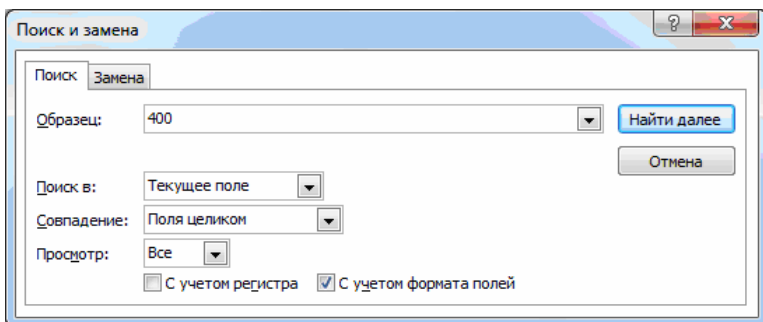


Рис. 6.20. Диалоговое окно поиска записи

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.

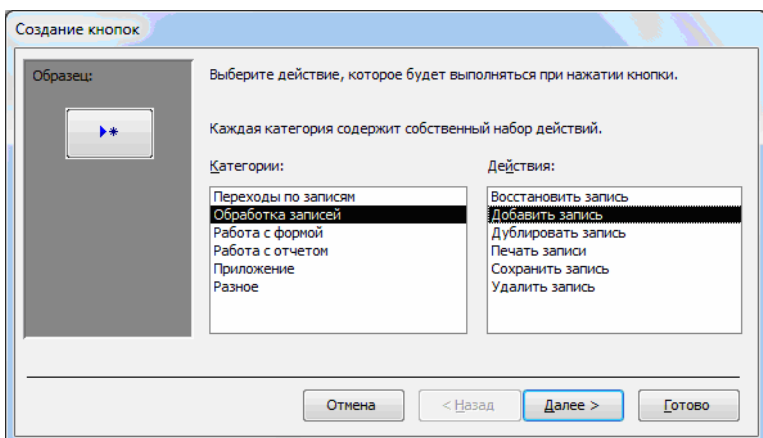


Рис. 6.21. Выбор параметров при создании кнопки для добавления записи

Установка свойств **Разрешить добавление**, **Разрешить удаление**, **Разрешить изменение**, обеспечивающих возмож-

ность добавления, удаления и изменения записей при загрузке и корректировке записей таблицы ИЗУЧЕНИЕ – источника записей, показана в окне свойств подчиненной формы (рис. 6.22).

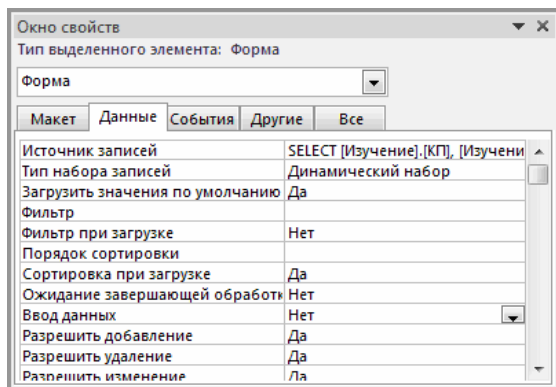


Рис. 6.22. Установка свойств подчиненной формы ИЗУЧЕНИЕ

Задание

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.

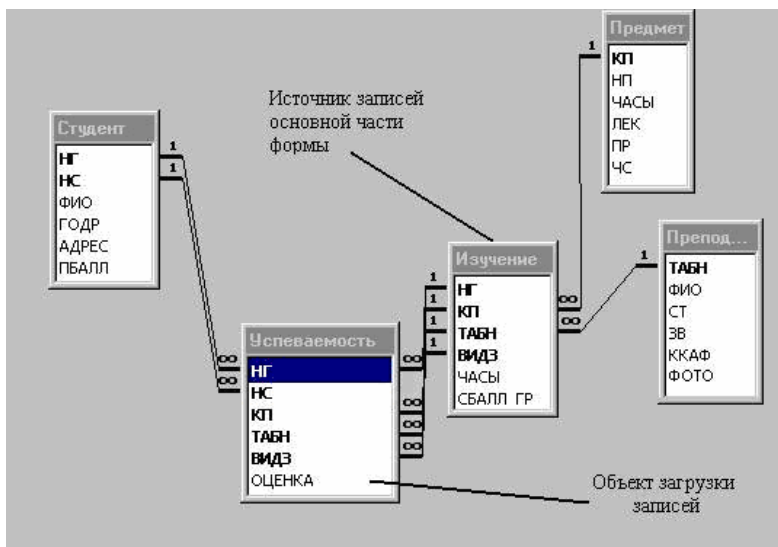


Рис. 6.23. Подсхема для составной формы ВЕДОМОСТЬ УСПЕВАЕМОСТИ

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

Успеваемость группы

ПРЕДМЕТ ПРЕПОДАВАТЕЛЬ

Вид занятий:
 Часы занятия:

Информатика
 Всего часов:

ФИО:
 Уч. степень:
 Уч. звание:

НС	ФАМИЛИЯ СТУДЕНТА	ОЦЕНКА
01	<input type="text" value="Аристов Р.П."/>	<input type="text" value="5"/>
02	<input type="text" value="Бондаренко С.А."/>	<input type="text" value="5"/>
03	<input type="text" value="Борисова Е.И."/>	<input type="text" value="4"/>
04	<input type="text" value="Макова Н.В."/>	<input type="text" value="3"/>
*	<input type="text"/>	<input type="text"/>

Средняя оценка в группе:

Записи: 14 | 1 из 4 | Нет фильтра | Поиск

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.

7. Запросы

В этой работе на конкретных примерах рассматривается технология конструирования запросов различного вида. Дано описание конкретных действий пользователя в процессе конструирования запросов. Подробно описан процесс конструирования однотоабличного и многотоабличного 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 однотоабличного запроса на выборку

Рассмотрим процесс конструирования однотоабличного 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**.

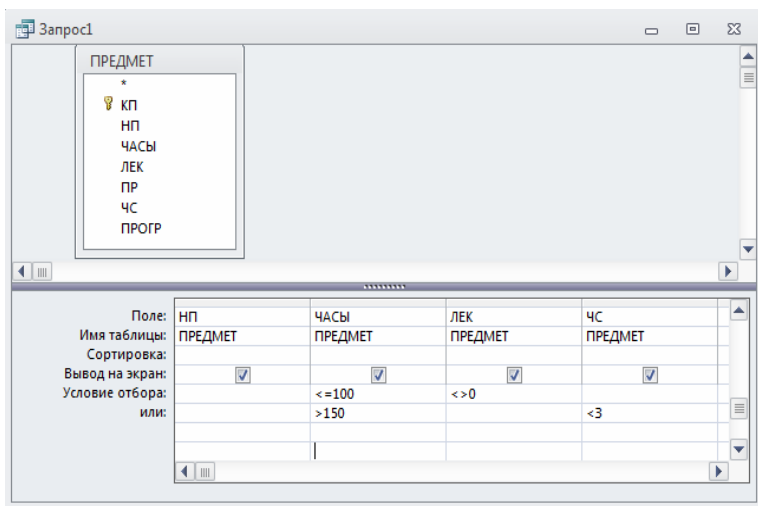


Рис. 7.1. Окно конструктора запроса на выборку с логическими операциями в условии отбора.

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**.

Сформулированные в задаче условия требуют формирования следующего логического выражения: $(ЧАСЫ \leq 100 \text{ AND } ЛЕК \neq 0) \text{ OR } (ЧАСЫ > 150 \text{ AND } ЧС < 3)$

Здесь $ЛЕК \neq 0$ (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 **И**. Условия из вторых скобок запишем в соответствующих полях ЧАСЫ и ЧС второй строки **Условие отбора**. Между условиями, записанными в разных строках, выполняется логическая операция **ИЛИ**.

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 **Заккрыть**. 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.

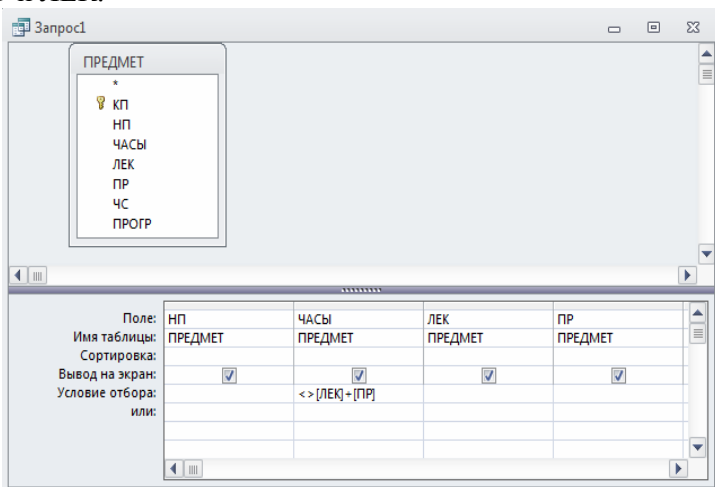


Рис. 7.2. Окно запроса на выборку из таблицы ПРЕДМЕТ записей, в которых количество часов не равно сумме часов лекций и практики

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.

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 **НП+НС** в соответствии с построенной ранее схемой данных.

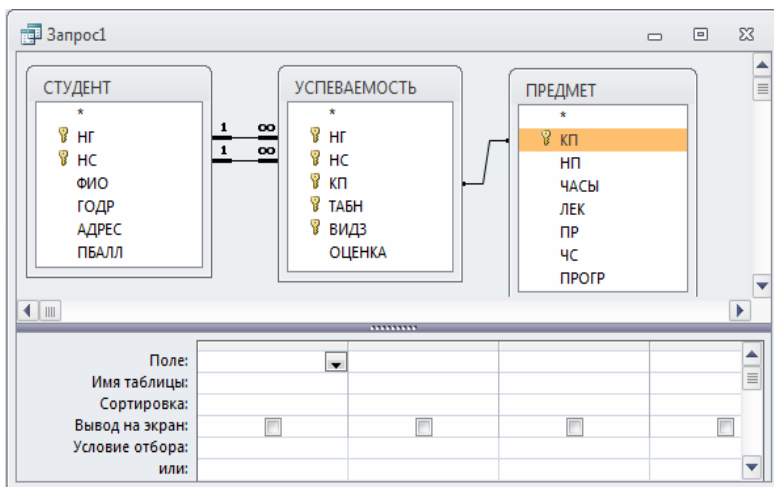


Рис. 7.3. окно запроса об успеваемости студента с созданной схемой данных.

Связь-объединение между УСПЕВАЕМОСТЬ и ПРЕДМЕТ поскольку эти таблицы имеют поля с одинаковым именем КП и одинаковым типом 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 инструментов конструктора или нажмем соответствующую кнопку в контекстном меню, который вызовем правой кнопкой мыши.

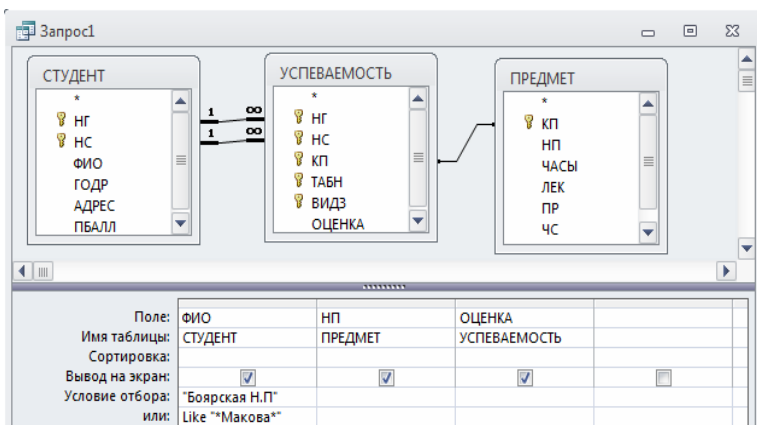


Рис. 7.4. Запрос на получение информации о сдаче экзаменов студентками Боярской Н.П. и Маковой.

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**.

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

Результат выполнения запроса об оценках заданных студентов.

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 условии 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 условии отбора поля НП введем второй параметр запроса: [Наименование предмета]

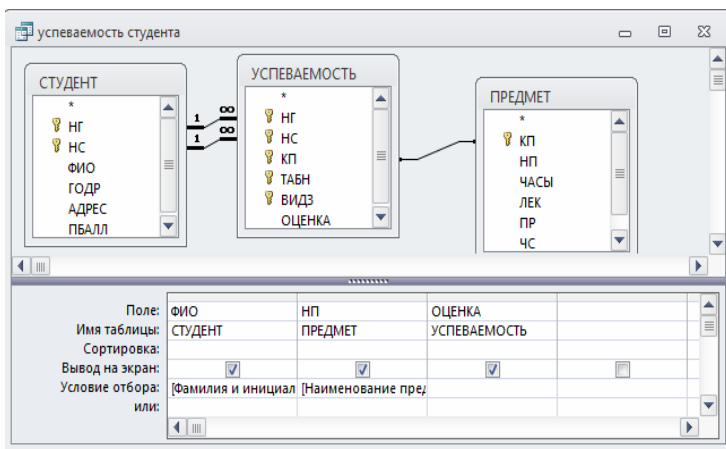


Рис. 7.5. Бланк запроса с параметрами для ввода ФИО и ИП.

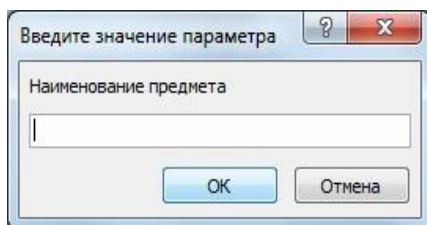
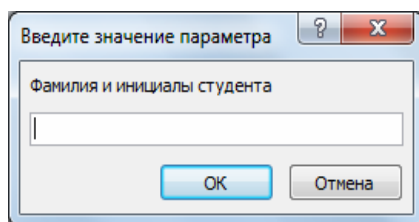


Рис. 7.6. Диалоговые окна для ввода параметров запроса.

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 ин-

форматике 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 ВИДЗ).

- **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 ВИДЗ (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: $[ИЗУЧЕНИЕ] \cdot [ЧАСЫ] \cdot [ЧС]$

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.

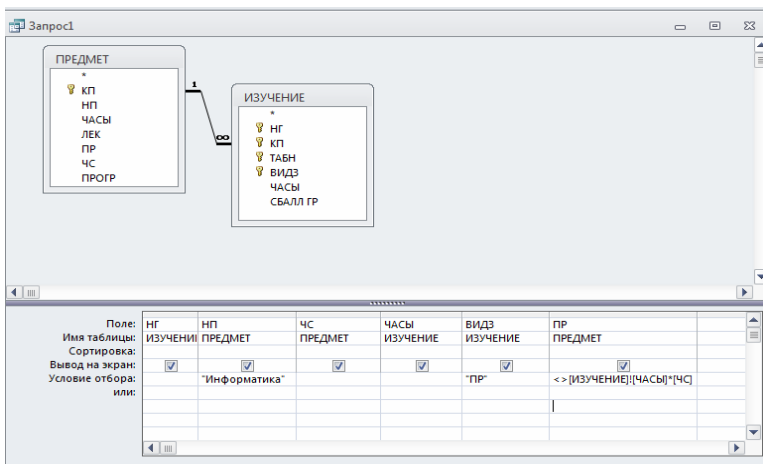


Рис. 7.7. Запрос с использованием в условиях отбора имен полей из разных таблиц.

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.

Ном. группы	Название п	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. Результат выполнения запроса

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).

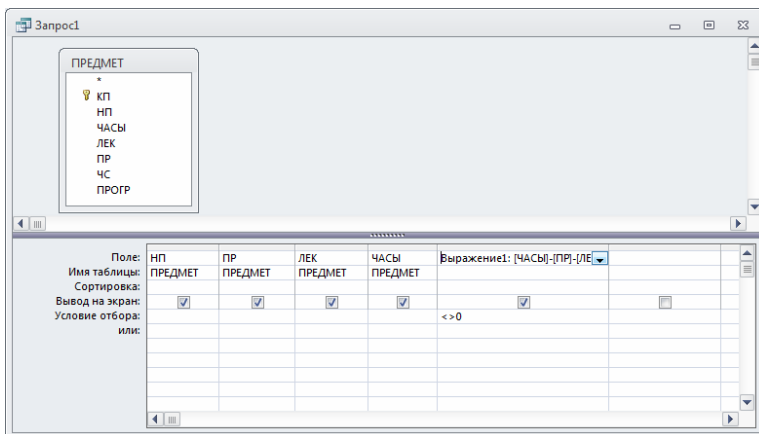


Рис. 7.9. Запрос с вычисляемым полем и условием отбора по его значению.

После ввода выражения система формирует имя вычисляемого поля по умолчанию - «Вы- ражение 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 на 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).

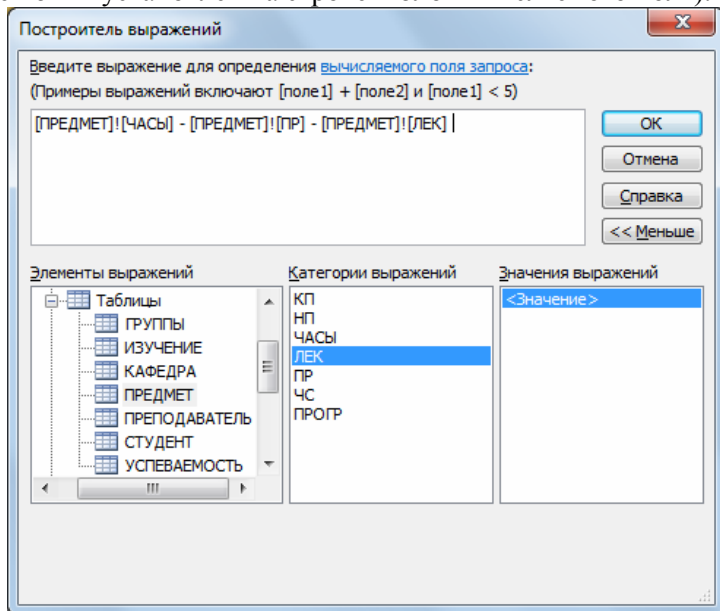


Рис. 7.10. Окно построителя выражений при формировании вычисляемого поля.

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 ча-

сти окна сформируется выражение.

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 однотобличного запроса с групповой операцией Рассмотрим технологию конструирования однотобличного 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.

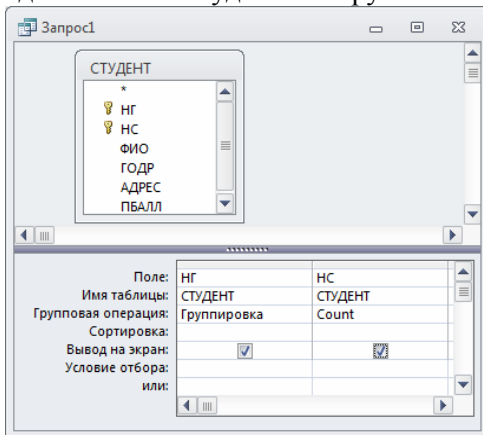


Рис. 7.11. Запрос с групповой операцией подсчета числа студентов в группе

Нажмем кнопку **Итоги**. Заменяем 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.

Группа	Count-НС
101	4
102	3
103	2
104	1

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".

Группа	Фактическое число студентов
101	4
102	3
103	2
104	1

Рис. 7.13. Таблица результата с измененной подписью поля

Таблица результата после доработки запроса показана на рис. 7.13.

Сохраним запрос на выборку под именем "Число студентов в группах".

Запрос с функцией Avg

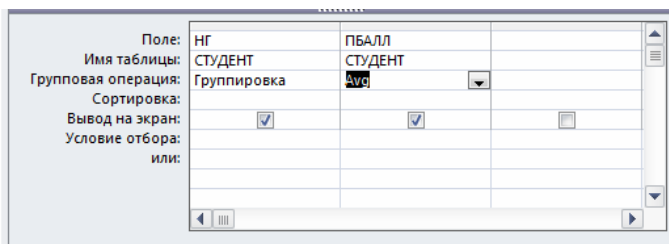


Рис. 7.14. Запрос с групповой операцией подсчета среднего значения

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.

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

Группа	Средний проходной балл группы
101	4,44
102	4,42
103	4,50
104	4,50

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 сред-

ний 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.

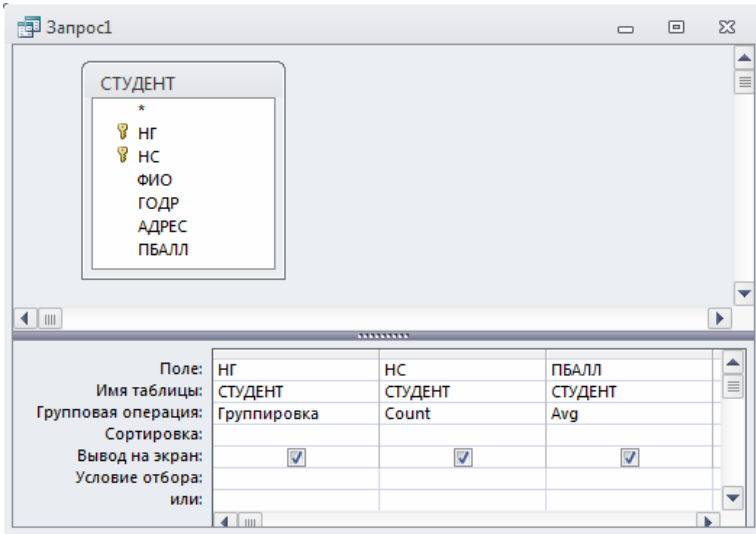


Рис. 7.16. Запрос с одновременным использованием двух групповых функций

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).

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

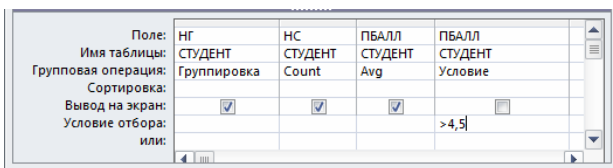


Рис. 7.17. Запрос с групповыми операциями и полем, введенным для определения условий отбора записей из таблицы СТУДЕНТ

Условие отбора, заданное в поле, по которому проводится группировка, или в поле, где записана функция группировки, позволяет отобразить 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).

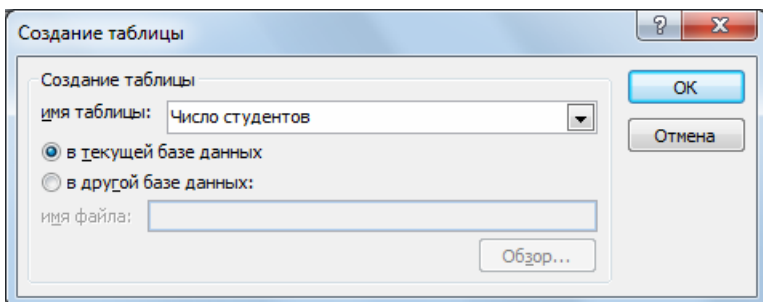


Рис. 7.18. Определение имени таблицы, создаваемой в запросе

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.

Задание

Преобразуйте запрос на выборку Средний проходной балл группы (см. рис. 7.14, 7.15) 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**.

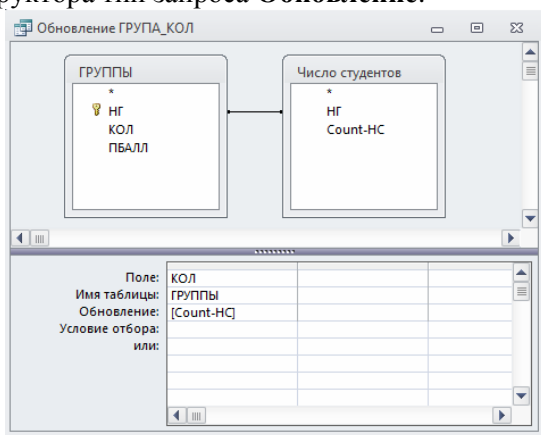


Рис. 7.19. Запрос на обновление таблицы ГРУППА

Заполним бланк запроса. Перетащим обновляемое поле КОЛ из списка таблицы ГРУППА. В строке **Обновление** введем имя поля "Count_NC" (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".

Задание

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.

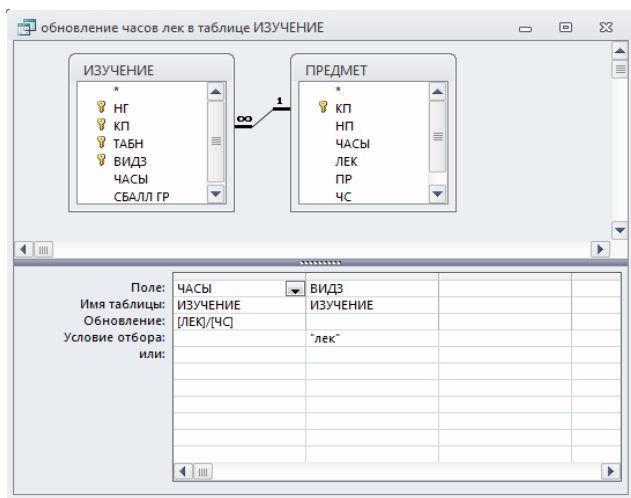


Рис.7.20. Запрос на обновление поля значениями, вычисляемыми по данным из другой таблицы

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

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 по-

лученной 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.

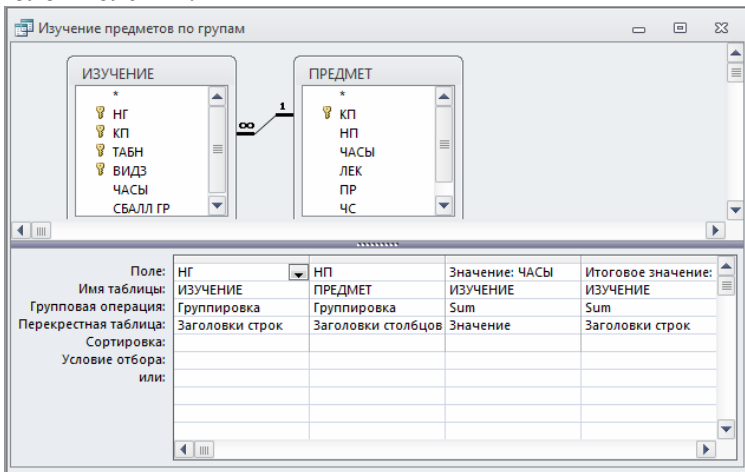


Рис. 7.21. Перекрестный запрос в режиме конструктора

Для изменения подписи поля ИТОГОВОЕ ЗНАЧЕНИЕ,

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

Ном. группы	Всего часов	Иностранцы	Информати	История	Математик	Физкультур	Философия
101	518	50	100	68	100	100	100
102	380	100	280				
105	100		100				
201	250		180		70		
202	100	100					
203	100		100				
204	100						100

Рис. 7.22. Результат выполнения перекрестного запроса

Результат выполнения полученного перекрестного запроса приведен на рис. 7.22.

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 ука-

занному в строке предмету. Число строк в таблице запроса будет равно числу строк в таблице УСПЕВАЕМОСТЬ.

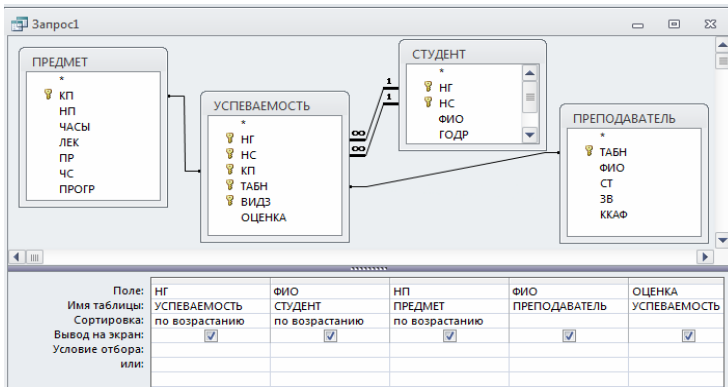


Рис.7.24. Многотабличный запрос об оценках студентов

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.

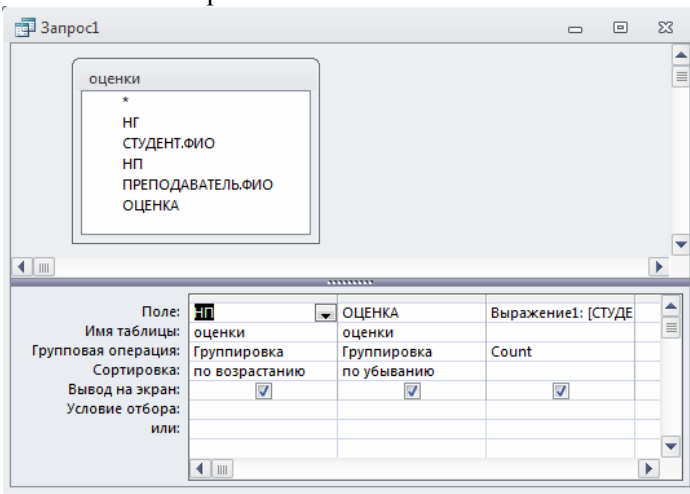


Рис.7.25. Построение запроса на основе запроса оценки

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

Название п.	ОЦЕНКА	Выражение
Информатика	5	2
Информатика	4	1
Информатика	3	1
История	0	4

Рис. 7.26. Результаты запроса, построенного на другом запросе

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.

Задание

- 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 НКДФ, по которому и производится группировка. Число преподавателей кафедры Count_ТАБН никак не нарушает требуемого объединения записей в группы, поскольку для каждой кафедры является единственным. Число записей в таблице результата запроса равно числу кафедр.

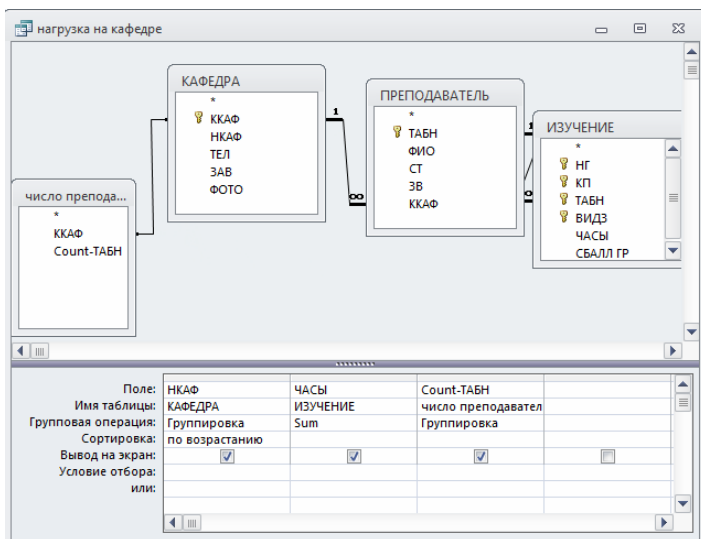


Рис. 7.27. Запрос для подсчета общего числа часов занятий, проводимых каждой кафедрой

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. Таблица, 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.

Название	Sum-ЧАСы	Count-ТАБН
ИНОСТРАННОГО ЯЗ	250	5
ИНФОРМАТИКИ	760	5
ИСТОРИИ	68	4
МАТЕМАТИКИ	170	4
ФИЗКУЛЬТУРЫ	200	3
ФИЛОСОФИИ	100	4

Рис. 7.28. Запрос Нагрузка на кафедре в режиме таблицы

НКАФ	Sum-ЧАСы	Count-ТАБН
ИНОСТРАННО	250	5
ИНФОРМАТИК	760	5
ИСТОРИИ	68	4
МАТЕМАТИКИ	170	4
ФИЗКУЛЬТУРЬ	200	3
ФИЛОСОФИИ	100	4
*		

Рис. 7.29. Сохраненная таблица НАГРУЗКА с результатами второго запроса

Третий запрос. Для окончательного решения задачи расчета средней нагрузки преподавателя кафедры подготовим на базе таблицы НАГРУЗКА третий запрос на выборку с вычисляемым полем. Для создания вычисляемого поля, рассчитывающего 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 столб-

ца **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"

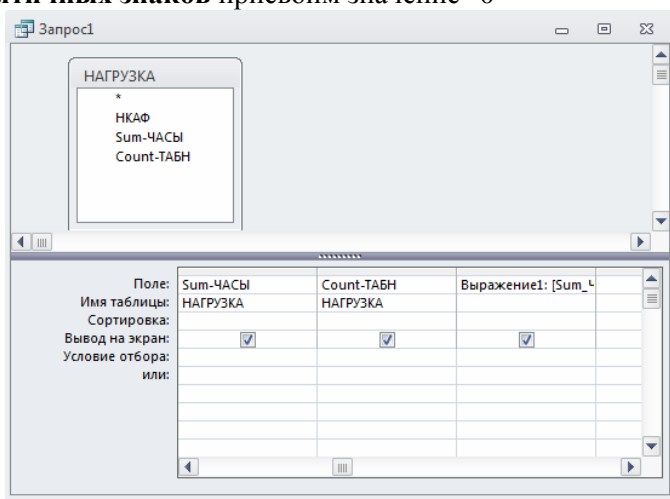


Рис. 7.30. Запрос, завершающий задачу расчета средней нагрузки

НКАФ	Всего часов	Количество	Средняя нагрузка
ИНОСТРАННО	170	5	34
ИНФОРМАТИК	500	5	100
ИСТОРИИ	30	4	7
МАТЕМАТИКИ	145	4	36
ФИЗКУЛЬТУРЕ	170	3	56
ФИЛОСОФИИ	0	4	0

Рис. 7.31. Результаты расчета средней нагрузки преподавателей по кафедрам

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

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 преподавателей 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.

СОДЕРЖАНИЕ

Введение	3
Создание таблиц базы данных	5
Ввод данных в таблицы. Ввод записей в режиме таблицы	18
Логическая структура базы данных	26
Однотабличные формы	32
Формы для загрузки двух таблиц	41
Многотабличные формы	54
Запросы	76