

**МИНОБРНАУКИ РОССИИ**

**Орский гуманитарно-технологический институт (филиал)  
федерального государственного бюджетного образовательного учреждения  
высшего образования «Оренбургский государственный университет»  
(Орский гуманитарно-технологический институт (филиал) ОГУ)**

Кафедра программного обеспечения

**Методические указания по выполнению и защите лабораторных работ  
по дисциплине «Б.1.В.ОД.2 Информационные технологии в экономике»**

Уровень высшего образования

**БАКАЛАВРИАТ**

Направление подготовки

38.03.01 Экономика

(код и наименование направления подготовки)

Бухгалтерский учет, анализ и аудит

(наименование направленности (профиля) образовательной программы)

Тип образовательной программы

Программа академического бакалавриата

Квалификация

Бакалавр

Форма обучения

Очная, заочная

Год начала реализации программы (набора)

2015, 2016, 2017

Методические указания предназначены для обучающихся очной и заочной форм обучения направления подготовки 38.03.01 Экономика профиля Бухгалтерский учет, анализ и аудит по дисциплине «Б.1.В.ОД.2 Информационные технологии в экономике»

Составитель \_\_\_\_\_  В.С. Богданова

Методические указания рассмотрены и одобрены на заседании кафедры программного обеспечения, протокол № 10 от «07» июня 2017 г.

Заведующий кафедрой \_\_\_\_\_  Е.Е. Сурина

Согласовано:  
Председатель методической комиссии по направлению подготовки 38.03.01 Экономика

\_\_\_\_\_  Т.В. Андреева  
«14» июня 2017 г.

© Богданова В.С., 2017  
© Орский гуманитарно-технологический институт (филиал) ОГУ, 2017

## Пояснительная записка

В результате изучения дисциплины «Б.1.В.ОД.2 Информационные технологии в экономике» у обучающихся должны быть сформированы знания, умения и навыки:

- 1) сбора, анализа и обработки данных, необходимых для решения профессиональных задач;
- 2) принципы работы с деловой информацией, а также иметь представление о корпоративных информационных системах и базах данных.
- 3) применения информационных технологии для решения задач в различных сферах деятельности.

Одной из наиболее эффективных форм закрепления теоретических знаний и выработки навыков самостоятельной работы являются лабораторные работы.

Целью проведения лабораторных работ является:

- закрепление знаний студентов по основам информационным технологиям в экономике,
- формирование у студентов навыков использования современных технических средств и технологий для решения аналитических и исследовательских задач.

## Тематический план

Таблица 1 – Тематический план выполнения лабораторных работ по дисциплине «Информационные технологии в экономике» для обучающихся направления подготовки 38.03.01 Экономика профиль подготовки Бухгалтерский учет, анализ и аудит:

а) очная форма обучения

№ п/п	Раздел / тема лабораторной работы	№ лабораторной работы	Количество часов
<b>2</b>	<b>MS Excel 2010. Ввод и редактирование таблиц. Создание формул. Мастер функций. Построение графиков и диаграмм. Решение основных экономических задач. Преобразование массивов информации в электронные таблицы</b>	-	<b>24</b>
2.1	формулы и функции MS Excel	1	2
2.2	функции даты и времени	2	2
2.3	функции просмотра и ссылок	3	2
2.4	финансовые функции MS Excel	4	2
2.5	подбор параметра	5	2
2.6	таблица подстановки	6	2
2.7	диспетчер сценариев	7	2
2.8	диаграммы и графики	8	2
2.9	расширенный фильтр MS Excel	9	2
2.10	автоматическое подведение итогов в MS Excel	10	2
2.11	консолидация данных	11	2
2.12	сводные таблицы	12	2
<b>3</b>	<b>MS Access 2010. Создание реляционной базы данных, отображающей информационное содержание</b>	-	<b>10</b>
3.1	создание базы данных MS Access	13	2
3.2	формирование запросов MS Access	14	2
3.3	закрепление изученного материала	15	2
3.4	проблемы манипулирования данными и запросы на создание таблиц MS Access	16, 17	4
	<b>ИТОГО</b>	-	<b>34</b>

а) заочная форма обучения

№ п/п	Раздел / тема лабораторной работы	№ лабораторной работы	Количество часов
<b>2</b>	<b>MS Excel 2010. Ввод и редактирование таблиц. Создание формул. Мастер функций. Построение графиков и диаграмм. Решение основных экономических задач. Преобразование массивов информации в электронные таблицы</b>	-	<b>4</b>
2.1	расширенный фильтр MS Excel	1	1
2.2	автоматическое подведение итогов в MS Excel	1	1
2.3	консолидация данных	2	1
2.4	сводные таблицы	2	1
<b>3</b>	<b>MS Access 2010. Создание реляционной базы данных, отображающей информационное содержание</b>	-	<b>2</b>
3.1	создание базы данных MS Access	3	2
	<b>ИТОГО</b>	-	<b>6</b>

### **Методические указания по выполнению и оформлению лабораторных работ**

Лабораторные работы по дисциплине «Информационные технологии в экономике» предполагают решение задач по темам, представленным в тематическом плане.

В лабораторной работе должны быть выполнены все предусмотренные задания. В работе должна просматриваться логическая последовательность и взаимная увязка основных частей работы.

Рекомендуемая структура лабораторных работ:

- 1) цель лабораторной работы;
- 2) задание в соответствии с выбранным вариантом;
- 3) теоретическая часть, включающая краткое изложение теоретических положений по теме лабораторной работы, формулы для решения задания;
- 4) практическая часть, включающая решение задания по теме лабораторной работы. Дополнительно для наглядности расчетный материал может быть представлен в виде таблиц, графиков;

5) выводы по лабораторной работе;

6) список использованной литературы.

Лабораторные работы могут быть оформлены:

- машинописным текстом на листах формата А4.

Титульный лист оформляется на основе СТО 02069024.101-2015 «Работы студенческие. Общие требования и правила оформления». Режим доступа: [http://www.osu.ru/docs/official/standart/standart\\_101-2015.pdf](http://www.osu.ru/docs/official/standart/standart_101-2015.pdf). Работа защищается устно и принимается к зачету, если нет замечаний по ее выполнению и оформлению. При отсутствии зачетных лабораторных работ студент не допускается к дифференцированному зачету по дисциплине «Информационные технологии в экономике».

## Лабораторная работа Формулы и функции MS Excel

Подготовить соответствующие ведомости, воспользовавшись приведенными рекомендациями.

**Задание 1.** Сформировать объявления о продаже квартир согласно образцу (рис. 1).

G	
1	<b>Объявления - Продажа квартир</b>
2	2-х кв., по ул. Соломовой, площадь: 48.6-31.4, 5 <sup>1/2</sup> этаж, 11 400р., телефон
3	2-х кв., по ул. Советская, площадь: 48.8-27.8, 1 <sup>1/2</sup> этаж, 12 500р., телефона нет
4	3-х кв., по ул. Медногорская, площадь: 65.8-41.2, 8 <sup>1/2</sup> этаж, 16 000р., телефон
5	3-х кв., по ул. Стартовая, площадь: 65.1-38.9, 9 <sup>1/2</sup> этаж, 15 700р., телефон

Рис. 1. Объявления о продаже квартир

### Рекомендации по созданию ведомости объявлений

Сгруппировать имеющиеся данные по квартирам в виде списка (рис. 2).

	A	B	C	D	E	F
1	Кол-во комнат	Адрес	Цена	Площадь	Этаж	Тел
2	2-х	ул. Соломовой	11400	48.6-31.4	5 <sup>1/2</sup>	+
3	2-х	ул. Советская	12500	48.8-27.8	1 <sup>1/2</sup>	-
4	3-х	ул. Медногорская	16000	65.8-41.2	8 <sup>1/2</sup>	+
5	3-х	ул. Стартовая	15700	65.1-38.9	9 <sup>1/2</sup>	+

Рис. 2. Данные о квартирах, выставленных на продажу

В ячейку **G2** ввести формулу:

=A2&" кв., по "&B2&", площадь: "SD2&", "&E2&"этаж, "&ТЕКСТ(C2;"# ##0р.")&", "&ЕСЛИ(F2="+"; "телефон"; "телефона нет")

Объясните данную формулу, воспользовавшись справкой.

Для диапазона **G3:G5** воспользоваться маркером автозаполнения либо скопировать данную формулу.

При необходимости отформатировать полученные объявления, используя команду **Формат | Автоформат**.

**Задание 2.** Сформировать и заполнить ведомость переоценки основных средств производства по форме, приведенной на рис. 3.

### Рекомендации по созданию ведомости

В ячейку **A1** ввести название ведомости.

В ячейки **A4:F4** ввести названия полей ведомости: Наименование объекта, Балансовая стоимость (БС), Износ объекта (ИО), Остаточная стоимость (ОС), Восстановительная полная стоимость (ВПС), Восстановительная остаточная стоимость (ВОС). **Поле** Наименование объекта **включает следующие строки:** Отдел менеджмента и маркетинга, Отдел транспортировок, Сборочный цех, Отделочный цех, Склад № 1, Склад № 2, Склад № 3, Итого.

Формулы для расчетов:

$$ОС = БС - ИО$$

$$ВПС = БС * К$$

$$ВОС = ОС * К$$

где  $k$  — коэффициент, равный:

3,3 — если БС меньше либо равен 650 млн руб.;

4,2 — если БС больше 650 млн руб., но меньше 1000 млн руб.;

5,1 — если БС равен 1000 млн руб. или более.

Для формирования автоматических расчетов используйте следующие формулы:

для ячейки **D5:** =B5-C5

для ячейки **E5**: =B5\*ЕСЛИ(B5<=650;3,3;ЕСЛИ(И(B5>650;B5<1000);4,2;5,1))

для ячейки **F5**: =D5\*ЕСЛИ(B5<=650;3,3;ЕСЛИ(И(B5>650;B5<1000);4,2;5,1))

Результирующую строку итогов получить использованием, например, для ячейки **B12** формулы: =СУММ(B5:B11), либо следует выделить диапазон ячеек B12:F12 и воспользоваться возможностью автосуммирования (нажать кнопку Автосумма на панели инструментов).

Отформатировать полученные в таблице результаты, а также название ведомости.

	A	B	C	D	E	F
1	<b>ВЕДОМОСТЬ ПЕРЕОЦЕНКИ ОСНОВНЫХ СРЕДСТВ ПРОИЗВОДСТВА</b>					
2						
3						
4	Наименование объекта	Балансовая стоимость (БС)	Износ объекта (ИО)	Остаточная стоимость (ОС)	Восстановительная полная стоимость (ВПС)	Восстановительная остаточная стоимость (ВОС)
5	Отдел менеджмента и маркетинга	19087,8	568,8	18519,0	97347,8	94446,9
6	Отдел транспортировок	407,2	203,0	204,2	1343,8	673,9
7	Сборочный цех	673,0	198,8	474,2	2826,6	1991,6
8	Отделочный цех	821,6	401,2	420,4	3450,7	1765,7
9	Склад №1	598,7	131,0	467,7	1975,7	1543,4
10	Склад №2	610,0	311,2	298,8	2013,0	986,0
11	Склад №3	756,8	159,5	597,3	3178,6	2508,7
12	<b>Итого</b>	22955,1	1973,5	20981,6	112136,1	103916,2

Рис. 3. Ведомость переоценки основных средств производства

**Задание 3.** Сформировать и заполнить отчетную ведомость работы сети компьютерных клубов по форме, приведенной на рис. 4.

	A	B	C	D	E	F	G	H
1	<b>Ведомость работы сети компьютерных клубов</b>							
2								
3	<b>Клуб</b>	<b>Январь</b>	<b>Февраль</b>	<b>Март</b>	<b>Суммарная выручка</b>	<b>Место</b>	<b>Средняя выручка</b>	<b>Процент</b>
4	Альтаир	345,0	543,9	423,9	1312,8	10,0	437,6	6,07%
5	Грувит	657,7	234,0	982,4	1874,1	7,0	624,7	8,66%
6	Полигон	765,2	1007,5	873,1	2645,8	4,0	881,9	12,22%
7	Гелакс	123,5	734,0	487,7	1345,2	9,0	448,4	6,21%
8	Звезда	879,0	985,9	980,3	2845,2	1,0	948,4	13,14%
9	Хексен	348,0	591,2	678,0	1617,2	8,0	539,1	7,47%
10	Антей	987,0	634,0	1009,4	2630,4	5,0	876,8	12,15%
11	Арсенал	1009,5	793,2	987,9	2790,6	2,0	930,2	12,89%
12	Арена	434,0	934,0	567,0	1935,0	6,0	645,0	8,94%
13	Блиндаж	835,8	879,0	934,0	2648,8	3,0	882,9	12,24%
14	<b>Итого</b>	6384,7	7336,7	7923,7	21645,1		721,50333	100%

Рис. 4. Ведомость работы сети компьютерных клубов

#### Рекомендации по созданию ведомости

В ячейку **A1** ввести название ведомости.

В ячейки **A3:H3** ввести названия полей ведомости: клуб, Январь, Февраль, Март, Суммарная выручка, Место, Средняя выручка, процент. Поле Клуб включает следующие строки: Альтаир, Грувит, Полигон, Гелакс, Звезда, Хексен, Антей, Арсенал, Арена, Блиндаж, Итого.

Основные формулы для вычислений, которые копируются для аналогичных вычислений по строкам, представлены в таблице 1.

#### Формулы для расчета

Ячейка	Формула
--------	---------

<b>E4</b>	=СУММ(B4:O4)
<b>B14</b>	=СУММ(B4:B13)
<b>F4</b>	=РАНГ(E4;\$E\$4:\$E\$13)
<b>G4</b>	=СРЗНАЧ(B4:D4)
<b>G14</b>	=СРЗНАЧ(G4:G13)
<b>H4</b>	=E4/\$E\$14

Отформатировать полученную ведомость.

**Задание 4.** Сформировать на рабочем листе ведомость "Расчет заработной платы работников научно-проектного отдела "Альфа"(рис. 5).

	A	B	C	D	E	F	G	H	I	J	K
1	<b>РАСЧЕТ ЗАРАБОТНОЙ ПЛАТЫ</b>										
2	<b>РАБОТНИКОВ НАУЧНО-ПРОЕКТНОГО ОТДЕЛА "АЛЬФА"</b>										
3	<b>№ п/п</b>	<b>ФИО</b>	<b>Должность</b>	<b>Тарифная ставка</b>	<b>Стаж</b>	<b>К</b>	<b>Надбавка за стаж</b>	<b>Итого</b>	<b>Процент налога</b>	<b>Удержать</b>	<b>выплата</b>
4	1	Вольская А.Д.	лаборант	5 670,00р.	2	0,1	567	6 237,00р.	2%	124,74р.	6 112,3р.
5	2	Еремина П.П.	инженер	8 000,00р.	3	0,1	800	8 800,00р.	10%	880,00р.	7 920,0р.
6	3	Зяц В.Д.	мл.н.сотрудник	7 700,00р.	11	0,25	1925	9 625,00р.	10%	962,50р.	8 662,5р.
7	4	Иванова А.С.	лаборант	5 500,00р.	4	0,1	550	6 050,00р.	2%	121,00р.	5 929,0р.
8	5	Фирсова Н.Л.	ст.н.сотрудник	9 700,00р.	6	0,2	1940	11 640,00р.	20%	2 328,00р.	9 312,0р.
9	6	Козаков Р.Л.	инженер	8 000,00р.	3	0,1	800	8 800,00р.	10%	880,00р.	7 920,0р.
10	7	Усманов В.Д.	инженер	8 000,00р.	8	0,2	1600	9 600,00р.	10%	960,00р.	8 640,0р.
11	8	Щетинина Е.Д.	ст.н.сотрудник	9 700,00р.	1	0,1	970	10 670,00р.	20%	2 134,00р.	8 536,0р.
12	9	Уткина А.Р.	мл.н.сотрудник	7 700,00р.	2	0,1	770	8 470,00р.	10%	847,00р.	7 623,0р.
13	10	Соломина А.А.	зав лаборатории	12 200,00р.	16	0,3	3660	15 860,00р.	20%	3 172,00р.	12 688,0р.
14	11	Радионова Е.Н.	лаборант	5 670,00р.	5	0,1	567	6 237,00р.	2%	124,74р.	6 112,3р.
15	12	Петров А.А.	мл.н.сотрудник	7 700,00р.	11	0,25	1925	9 625,00р.	10%	962,50р.	8 662,5р.
16	13	Сидоров В.В.	ст.н.сотрудник	9 700,00р.	4	0,1	970	10 670,00р.	20%	2 134,00р.	8 536,0р.
17	14	Сорозов К.Б.	лаборант	5 500,00р.	7	0,2	1100	6 600,00р.	2%	132,00р.	6 468,0р.
18	15	Погодин П.П.	ст.н.сотрудник	9 700,00р.	6	0,2	1940	11 640,00р.	20%	2 328,00р.	9 312,0р.
19	<b>Всего к выдаче</b>							<b>140 524,00р.</b>		<b>18 090,48р.</b>	<b>122 433,5р.</b>

Рис. 5 Ведомость по расчету заработной платы

### Рекомендации по созданию ведомости

**В ячейку A2** поместить название ведомости — Расчет заработной платы работников научно-проектного отдела "Альфа", отцентрировать по левому краю (например, командой **Формат | Ячейки |** вкладка **Выравнивание** либо соответствующей кнопкой **По правому краю** (на панели инструментов).

В ячейки **A3:K3** ввести названия полей ведомости: № пп, Фамилия И.О., Должность, Тарифная ставка, Стаж, к, Надбавка за стаж, Итого, Процент налога, Удержать, 25ж.

шапке ведомости — к каждому столбцу — создать скрытые примечания. Примечания создаются командой **Вставка | Примечание** (также можно использовать панель инструментов **Рецензирование**, которая добавляет командой **Вид | Панели инструментов | Рецензирование**):

№ пп — номер работника отдела;

Фамилия и.о. — заносятся все фамилии работающих в научно-проектном отделе;

Должность — занимаемая должность на момент заполнения ведомости;

тарифная ставка— денежный эквивалент занимаемой должности;

Стаж— вносится целое число отработанных лет на момент заполнения ведомости;

к— коэффициент за стаж работы;

надбавка за стаж — денежный эквивалент за стаж работы;

итого— начисление заработной платы с учетом тарифной ставки и стажа работы;

процент налога — определяет процент отчислений в бюджет;

Удержать — денежный эквивалент отчислений в бюджет;

Выплата — сумма, предназначенная к выдаче.

При расчетах в ведомости учитывать следующее:

к, Надбавка за стаж, Итого, Процент налога, Удержать, выплата— вычисляются с помощью соответствующих формул, с использованием автозаполнения или копирования формулы.

Коэффициент к присваивается из следующего расчета:

0,1 — отработано до 5 лет включительно, 0,2— от 5 до 10 лет включительно, 0,25 — от 10 до 15 лет включительно, 0,3 — свыше 15 лет. Формула для ячейки **F4**:

=ЕСЛИ(Е4<=5;0,1;ЕСЛИ(И(Е4>5;Е4<=10);0,2;ЕСЛИ(И(Е4>10; Е4<=15);0,25;0,3)))

надбавка за стаж — денежный эквивалент за стаж работы. Формула для ячейки **G4**:

=D4\*F4

Пользовательский формат числа для ячейки **G4**:

# ##0,00р.;

(вводится командой **Формат | Ячейки** | вкладка **Число**, из списка **Числовые форматы** выбрать **Все форматы** и в поле **Тип** ввести указанный формат).

итого — тарифная ставка с учетом стажа. Формула для ячейки **H4**:

=D4+G4

Пользовательский формат числа для ячейки **H4**:

# ##0,00р.;

Процент налога — учитывает, что: 2% — начисление (по итогу) составляет до 7000 р. включительно, 10%— более 7000 р. до 10 000 р. включительно, 20%— более 10 000 р. до 25 000 р. включительно, 35%— превышающие 25 000 р. Формула для ячейки **I4**:

=ЕСЛИ(Н4<=7000;0,02;ЕСЛИ(И(Н4>7000;Н4<=10000);0,1;ЕСЛИ(И(Н4>10000;Н4<=25000);0,2;0,35)))

Формат числа для ячейки **I4** — Процентный.

Удержать — денежный эквивалент налогов. Формула для ячейки **J4**:

=H4\*14

Пользовательский формат числа для ячейки **J4**:

# ##0,00р.;

Выплата — **сумма К выдаче**: Итого без Удержать.

Требования к столбцу стаж:

Создать пользовательский формат данных, учитывающий стаж работы: до 5 лет — данные представлены желтым цветом, от 5 до 10 — синим, от 10 до 15 — зеленым, свыше 15 — красным.

Воспользоваться командой **Формат | Ячейки** и ввести пользовательский формат для ячейки **E4**:

[Красный]# ##0;

а также использовать команду **Формат | Условное форматирование**.

В случае ввода отрицательного числа лет должно появляться соответствующее окно. Для проверки ввода чисел использовать команду **Данные | Проверка** | вкладка **Сообщение об ошибке**.

Для поля Тарифная ставка — вывести постоянное сообщение: **Тарифная ставка. Будьте внимательны при вводе тарифной ставки** для получения которого использовать команду **Данные | Проверка** | вкладка **Сообщение для ввода**.

В случае ввода отрицательных значений в столбце тарифная ставка появляется соответствующее предупреждение: **Тарифная ставка не может быть отрицательной**, которое формируется через пользовательский формат:

# ##0,00р.;[Красный]"Тарифная ставка не может быть отрицательной!"



## Лабораторная работа Функции даты и времени

**Задание 1.** Определить для работающих сотрудников стаж работы. Результат представлен на рис. 6

	А	В	С	Д	Е
1	<b>Определение стажа работы</b>				
2					
3	<b>Фамилия</b>	<b>дата рождения</b>	<b>Возраст</b>	<b>Начало работы</b>	<b>стаж работы</b>
4	Комаров И.П.	30.03.1975	34	06.09.1994	14,83
5	Игнатов С.Н.	05.07.1970	39	12.07.1988	21,00
6	Сидорова О.Б.	12.09.1960	48	05.12.1980	28,58
7	Леонтьева А.Н.	04.08.1965	43	09.11.1983	25,67

Рис. 6. Определение стажа работы

**Для выполнения задания:**

1. Введите необходимые данные для вычисления стажа работы по столбцам Фамилия, Дата рождения и Начало работы.
2. Формулы для расчета возьмите из табл. 2.
3. Скопируйте формулы в соответствующие ячейки диапазона таблицы.
4. Отформатируйте данные и результаты расчета.

*Таблица 2. Формулы для расчета стажа работы*

Ячейка	Формула	Формат вывода	Описание
<b>С4</b>	=СЕГОДНЯ()-В4	ГГ	Возраст
<b>Е4</b>	=ГОД(СЕГОДНЯ()-D4)-1900+МЕСЯЦ(СЕГОДНЯ()-D4)/12	#0.00	Стаж работы

**Задание 2.** Сформировать ведомость для учета проката видеокассет и CD со следующими графами: № пп, наименование видеокассеты или CD, Фамилия, Дата выдачи, Дата возврата, Срок эксплуатации (в часах), Срок эксплуатации (в днях), Оплата. Произвести необходимые вычисления.

Оплату начислять исходя из следующих положений:

- если срок эксплуатации  $\leq 24$  ч, то ОПЛАТА = тарифная ставка (определить произвольно);
- если срок эксплуатации  $> 24$  ч и  $\leq 48$  ч, то ОПЛАТА = тарифная ставка +  $0,8 \cdot$ тарифная ставка;
- если срок эксплуатации  $> 48$  ч, то за каждый просроченный день взимается 3 тарифных ставки;
- если видеокассета или CD утеряны, то взимается штраф в размере 30 тарифных ставок.

	А	В	С	Д	Е	Ф	Г	Н
1	<b>ВЕДОМОСТЬ ПО УЧЕТУ ПРОКАТА ВИДЕОКАССЕТ И CD</b>							
2								
3	<b>№ п/п</b>	<b>Наименование</b>	<b>Фамилия</b>	<b>Дата выдачи</b>	<b>Дата возврата</b>	<b>Срок эксплуатации (в часах)</b>	<b>Срок эксплуатации (в днях)</b>	<b>Оплата</b>
4	1	Эмили	Вольская А.Д.	21.09.2008	22.09.2008	24	1	5
5	2	Бэмби	Еремина П.П.	01.09.2008	03.09.2008	48	2	9
6	3	Такси	Заяц В.Д.	11.09.2008	15.09.2008	96	4	39
7	4	5 элемент	Иванова А.С.	02.09.2008	22.09.2008	480	20	279
8								
9								
10	<b>Тарифная ставка</b>	<b>5</b>						

Рис. 7. Ведомость по учету проката видеокассет и CD

**Для выполнения задания:**

1. Сформируйте строку заголовка ведомости и внесите необходимые данные в столбцы: № пп, Наименование, Фамилия, Дата выдачи, Дата возврата.
2. В ячейку **С10** введите величину тарифной ставки - 5.
3. В ячейку **Ф4** введите формулу для учета времени проката (в часах):  
 $=\text{ГОД}(\text{Е4}-\text{D4})-1900+\text{МЕСЯЦ}(\text{Е4}-\text{D4})+\text{ДЕНЬ}(\text{Е4}-\text{D4})*24-1$

4. В ячейку **G4** введите формулу для учета времени проката (в днях):  
=ГОД(Е4-D4)-1900+МЕСЯЦ(Е4-D4)+ДЕНЬ(Е4-D4)-1
5. В ячейку **H4** введите формулу для расчета оплаты:  
=ЕСЛИ(F4<=24;C10;ЕСЛИ(И(F4>24;F4<=48);C10+C10\*0,8;ЕСЛИ(F4>48;C10+C10\*0,8+(F4-48)/24\*3\*C10)))
6. Скопируйте формулы в соответствующие однотипные диапазоны — **F5:F7; G5:G7** и **H5:H7**.
7. Отформатируйте все результаты, представленные на рабочем листе.

### **Лабораторная работа Функции просмотра и ссылок**

**Задание 1.** Сформировать ведомость заработной платы, в которой надбавка за стаж определяется исходя из соответствующих надбавок за количество отработанных лет.

**Ход работы:**

1. Сформируйте ведомость и дополнительную таблицу, как показано на рис. 8.

2. В ячейку **D2** введите формулу:

=C2\*ВПР(B2;\$A\$16:\$B\$20;2),

где:

- **B2** — искомое значение (стаж сотрудника), которое необходимо найти в крайнем левом столбце таблицы, определяемой диапазоном \$A\$12:\$B\$16;
- индекс 2 определяет номер столбца таблицы, из которого будет возвращено значение (в данном случае, процент надбавки);
- **C2** — оклад сотрудника.

функции ВПР() и ГПР() используются для поиска информации в прямоугольных таблицах. Данные функции имеют следующий синтаксис:

=ВПР(искомое\_знач; таблица; номер\_столбца; тип\_просмотра);

=ГПР(искомое\_знач; таблица; номер\_строки; тип\_просмотра),

где:

- **искомое\_знач**— это значение, которое необходимо найти в первом столбце (строке) таблицы;

	А	В	С	Д	Е
1	<b>Фамилия И.О.</b>	<b>Стаж</b>	<b>Оклад</b>	<b>Надбавка за стаж</b>	<b>Всего</b>
2	Григорьев Р.П.	5	4500,00		
3	Сидоров В.В.	20	8700,00		
4	Баранова А.С.	15	6200,00		
5	Сухарева Н.Е.	25	5000,00		
6	Иванов Р.П.	10	5430,00		
7	Степанов П.Р.	5	8300,00		
8	Николаева А.А.	10	4800,00		
9					
10					
	<b>Стаж</b>	<b>Надбавка за стаж (%)</b>			
11					
12	5	3			
13	10	5			
14	15	7			
15	20	10			
16	25	15			

Рис. 8. Исходные данные для ведомости зарплаты

- **таблица**— массив или имя диапазона, который определяет таблицу с данными. Первый столбец (строка) должны быть упорядочены в алфавитном порядке текстовых значений или в порядке возрастания числовых данных, а также значений даты/времени;
- **номер\_столбца (номер\_строки)** — указывает, из какого столбца (строки) таблицы следует выбирать возвращаемое значение;

• тип\_просмотра — определяет логическое значение для указания типа соответствия: точное или приближенное. Если аргумент опущен, то используется приближенное сравнение, что означает поиск наибольшего сравниваемого значения, которое меньше или равно искомому значению.

Поиск осуществляется по следующему правилу: Искомое значение сравнивается с данными, расположенными в крайнем левом столбце для функции ВПРО (ИЛИ В верхней строке для функции ГПРО) таблицы, и таким образом определяется первый индекс возвращаемого значения. Номер столбца (номер строки) задает второй индекс, определяющий строку или столбец таблицы, из которой возвращается значение.

3. В ячейку **E2** введите формулу =C2+D2.

4. Скопируйте формулы в соответствующие диапазоны.

5. Отформатируйте таблицу.

Результат для этого примера представлен на рис. 9.

	А	В	С	Д	Е
1	<b>Фамилия И.О.</b>	<b>Стаж</b>	<b>Оклад</b>	<b>Надбавка за стаж</b>	<b>Всего</b>
2	Григорьев Р.П.	5	4500,00	135,00р.	4 635,00р.
3	Сидоров В.В.	20	8700,00	870,00р.	9 570,00р.
4	Баранова А.С.	15	6200,00	434,00р.	6 634,00р.
5	Сухарева Н.Е.	25	5000,00	750,00р.	5 750,00р.
6	Иванов Р.П.	10	5430,00	271,50р.	5 701,50р.
7	Степанов П.Р.	5	8300,00	249,00р.	8 549,00р.
8	Николаева А.А.	10	4800,00	240,00р.	5 040,00р.
9					
10					
11	<b>Стаж</b>	<b>Надбавка за стаж (%)</b>			
12	5	3%			
13	10	5%			
14	15	7%			
15	20	10%			
16	25	15%			

Рис. 9. Ведомость заработной платы

### Лабораторная работа Финансовые функции MS Excel.

Финансовые функции MS Excel предназначены для вычисления базовых величин, необходимых при проведении сложных расчетов.

**Задание 1.** Определить величину основного платежа за четвертый год, если выдана ссуда размером 1 000000000 сроком на 5 лет под 12% годовых.

#### **Решение**

Для основных платежей по займу, который погашается равными платежами в конце или начале каждого расчетного периода, в MS Excel XP используется функция:

ОСПЛТ (Ставка, Период, Кпер, Пс, Бс)

(в более ранних версиях MS Excel эта функция называлась ОСНПЛАТ) В нашем случае функция СППЛТ имеет вид: ОСПЛТ(12%, 4, 5, 1000000000)

Ввод данных и расчеты производятся в соответствии с рис. 4.4.

	А	В
1	<b>Расчет основных платежей по займу</b>	
2		
3	Размер выданной ссуды	1000000
4	Срок ссуды	5
5	Годовая процентная ставка	12%
6	Период выплаты	4
7		
8	Размер выплаты по займу в период выплаты	-221 149,34р.

Рис. 11. Расчет основных платежей по займу

В ячейку **В8** вводится формула:

**ОСПЛТ** (В5;В6;В4;В3)

**Задание 2.** Рассчитать 20-летнюю ипотечную ссуду ей ставкой 10% годовых при начальном взносе 25% и ежемесячной (ежегодной) выплате.

**Решение**

Для вычисления величины постоянной периодической выплаты ренты (например, регулярных платежей по займу) при постоянной процентной ставке используется функция ПЛТ (в ранних версиях — функция ППЛАТ):

ПЛТ(Ставка; Кпер; Бс; Пс; Тип)

В нашем случае функция ПЛТ имеет вид:

ПЛТ(10%/12; 20\*12; -(350000\* (1-25%)))— ежемесячные выплаты;

ПЛТ(10%; 20; -(350000\* (1-25%)))—ежегодные выплаты.

Решение задачи приведено на рис. 12 и 13.

	А	В	С	Д	Е
1	<b>Расчет ипотечной ссуды</b>				
2	<b>Исходные данные</b>				
3	Цена	350 000,00р.			
4	Первый взнос	25%			
5	Годовая процентная ставка	10%			
6	Размер ссуды	262500			
7					
8	Срок погашения ссуды	240	мес.	20	лет
9	<b>Расчетные данные</b>	Выплаты в месяц		Выплаты в год	
10	Периодические выплаты	2 533,18р.		30 833,15р.	
11	Общая сумма выплат	607 963,64р.		616 663,03р.	
12	Общая сумма комиссионных	345 463,64р.		354 163,03р.	
13					

Рис. 12. Расчет ипотечной ссуды

	А	В	С	Д	Е
1	<b>Расчет ипотечной ссуды</b>				
2	<b>Исходные данные</b>				
3	Цена	350000			
4	Первый взнос	0,25			
5	Годовая процентная ставка	0,1			
6	Размер ссуды	=B3*(1-B4)			
7					
8	Срок погашения ссуды	=D8*12	мес.	20	лет
9	<b>Расчетные данные</b>	Выплаты в месяц		Выплаты в год	
10	Периодические выплаты	=ПЛТ(В5/12;В8;-В6)		=ПЛТ(В5;D8;-В6)	
11	Общая сумма выплат	=В8*В10		=D8*D10	
12	Общая сумма комиссионных	=В11-В6		=D11-В6	
13					

Рис. 13. Формулы для расчета ипотечной ссуды

**Задание 3.** Определить, какая сумма окажется на счете, если 52 000 руб. положены на 20 лет под 11% годовых. Проценты начисляются ежемесячно.

**Решение**

Для расчета будущей стоимости единой суммы вклада используются сложные проценты, а расчетная формула основана на функции БС (в ранних версиях — функция БЗ):

БС(Ставка; Кпер; Плт; Пс; Тип)

Для нашей задачи функция БС примет вид:

БС(11%/12;20\*12;,-52000)

Решение задачи приведено на рис. 14, а формула для ячейки **В26**:

=БС(В22/В23;В24\*В23;;-В21)

	А	В
18		
19	<b>Расчет будущей стоимости вклада</b>	
20		
21	Величина начального вклада	52000
22	Годовая процентная ставка	11,00%
23	Количество выплат в год (по процентам)	12
24	Срок вклада (лет)	20
25		
26	Величина будущей стоимости	464 620,80р.
27		

Рис. 14. Расчет будущей стоимости вклада

**Задание 4.** Облигация номиналом 200 000 руб. выпущена на 7 лет. Предусматривается следующий порядок начисления процентов: в первый год— 11%, последующие три года— по 16%, в оставшиеся

три года— по 20%. Рассчитать будущую (наращенную) стоимость облигации по сложной процентной ставке.

**Решение**

Для расчета наращенной стоимости облигации по сложной процентной ставке используется функция:

БЗРАСПИС(Первичное; План)

Для нашей задачи функция принимает вид:

БЗРАСПИС(200000; {11%; 16%; 16%; 16%; 20%; 20%; 20%})

Решение приведено на рис. 15, а формула для расчета в ячейке **В42**:

=БЗРАСПИС(В30;В34:В40)

Аналогичным образом можно использовать встроенные функции MS Excel и для других финансовых расчетов.

	А	В	С
28	<b>Расчет наращенной стоимости</b>		
29			
30	Номинал облигации	200000	
31	Срок	7 лет	
32			
33	Порядок начисления процентов		
34	Первый год	11%	
35	Второй год	16%	
36	Третий год	16%	
37	Четвертый год	16%	
38	Пятый год	20%	
39	Шестой год	20%	
40	Седьмой год	20%	
41			
42	<b>Наращенная стоимость</b>	598 784,68р.	

Рис. 15 Расчет наращенной стоимости облигации по сложной процентной ставке

### Лабораторная работа Подбор параметра

Средство MS Excel **Подбор параметра** позволяет определить значение одной входной ячейки, которое требуется для получения желаемого результата в зависимой ячейке (ячейке результата).

**Задание 1.** Предполагается, что доходы по проекту в течение 5 лет составят: 120 000 000 руб., 200 000 000 руб., 300 000 000 руб., 250 000 000 руб. 320 000 000 руб. Определить первоначальные затраты на проект, чтобы обеспечить скорость оборота 12% .

#### **Решение**

Расчет внутренней скорости оборота инвестиций производится с помощью функции ВСД (в ранних версиях — ВНДОХ): ВСД (Значения; Предположения)

	А	В	С
1	<b>Расчет внутренней скорости оборота инвестиций</b>		
2			
3	Ожидаемые доходы в течении		5 лет
4	Затраты по проекту	-700 000 000,00р.	
5	Первый год	120 000 000,00р.	
6	Второй год	200 000 000,00р.	
7	Третий год	300 000 000,00р.	
8	Четвертый год	250 000 000,00р.	
9	Пятый год	320 000 000,00р.	
10			
11			
12	Внутренняя скорость оборота инвестиций	18%	
13			

Рис. 16 Рабочий лист для определения первоначальных затрат по проекту

Ввод исходных данных производится в соответствии с рис. 16. Первоначально для расчета величина затрат на проект выбирается произвольно (ячейку для этой суммы можно оставить даже пустой) и производятся вычисления.

В ячейку **В12** вводится формула =ВСД(В4:В9).

Далее, используя команду **Сервис | Подбор параметра** (рис. 17), находим величину первоначальных затрат на проект, обеспечивающих скорость оборота инвестиций в 12%.

Результат для этого

примера представлен на рис. 18.

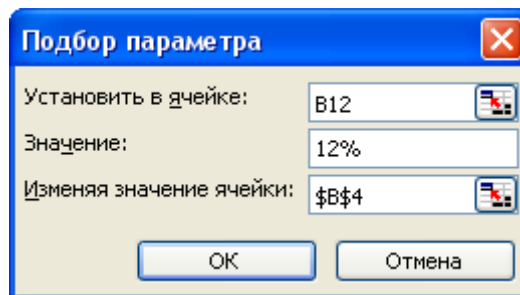


Рис. 17 Окно Подбор параметра

	А	В	С
1	<b>Расчет внутренней скорости оборота инвестиций</b>		
2			
3	Ожидаемые доходы в течении		5 лет
4	Затраты по проекту	-820 389 165,92р.	
5	Первый год	120 000 000,00р.	
6	Второй год	200 000 000,00р.	
7	Третий год	300 000 000,00р.	
8	Четвертый год	250 000 000,00р.	
9	Пятый год	320 000 000,00р.	
10			
11			
12	Внутренняя скорость оборота инвестиций		12%
13			

Рис. 18 Рассчитанная величина первоначальных затрат по проекту

### Пример расчета эффективности неравномерных капиталовложений

**Задание 2.** Вас просят дать в долг 15 000 руб. и обещают вернуть через год 3000 руб., через два — 5000 руб., через три — 9000 руб. При какой процентной ставке эта сделка выгодна?

#### Решение

При решении этой задачи следует использовать функцию ЧПС и средство **Подбор параметра**: ЧПС(Ставка; значение 1; значение 2; . . . )

Ввод исходных данных производится в соответствии с рис. 19. Первоначально для расчета выбирается произвольный процент годовой учетной ставки (ячейку с этой величиной можно оставить даже пустой) и производятся вычисления. В ячейку **В9** вводится формула: =ЧПС(В6;В2:В4)

	А	В	С
1	<b>Неравномерные капиталовложения</b>		
2			
3	<b>Размер ссуды</b>	15 000,00р.	
4	<b>1 год</b>	3 000,00р.	
5	<b>2 год</b>	5 000,00р.	
6	<b>3 год</b>	9 000,00р.	
7	<b>Срок</b>		3 года
8	<b>Годовая учетная ставка</b>		8%
9	<b>Чистый текущий объем вклада</b>	14 208,96р.	

Рис. 19. Рабочий лист для решения задачи с неравномерными капиталовложениями

В ячейку **С7** можно ввести следующую формулу:

=ЕСЛИ(В7=1;"год";ЕСЛИ(И(В7>=2;В7<=4);"года";"лет"))

Далее, выполнив команду **Сервис | Подбор параметра**, в открывшемся диалоговом окне Подбор параметра (рис. 19) задайте исходные данные для нахождения оптимальной процентной ставки:

- в поле **Установить в ячейке** введите **B9**, т. е. адрес ячейки, в которой необходимо получить искомое значение суммы сделки (15000);
  - в поле **Значение** введите 15000, т. е. само искомое значение суммы сделки;
  - в поле **Изменяя значение ячейки** введите адрес ячейки — B8, в которой с помощью средства **Подбор параметра** будет получена необходимая процентная ставка для рассматриваемой задачи (в случае, если такая существует).
- Окончательное решение задачи приведено на рис. 20.

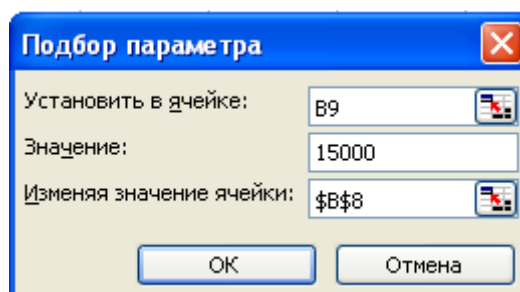


Рис. 19. Окно **Подбор параметра** для задачи о неравномерных капиталовложениях

	A	B	C	D	E	F	G
1	<b>Неравномерные капиталовложения</b>						
2							
3	<b>Размер ссуды</b>	15 000,00р.					
4	<b>1 год</b>	3 000,00р.					
5	<b>2 год</b>	5 000,00р.					
6	<b>3 год</b>	9 000,00р.					
7	<b>Срок</b>		3	года			
8	<b>Годовая учетная ставка</b>		5,50%				
9	<b>Чистый текущий объем вклада</b>	15 000,00р.					

Если банк предлагает Вам годовую процентную ставку больше рассчитанной здесь - ПРЕДЛАГАЕМАЯ СДЕЛКА НЕ ВЫГОДНА!!!

Рис. 20 Оптимальная процентная ставка

### Лабораторная работа Таблица подстановки

*Таблица подстановки* позволяет проводить анализ изменения результата при произвольном диапазоне исходных данных. На одном рабочем листе можно расположить несколько таблиц подстановок. Это дает возможность одновременно анализировать различные формулы и статистические данные.

Таблицу подстановки можно использовать для:

- изменения одного исходного значения, просматривая при этом результаты одной или нескольких формул;
- изменения двух исходных значений, просматривая результаты только одной формулы.

Использование таблицы подстановки с одной изменяющейся переменной и несколькими формулами

Рассмотрим эту методику на примере расчета ежемесячных выплат по займу (расчет происходит с помощью функции ПЛТ) и платежей по процентам (функция ПРОЦПЛАТ):

Решение такой задачи предполагает следующие шаги:

1. Создать или перейти на рабочий лист, где будет решаться анализируемая задача.
2. Организовать интерфейс таким образом, чтобы все вводимые данные были понятны пользователю:

- в соответствующие ячейки рабочего листа вводятся необходимые подписи и данные (рис. 21).
- в ячейку **B5** — формула: =ПЛТ(\$B\$4/12;\$B\$3\*12;\$B\$2)
- в ячейку **D6** — формула: =ПРОЦПЛАТ(\$D\$4;\$D\$5;\$D\$3;\$D\$2)



	А	В	С	Д
1	Расчет ежемесячных выплат по займу		Платежи по процентам	
2	<b>Сумма ссуды</b>	350 000,00р.	<b>Сумма займа на начало года</b>	60 000,00р.
3	<b>Срок погашения</b>	3	<b>Срок погашения</b>	4
4	<b>Процентная ставка</b>	3%	<b>Процентная ставка</b>	3%
5	<b>Выплаты в месяц</b>	-10 178,42р.	<b>Год выплаты</b>	1
6			<b>Сумма выплат</b>	-1 800,00р.
7				

Рис. 21 Подготовка исходных данных

*Совет: При решении задач, связанных с использованием таблицы подстановки, рекомендуется применять в формулах абсолютную адресацию ячеек. Это способствует правильному выполнению вычислений в ячейках рабочего листа.*

3. После подготовки исходных данных перейти к тому месту рабочего листа, где будут располагаться рассчитываемые значения в зависимости от изменения одной переменной и от различных рассчитываемых формул (рис. 22).

В верхней строке будущей таблицы над местом расположения результатов указать адрес каждой формулы, для которой нужно получить список результатов. Вместо адреса можно поместить в ячейку саму формулу (ячейки **В10** и **С10** содержат формулы).

Слева расположить различные значения исходных данных, которые необходимо протестировать.

	А	В	С	Д
1	Расчет ежемесячных выплат по займу		Платежи по процентам	
2	<b>Сумма ссуды</b>	350 000,00р.	<b>Сумма займа на начало года</b>	60 000,00р.
3	<b>Срок погашения</b>	3	<b>Срок погашения</b>	4
4	<b>Процентная ставка</b>	3%	<b>Процентная ставка</b>	3%
5	<b>Выплаты в месяц</b>	-10 178,42р.	<b>Год выплаты</b>	1
6			<b>Сумма выплат</b>	-1 800,00р.
7				
8		<b>Расчет ежемесячных выплат по займу</b>	<b>Платежи по процентам</b>	
9		-10 178,42р.	-1 800,00р.	
10	3%			
11	6%			
12	9%			
13	12%			
14	15%			
15	18%			
16	21%			
17	24%			
18	27%			
19	30%			

Рис. 22 Подготовка изменяемого диапазона и расчетных формул для использования одномерной таблицы подстановки

4. Выделить ячейки, которые будут содержать таблицу. При этом самым левым столбцом этой таблицы должен быть столбец исходных значений, а самой верхней строкой должна быть строка анализируемых формул. Результат будет помещен в пустые ячейки.

5. Воспользоваться командой **Данные | Таблица подстановки** и в диалоговом окне **Таблица подстановки** (рис. 23) указать, куда и какие значения необходимо подставлять.

В нашем примере — подстановка значений процентной ставки (столбец исходных значений **А10:А19**) происходит в ячейку **В4**, т. к. в этой ячейке изначально указывается величина

процентной ставки, входящей в рассчитываемые формулы. Результаты вычислений будут помещены в незаполненные ячейки выделенного диапазона (рис. 24).

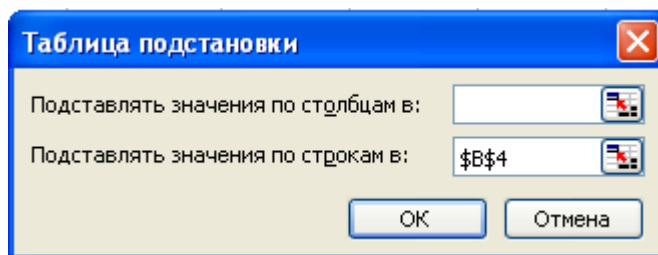


Рис. 23 Использование таблицы подстановки

	А	В	С	Д
1	Расчет ежемесячных выплат по займу		Платежи по процентам	
2	Сумма ссуды	350 000,00р.	Сумма займа на начало года	60 000,00р.
3	Срок погашения	3	Срок погашения	4
4	Процентная ставка	3%	Процентная ставка	3%
5	Выплаты в месяц	-10 178,42р.	Год выплаты	1
6			Сумма выплат	-1 800,00р.
7				
8		Расчет ежемесячных выплат по займу	Платежи по процентам	
9		-10 178,42р.	-1 800,00р.	
10	3%	-10 178,42р.	-1800	
11	6%	-10 647,68р.	-3600	
12	9%	-11 129,91р.	-5400	
13	12%	-11 625,01р.	-7200	
14	15%	-12 132,86р.	-9000	
15	18%	-12 653,34р.	-10800	
16	21%	-13 186,27р.	-12600	
17	24%	-13 731,50р.	-14400	
18	27%	-14 288,83р.	-16200	
19	30%	-14 858,05р.	-18000	

Рис. 24 Рассчитанные значения для одномерной таблицы подстановки

Использование таблицы подстановки с двумя изменяющимися переменными и одной формулой  
Рассмотрим эту методику на примере расчета ежемесячных выплат по займу в зависимости от различных сроков погашения и различных процентных ставок. Решение задачи предполагает следующие шаги:

1. Организовать на рабочем листе соответствующий интерфейс пользователя для некоторого набора входных данных (рис. 25):

- конкретная процентная ставка — 3% (ячейка **В4**);
- конкретный срок погашения — 3 года (ячейка **В3**);
- формула для ячейки **В5**:  
=ПЛТ(\$B\$4/12;\$B\$3\*12;\$B\$2)

	А	В
1	<i>Расчет ежемесячных выплат по займу для различных сроков погашения</i>	
2	Сумма ссуды	350 000,00р.
3	Срок погашения	3
4	Процентная ставка	3%
5	Выплаты в месяц	-10 178,42р.

Рис. 25. Подготовка данных задачи

2. Подготовить следующую таблицу (рис. 26):

- изменяемые данные поместить в левый столбец и верхнюю строку— в нашем случае значения процентной ставки (ячейка **В4**) располагаются в диапазоне **В10:В14**, а значения срока погашения (ячейка **В3**) — в диапазоне **С9:Ф9**;
- на пересечении строки и столбца в верхнем левом углу расположить необходимую формулу или ссылку на нее (ячейка **В9** содержит формулу ячейки **В5**, в которой обязательна абсолютная адресация ячеек).

	А	В	С	Д	Е	Ф
1	<i>Расчет ежемесячных выплат по займу для различных сроков погашения</i>					
2	Сумма ссуды	350 000,00р.				
3	Срок погашения	3				
4	Процентная ставка	3%				
5	Выплаты в месяц	-10 178,42р.				
6						
7						
8			Сроки погашения			
9	Процентная ставка	-10 178,42р.	3	5	10	15
10		3%	-10178,4	-6289,04	-3379,63	-2417,04
11		4%	-10333,4	-6445,78	-3543,58	-2588,91
12		5%	-10489,8	-6604,93	-3712,29	-2767,78
13		6%	-10647,7	-6766,48	-3885,72	-2953,5

Рис. 26. Подготовка диапазона для использования двумерной таблицы подстановки

3. Выделить диапазон ячеек, представляющий собой прямоугольную область и содержащий формулу для расчета, изменяемые данные для двух переменных и ячейки для результатов вычислений.

4. Выполнить команду **Данные | Таблица подстановки** и в появившемся окне (рис. 27) указать, куда и какие значения необходимо подставлять.

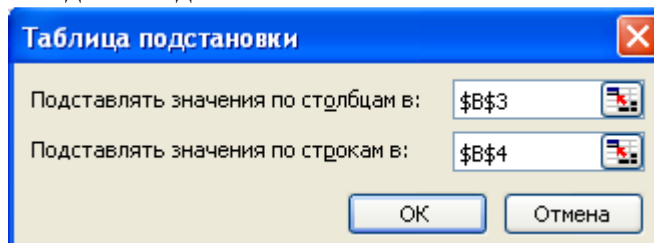


Рис. 27 Использование таблицы подстановки при расчетах по двум параметрам

В рассматриваемом примере подстановка значений процентной ставки (столбец исходных значений **В10:В14**) происходит в ячейку **В4**, т. к. в этой ячейке изначально указывается величина процентной ставки, входящей в рассчитываемую формулу, а подстановка значений сроков

погашения (строка значений **C9:F9**) — в ячейку **B3**. Результаты вычислений будут помещены в незаполненные ячейки выделенного диапазона (28).

	А	В	С	Д	Е	Ф
1	<b>Расчет ежемесячных выплат по займу для различных сроков погашения</b>					
2	<b>Сумма ссуды</b>	350 000,00р.				
3	<b>Срок погашения</b>	3				
4	<b>Процентная ставка</b>	3%				
5	<b>Выплаты в месяц</b>	-10 178,42р.				
6						
7						
8			Сроки погашения			
9	Процентная ставка	-10 178,42р.	3	5	10	15
10		3%	-10178,4	-6289,04	-3379,63	-2417,04
11		4%	-10333,4	-6445,78	-3543,58	-2588,91
12		5%	-10489,8	-6604,93	-3712,29	-2767,78
13		6%	-10647,7	-6766,48	-3885,72	-2953,5

Рис. 28. Рассчитанные данные с использованием двумерной таблицы подстановки

*Замечание:* После построения таблицы подстановки нельзя редактировать отдельно взятую формулу внутри таблицы. Значения данных внутри таблицы можно изменить, меняя значения исходных данных в левом столбце и верхней строке.

### Лабораторная работа Диспетчер сценариев

Во многих экономических задачах результат расчета зависит от нескольких параметров, которыми можно управлять. Часто бывает удобно проанализировать ситуацию для нескольких возможных параметров.

*Диспетчер сценариев* MS Excel позволяет автоматически выполнить анализ "что-если" для различных моделей. Можно создать несколько входных наборов данных (изменяемых ячеек) для любого количества переменных и присвоить имя каждому набору.

По имени выбранного набора данных MS Excel сформирует на рабочем листе результаты анализа. Кроме этого, диспетчер сценариев позволяет создать итоговый отчет по сценариям, в котором отображаются результаты подстановки различных комбинаций входных параметров.

Диспетчер сценариев открывается командой **Сервис | Сценарии** (рис. 29).

В окне диспетчера сценариев с помощью соответствующих кнопок можно добавить новый сценарий, изменить, удалить или вывести существующий, а также— объединить несколько различных сценариев и получить итоговый отчет для существующих сценариев.

#### **Задание 1. Рассчитать внутреннюю скорость оборота инвестиций**

Исходные данные: затраты по проекту составят 700 млн руб. Ожидаемые доходы в течение последующих 5 лет составят, соответственно, 70, 90,300,250, 300 млн руб. Оценить экономическую

целесообразность проекта по скорости оборота инвестиции, если рыночная норма дохода 12%. Рассмотреть также следующие варианты (затраты на проект представлены числом со знаком минус):

- -600; 50;100; 200; 200; 300;
- -650; 90;120;200;250; 250;
- -500, 100,100, 200, 250, 250.

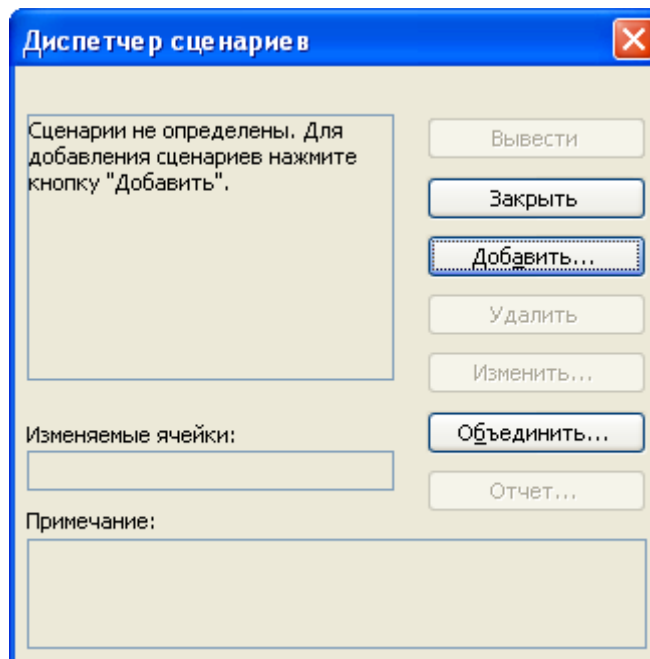


Рис. 29 Окно Диспетчер сценариев

### Решение

Для вычисления внутренней скорости оборота инвестиции (внутренней нормы доходности) используется функция ВСД.:

ВСД (Значения; Предположения)

В нашем случае функция для решения задачи использует только аргумент Значения, один из которых обязательно отрицателен (затраты по проекту). Если внутренняя скорость оборота инвестиций окажется больше рыночной нормы доходности, то проект считается экономически целесообразным. В противном случае проект должен быть отвергнут.

1. Решение приведено на рис. 30. Формулы для расчета:

- в ячейке **B11**: =ВСД(B75:B80)
- в ячейке **C11**: =ЕСЛИ(B84>B82;"Проект экономически целесообразен"; "Проект необходимо отвергнуть")

	А	В	С	Д	Е
1	<b>Расчет внутренней скорости оборота инвестиций</b>				
2					
3	Ожидаемые доходы в течении		5 лет		
4	Затраты по проекту	-700 000 000,00р.			
5	Первый год	70 000 000,00р.			
6	второй год	90 000 000,00р.			
7	Третий год	300 000 000,00р.			
8	Четвертый год	250 000 000,00р.			
9	Пятый год	300 000 000,00р.			
10					
11	Рыночная норма дохода		12%		
12					
13	Внутренняя скорость оборота инвестиций		11%	Проект необходимо отвергнуть	

Рис. 30. Расчет внутренней скорости оборота инвестиций

2. Рассмотрим этот пример для всех комбинаций исходных данных. Для создания сценария следует использовать команду **Сервис | Сценарии | кнопка Добавить** (рис. 31).

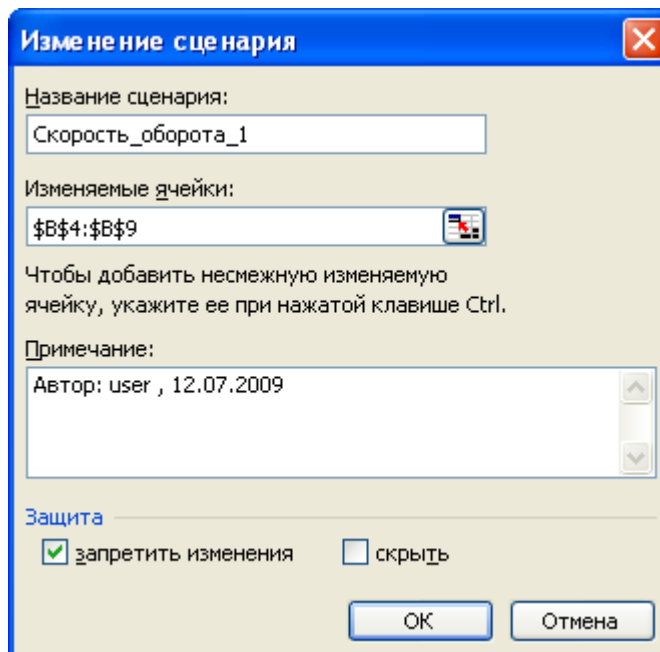


Рис. 31. Добавление сценария для первой комбинации исходных данных

После нажатия на кнопку ОК появляется возможность внесения новых значений для изменяемых ячеек (рис. 32).

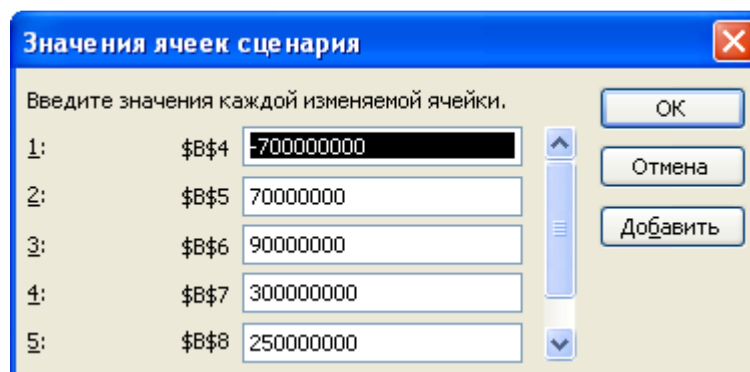


Рис. 32. Окно для изменения значений ячеек сценария

Для сохранения результатов по первому сценарию нет необходимости редактировать значения ячеек— достаточно нажать кнопку ОК для подтверждения значений, появившихся по умолчанию, и выхода в окно **Диспетчер сценариев** (рис. 33).

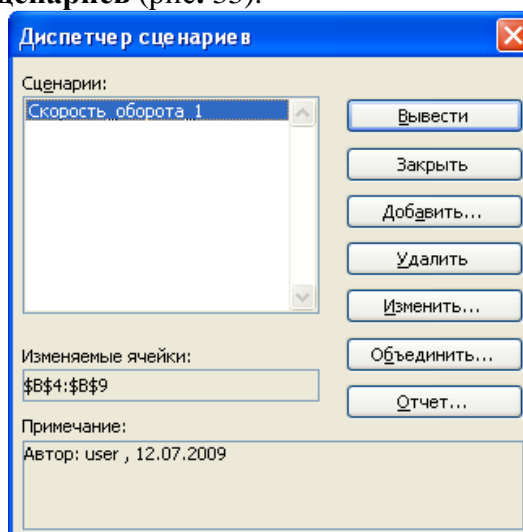


Рис. 33 Окно **Диспетчер сценариев** с первым сохраненным сценарием

3. Для добавления к рассматриваемой задаче новых сценариев достаточно нажать кнопку **Добавить** в окне **Диспетчер сценариев** и повторить вышеописанные действия, изменив значения в ячейках исходных данных (рис. 34).

На рис. 34 сценарий **Скорость\_оборота\_1** соответствует данным (-700; 70; 90; 300; 250; 300), сценарий **Скорость\_оборота\_2** — данным (-600; 50; 100; 200; 200; 300), сценарий **Скорость\_оборота\_3**— данным (-650; 90; 120; 200; 250; 250), сценарий **Скорость\_оборота\_4**— данным (-500, 100, 100, 200, 250,250).

Нажав кнопку **Вывести**, можно просмотреть на рабочем листе результаты расчета для соответствующей комбинации исходных значений.

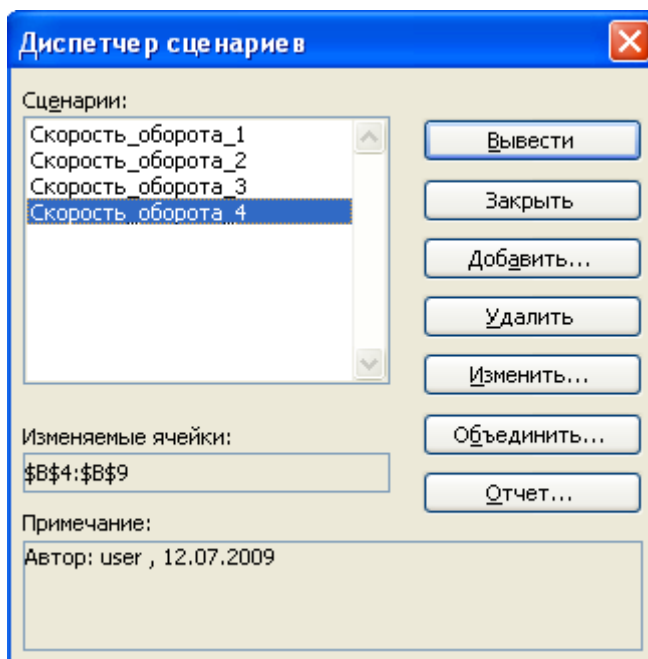


Рис. 34. Окно **Диспетчер сценариев** с добавленными сценариями по расчету скорости оборота инвестиций

	А	В	С	Д	Е	Ф
1	<b>Расчет внутренней скорости оборота инвестиций</b>					
2						
3	Ожидаемые доходы в течении		5 лет			
4	Затраты по проекту	-650 000 000,00р.				
5	Первый год	90 000 000,00р.				
6	второй год	120 000 000,00р.				
7	Третий год	200 000 000,00р.				
8	Четвертый год	250 000 000,00р.				
9	Пятый год	250 000 000,00р.				
10						
11	Рыночная форма дохода		12%			
12						
13	Внутренняя скорость оборота инвестиций		10%	Проект необходимо отвергнуть		

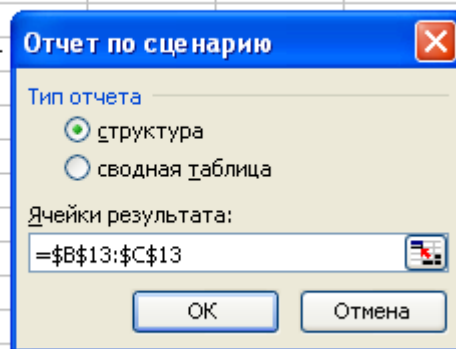


Рис. 4.31. Добавление ячеек результата в окно **Отчет по сценарию**

4. Для получения итогового отчета по всем добавленным сценариям следует нажать кнопку **Отчет** в окне диспетчера сценариев. В появившемся окне **Отчет по сценарию** (рис. 35) выбрать необходимый тип отчета и дать ссылки на ячейки, в которых вычисляются результирующие

функции. При нажатии на кнопку ОК на соответствующий лист рабочей книги выводится отчет по сценариям (рис. 36 и рис. 37)

Структура сценария					
Текущие значения: Скорость_оборота_1 Скорость_оборота_2 Скорость_оборота_3 Скорость_оборота_4					
Изменяемые:					
\$B\$4	-650 000 000,00р.	-700 000 000,00р.	-600 000 000,00р.	-650 000 000,00р.	-500 000 000,00р.
\$B\$5	90 000 000,00р.	70 000 000,00р.	50 000 000,00р.	90 000 000,00р.	100 000 000,00р.
\$B\$6	120 000 000,00р.	90 000 000,00р.	100 000 000,00р.	120 000 000,00р.	100 000 000,00р.
\$B\$7	200 000 000,00р.	300 000 000,00р.	200 000 000,00р.	200 000 000,00р.	200 000 000,00р.
\$B\$8	250 000 000,00р.	250 000 000,00р.	200 000 000,00р.	250 000 000,00р.	250 000 000,00р.
\$B\$9	250 000 000,00р.	300 000 000,00р.	300 000 000,00р.	250 000 000,00р.	250 000 000,00р.
Результат:					
\$B\$13	10%	11%	10%	10%	19%
\$C\$13	необходимо отвергнуть	Проект необходимо отвергнуть	Проект необходимо отвергнуть	Проект необходимо отвергнуть	экономически целесообразен

Примечания: столбец "Текущие значения" представляет значения изменяемых ячеек в момент создания Итогового отчета по Сценарию. Изменяемые ячейки для каждого сценария выделены серым цветом.

Рис. 36 Отчет типа **Структура** по сценариям расчета скорости оборота инвестиций

	А	В	С
1	\$B\$4:\$B\$9 на	(Все)	
2			
3		Ячейки результата	
4	\$B\$4:\$B\$9	\$B\$13	\$C\$13
5	Скорость_оборота_1	0,109167533	1
6	Скорость_оборота_2	0,100639141	1
7	Скорость_оборота_3	0,103664967	1
8	Скорость_оборота_4	0,192058425	1

Рис. 37 Отчет типа **Сводная таблица** по сценариям расчета скорости оборота инвестиций

### Лабораторная работа Диаграммы и графики

Диаграммы MS Excel дают возможность графического представления различных числовых данных.

MS Excel предлагает различные типы диаграмм и предусматривает широкий спектр возможностей для их изменения. Для создания диаграмм в MS Excel прежде всего следует подготовить данные для построения диаграммы и определить ее тип.

При этом необходимо учитывать следующее:

- MS Excel предполагает, что количество рядов данных (Y) должно быть меньше, чем категорий (X). Исходя из этого, определяется расположение рядов (в строках или столбцах), а также — снабжены ли ряды и категории именами.
- Если диаграмма строится для диапазона ячеек, имеющего больше столбцов, чем строк, или равное их число, то рядами данных считаются строки.
- Если диапазон ячеек имеет больше строк, то рядами данных считаются столбцы.
- MS Excel предполагает, что названия, связанные с рядами данных, считаются их именами и составляют *легенду* диаграммы. Данные, интерпретируемые как категории, считаются названиями категорий и выводятся вдоль оси X.
- Если в ячейках, которые MS Excel будет использовать как названия категорий, содержатся числа (не текст и не даты), то MS Excel предполагает, что в этих ячейках содержится ряд данных, и строит диаграмму без меток на оси категорий (X), вместо этого нумеруя категории.
- Если в ячейках, которые MS Excel намерен использовать как названия рядов, содержатся числа (не текст и не даты), то MS Excel предполагает, что в этих ячейках содержатся первые точки рядов данных, а в каждом ряду данных присваивается имя **Ряд1**, **Ряд 2** и т.д.



**Задание 1.** Построить диаграмму по результатам годовых поставок товара в период с 1996 года по 2003 год.


**Решение**

Для выполнения задания:

1. Введите данные на рабочий лист в соответствии с рис. 38.

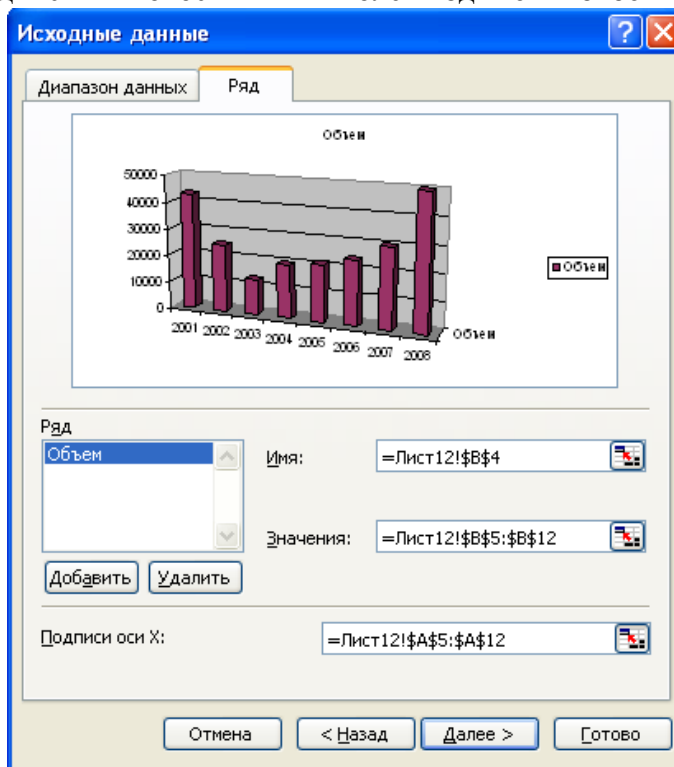
	А	В
1		
2	<b>Выполнение плана поставок</b>	
3		
4	Год	Объем
5	2001	43000
6	2002	25000
7	2003	13000
8	2004	20000
9	2005	21300
10	2006	24000
11	2007	30000
12	2008	50000

Рис.38 Подготовка данных для построения диаграммы

2. Выделите мышью диапазон **A5:B12** и выполните команду **Вставка | Диаграмма** либо нажмите кнопку мастера диаграмм  на панели инструментов **Стандартная**.

3. Далее работа с мастером диаграмм осуществляется в пошаговом режиме:

- Выбор вида диаграммы — с учетом того, что в примере предложены дискретные статистические данные, подходящий вид диаграммы — гистораммного типа.
- Нажав кнопку **Далее**, переходим к шагу 2. На вкладке **Ряд** укажите:
- какой ряд выбран в качестве значения данных (заполните поля **Ряд** и **Значения**). В нашем случае в поле **Ряд** вносим объем и в поле **Значения** — диапазон **\$B\$5: \$B\$12**;
- в какой ряд будет служить подписями по оси X— в поле **Подписи по оси X** вносим год и



указываем диапазон **\$A\$5:\$A\$12**.

Рис. 39. Определение рядов данных

4. Отформатируйте полученную диаграмму, используя контекстное меню каждого ее элемента (рис. 40).



Рис. 40. Построенная диаграмма Объем продаж

**Задание 2.** Построить график функции:  $y = \cos^3(\pi x)$ .

**Решение**

Результат для этого примера представлен на рис. 41.

Для выполнения задания:

1. Задайте область определения  $X$  вводом начальных данных: 0 и 0,1, а затем маркером автозаполнения подготовьте весь диапазон **A7:A27**.
2. В ячейку **B7** введите формулу:  $=(\text{КОЗ}(\text{ПИ}()*\text{A7}))^3$  и скопируйте ее на диапазон **B7:B27**.
3. Постройте график функции с помощью мастера диаграмм.
4. Отформатируйте полученный график.

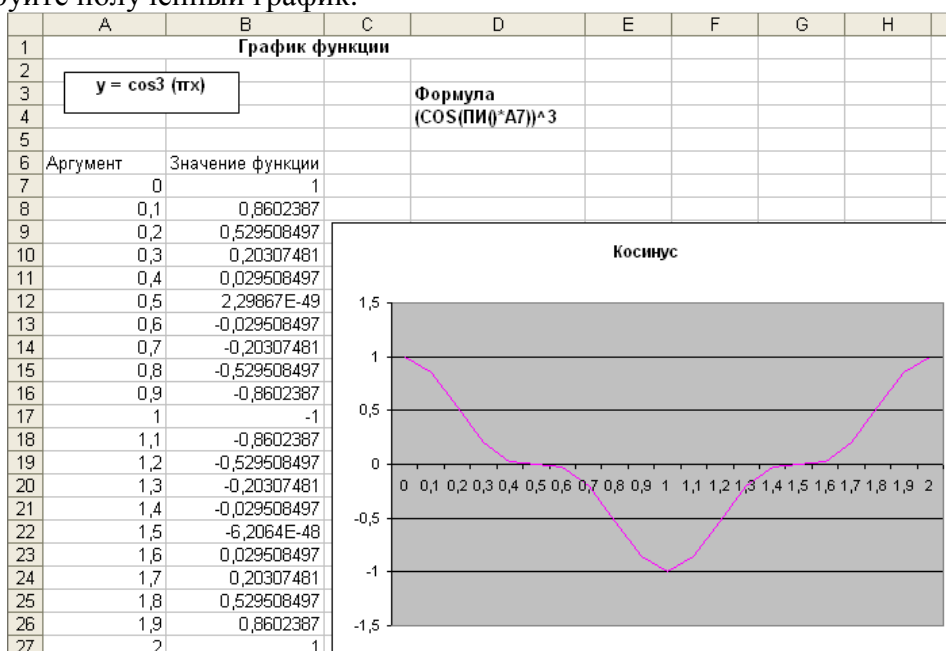


Рис. 41 Пример построения графика функции

## Лабораторная работа Расширенный фильтр MS Excel

**Задание 1.** Определить, имеются ли в предложенном списке (см. рис. 42) белые машины, год выпуска которых больше 2000 и цена которых находится в диапазоне от 3000 до 16 000 у. е., или черные Мерседесы, пробег которых более 23 000 км, но менее 150 000 км.

	A	B	C	D	E	F	G	H	I	J
1	Цифры номера	Буквы номера	Марка машины	Год вып.	Год приобретения	Цвет	Пробег	Цена у.е.	Техосмотр	Владелец
2	00-02	хр	мерседес	1998	2001	Белый	3000	16000	да	Иванов Б.Б.
3	00-02	сс	мерседес	1997	2000	Черный	1000	25000	да	Григорьев Р.П.
4	00-04	хр	Пижо	1996	1999	Красный	50000	6000	да	Сидоров В.В.
5	00-05	хх	мерседес	1998	1999	Синий	23000	18000	да	Баранова А.С.
6	00-05	са	Ауди	1998	2000	Бежевый	34000	21000	нет	Сухарева Н.Е.
7	00-05	сі	Ауди	1999	2001	Черный	65000	6800	да	Иванов Р.П.
8	00-06	сі	Мазда	2001	2001	Белый	96000	8000	да	Степанов П.Р.
9	00-09	сп	Мазда	2006	2006	Синий	120000	12000	нет	Николаева А.А.
10	00-12	сі	Форд	2005	2006	Красный	6000	11000	да	Федоров А.А.
11	00-23	са	Форд	2007	2008	Красный	42000	11500	да	Боодаренко Р.Т.
12	00-23	сп	Рено	2003	2005	Синий	8000	14000	нет	Кузмин Л.Р.
13	00-25	сс	Пижо	1999	2001	Черный	64000	9000	нет	Королев К.К.
14	00-32	са	мерседес	2007	2009	Черный	85000	9700	да	Сидоренко Т.А.
15	00-32	нг	Пижо	2005	2006	Белый	79000	16000	да	Елисеев Д.О.
16	00-34	са	Рено	2000	2003	Бежевый	80000	13000	да	Гончарова М.А.
17	00-36	сс	БМВ	2001	2002	Бежевый	90000	8000	да	Грибков А.Л.
18	00-45	нг	БМВ	2002	2002	Синий	94000	8500	нет	Горохов П.П.
19	00-45	сс	БМВ	2006	2007	Синий	136000	17000	да	Горохов А.С.

Рис. 42 Список MS Excel

### Решение

- Откройте список, подлежащий фильтрации (список располагается в диапазоне **A1:J35**, строка заголовка — в диапазоне **A1:J1**)
- Сформируйте диапазон критериев для расширенного фильтра в соответствии с рис. 43.

J	K	L	M	N	O	P	Q	R
Степанов П.Р.								
Николаева А.А.								
Федоров А.А.		<b>Марка машины</b>	<b>Год выпуска</b>	<b>Цвет</b>	<b>Пробег</b>	<b>Пробег</b>	<b>Цена, у.е.</b>	<b>Цена, у.е.</b>
Боодаренко Р.Т.			>2000	Белый			>=3000	<=16000
Кузмин Л.Р.		Мерседес		Черный	>23000	<150000		
Королев К.К.								
Сидоренко Т.А.								

Рис. 43. Диапазон критериев для расширенного фильтра к задаче про белые и черные машины

- Выполните команду **Данные | Фильтр | Расширенный фильтр**.
- Отфильтрованные данные приведены на рис. 44.

	A	B	C	D	E	F	G	H	I	J
1	Цифры номера	Буквы номера	Марка машины	Год выпуска	Год приобретения	Цвет	Пробег	Цена у.е.	Техосмотр	Владелец
8	00-06	сі	Мазда	2001	2001	Белый	96000	8000	да	Степанов П.Р.
14	00-32	са	мерседес	2007	2009	Черный	85000	9700	да	Сидоренко Т.А.
15	00-32	нг	Пижо	2005	2006	Белый	79000	16000	да	Елисеев Д.О.
25	19-25	нг	Ауди	2003	2004	Белый	64000	9600	да	Бочаров Е.Е.
30	19-30	нг	мерседес	2004	2005	Белый	61000	10200	да	Зудилов А.В.

Рис. 44. Данные к задаче про белые и черные машины, отображенные расширенным фильтром

**Задание 2.** Определить, имеются ли в списке (см. рис. 42) машины, год выпуска которых больше 2000 и пробег которых более 100 00 км, но менее 100 000 км, или черные Мерседесы, цена которых более 20 000 у. е., но менее 30 000 у. е.

**Решение**

1. Откройте список, подлежащий фильтрации (список располагается в диапазоне **A1:J35**, строка заголовка— в диапазоне **A1:J1**)
2. Сформируйте вычисляемый критерий для расширенного фильтра в диапазоне **M3:M4**. В ячейку **M3** введите Условие. В ячейку **M4** введите формулу:  
 $=ИЛИ(И(G2>10000;G2<100000;D2>1990);И(C2="Мерседес";F2="Черный";H2>20000;H2<30000))$
3. Выполните команду **Данные | Фильтр | Расширенный фильтр**.
4. Отфильтрованные данные представлены на рис. 45.

	А	В	С	Д	Е	Ф	Г	Н	І	Ј
1	Цифры номера	Буквы номера	Марка машины	Год выпуска	Год приобретения	Цвет	Пробег	Цена у.е.	Техосмотр	Владелец
3	00-02	сс	Мерседес	1997	2000	Черный	1000	25000	да	Григорьев Р.П.
8	00-06	сі	Мазда	2001	2001	Белый	96000	8000	да	Степанов П.Р.
11	00-23	са	Форд	2007	2008	Красный	42000	11500	да	Боодаренко Р.Т.
14	00-32	са	Мерседес	2007	2009	Черный	85000	9700	да	Сидоренко Т.А.
15	00-32	ng	Пижо	2005	2006	Белый	79000	16000	да	Елисеев Д.О.
17	00-36	сс	БМВ	2001	2002	Бежевый	90000	8000	да	Грибков А.Л.
18	00-45	ng	БМВ	2002	2002	Синий	94000	8500	нет	Горохов П.П.
23	19-23	сі	Волга	2003	2003	Бежевый	86000	2200	нет	Косолапов А.С.
24	19-24	ng	БМВ	2004	2005	Черный	92000	14000	да	Следкова Т.П.
25	19-25	ng	Ауди	2003	2004	Белый	64000	9600	да	Бочаров Е.Е.
26	19-26	хр	Фольксваген	2006	2006	Красный	94000	9500	да	Королева А.И.
30	19-30	ng	Мерседес	2004	2005	Белый	61000	10200	да	Зудилов А.В.

Рис. 45. Данные к задаче о пробеге, отображенные расширенным фильтром

**Задание 3.** Определить автомобили белого или красного цвета, цена которых меньше средней цены для всех автомобилей и пробег которых больше либо равен среднему пробегу для всех автомобилей (см. рис. 42).

**Решение**

1. Откройте список, подлежащий фильтрации (список располагается в диапазоне **A1:J35**, строка заголовка— в диапазоне **A1:J1**)
2. Сформируйте вычисляемый критерий для расширенного фильтра в диапазоне **L1:L2**. В ячейку **L1** введите Условие. В ячейку **L2** введите формулу:  
 $=И(ИЛИ(Г2="белый";Г2="красный");H2<CPЗНАЧ($H$2:$H$133); G2 >=CPЗНАЧ($G$ 2:$G$13 3))$
3. Выполните команду **Данные | Фильтр | Расширенный фильтр**.

**Лабораторная работа Автоматическое подведение итогов в MS Excel**

Итоги необходимы для создания разнообразных отчетов и для обобщения большого количества однотипной информации. Итоги подразделяются на:

- простые промежуточные;
- сложные промежуточные;
- связанные с вычислением частичных сумм (используется мастер *частичных сумм*).

**Задание 1.** Дан список со следующими полями (рис. 46): **№ пп, Продавец, Товар, Номер партии, Цена, Количество, Итого, Дата продажи, Покупатель**. Определить количество товаров, проданных конкретным продавцом за конкретную дату.

## Решение

1. Выделите список (или — установите в список указатель ячейки) и проведите сортировку (команда **Данные | Сортировка**) сначала — по полю **Продавец**, затем — по полю **Дата продажи** (рис. 47).

	A	B	C	D	E	F	G	H	I
1	№ п/п	Продавец	Товар	Номер партии	Цена	Количество	Итого	Дата продажи	Покупатель
2	1	Сидоров	дискета	2	540	40	21600	12.05.2003	Голец
3	2	Сидоров	маркер	2	400	23	9200	04.08.2001	Голец
4	3	Белый	ручка	2	140	56	7840	23.11.2005	Жемчугов
5	4	Кремлев	пенал	3	1250	3	3750	04.05.2007	Жемчугов
6	5	Кремлев	карандаш	3	80	23	1840	04.12.2006	Жемчугов
7	6	Петров	ластик	3	80	5	400	07.03.2008	Жемчугов
8	7	Хлебниковх	маркер	3	400	7	2800	25.04.2005	Жемчугов
9	8	Белый	дискета	5	540	34	18360	21.01.2006	Жемчугов
10	9	Кремлев	ручка	1	140	12	1680	08.10.2004	Задворный
11	10	Кремлев	пенал	3	1250	7	8750	31.12.2005	Задворный
12	11	Хлебниковх	бумага	2	2000	5	10000	08.11.2003	Задворный
13	12	Хлебниковх	карандаш	2	210	87	18270	24.06.2006	Задворный
14	13	Белый	ластик	2	80	2	160	18.05.2008	Задворный
15	14	Кремлев	ластик	5	80	4	320	13.09.2007	Климова
16	15	Кремлев	пенал	5	400	5	2000	11.01.2004	Климова
17	16	Кремлев	дискета	4	540	17	9180	19.12.2001	Климова
18	17	Кремлев	ручка	3	140	1	140	23.17.2006	Климова
19	18	Кремлев	бумага	5	2000	8	16000	04.08.2006	Климова
20	19	Сидоров	карандаш	1	210	76	15960	10.10.2002	Климова
21	20	Хлебниковх	маркер	5	400	65	26000	19.12.2008	Климова
22	21	Иванов	дискета	1	540	10	5400	01.05.2006	Задворный
23	22	Иванов	пенал	4	1250	5	6250	19.06.2004	Задворный

Рис. 46. Список продаж

2. Примените команду **Данные | Итоги**. В окне **Промежуточные итоги** установите параметры в соответствии с рис. 48: для получения верхнего (первого) уровня итогов — общее количество товаров, проданных конкретным продавцом.

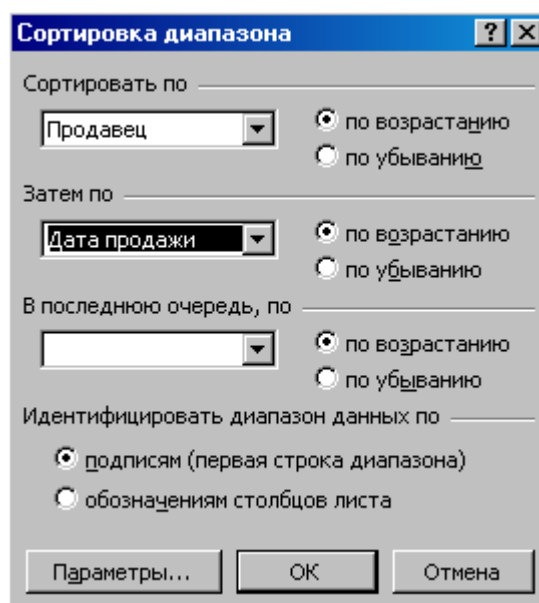


Рис. 47. Сортировка списка

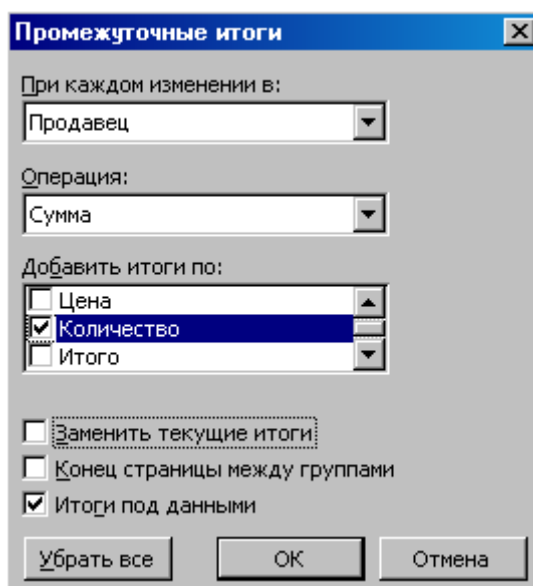


Рис. 48. Окно **Промежуточные итоги** для получения итогов по полю **Продавец**

3. Для получения второго уровня итогов поместите указатель ячейки в список с полученными итогами, затем выполните команду **Данные | Итоги**, установив в окне **Промежуточные итоги** параметры в соответствии с рис. 49.

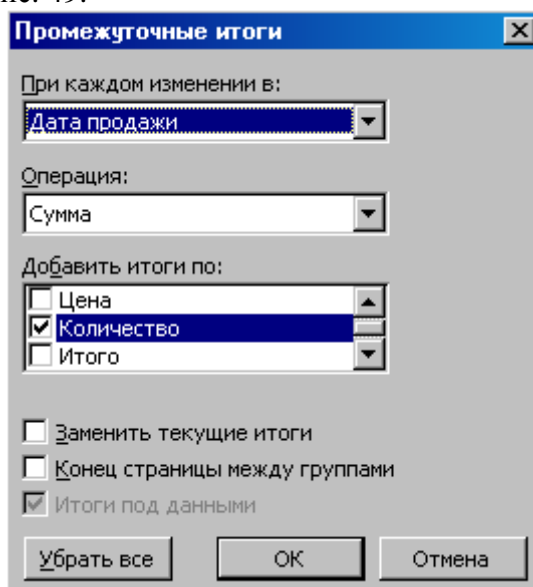


Рис. 49. Окно **Промежуточные итоги** для получения итогов по полю **Дата продажи**

4. Полученные промежуточные итоги представлены на рис. 50.

1	2	3	4	A	B	C	D	E	F	G	H	I
		№ п/п		Продавец	Товар	Номер партии	Цена	Количество	Итого	Дата продажи	Покупатель	
		1										
		2	3	Белый	ручка	2	140	56	7840	23.11.2005	Жемчугов	
		3						56		<b>23.11.2005 Итого</b>		0
		4	8	Белый	дискета	5	540	34	18360	21.01.2006	Жемчугов	
		5						34		<b>21.01.2006 Итого</b>		0
		6	13	Белый	ластик	2	80	2	160	18.05.2008	Задворный	
		7						2		<b>18.05.2008 Итого</b>		0
		8		<b>Белый Итого</b>					92			0
		9	22	Иванов	пенал	4	1250	5	6250	19.06.2004	Задворный	
		10						5		<b>19.06.2004 Итого</b>		0
		11	21	Иванов	дискета	1	540	10	5400	01.05.2006	Задворный	
		12						10		<b>01.05.2006 Итого</b>		0

Рис. 50. Вложенные промежуточные итоги

### Лабораторная работа Консолидация данных

Консолидация предназначается для обобщения однородных данных. Ее осуществление предполагает использование следующей методики:

1. Указать местоположение будущих консолидированных данных.
2. Выбрать команду **Данные | Консолидация**.
3. В открывшемся окне указать диапазоны данных, подлежащие консолидации.
4. Указать способ консолидации:
  - согласно расположению в диапазоне — сняты все флажки области **Использовать в качестве имен**;
  - согласно заголовкам строк и столбцов — установлены флажки **подписи верхней строки** и **значения левого столбца**.
5. Выбрать тип консолидации, т. е. указать, какая операция будет проводиться с консолидируемыми данными.
6. При необходимости указать добавление структуры — установить флажок **Создавать связи с исходными данными**.

**Задание 1.** Объединить данные о количестве и стоимости проданных товаров в сети магазинов, которые представлены в виде списка со следующими полями (рис. 51): **Товар**, **Стоимость**, **Количество**, расположены на листе 2, листе 4 и листе 5.

**Решение:** Следуя рекомендациям методики проведения консолидации, заполните окно

	А	В	С
1	<b>Товар</b>	<b>Стоимость</b>	<b>Количество</b>
2	Дискета	540	10
3	Маркер	400	23
4	Ручка	140	34
5	пенал	1250	5
6	карандаш	80	23
7	ластик	80	2
8	бумага	2000	2
9	Дискета	540	17
10	Маркер	400	7
11	Ручка	140	12
12	пенал	1250	4
13	карандаш	210	76
14	ластик	80	5
15	бумага	2000	5
16	Дискета	540	13
17	Маркер	400	65
18	Ручка	140	56
19	пенал	1250	7
20	карандаш	210	34
21	ластик	80	4

Рис. 51. Данные о реализованных товарах

**Консолидация** в соответствии с данными рис. 52. Объединенные данные представлены на рис. 53.  
**Сводные таблицы**

Сводные таблицы представляют собой средство для группировки, обобщения и анализа данных, находящихся в списках MS Excel или в таблицах, созданных в других приложениях. Внешне сводные

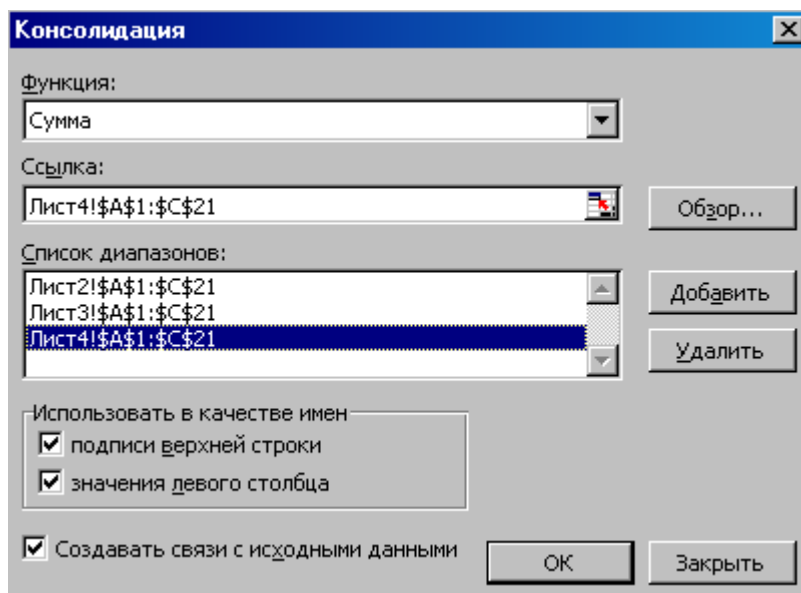


Рис. 52. Ввод данных в окно **Консолидация**

1	2	A	B	C	D
	1			Стоимость	Количество
+	11	Дискета		4860	120
•	12		Книга1	400	23
•	13			400	7
•	14			400	65
•	15	Книга1		400	23
•	16			400	7
•	17			400	65
•	18	Книга1		400	23
•	19			400	7
•	20			400	65
-	21	Маркер		3600	285
+	31	Ручка		1260	306
+	41	пенал		11250	48
+	51	карандаш		1500	399
+	61	ластик		720	33
+	68	бумага		12000	21

Рис. 53. Представление консолидированных данных

### Лабораторная работа Сводные таблицы

Сводные таблицы представляют собой средство для группировки, обобщения и анализа данных, находящихся в списках MS Excel или таблицах, связанных в других приложениях. Внешние сводные таблицы являются структурой, позволяющей размещать данные в трехмерном виде. Сводные таблицы могут использоваться:

- для обобщения большого количества однотипных данных;
- для реорганизации данных (с помощью перетаскивания);
- для отбора и группировки данных;
- для построения диаграмм.

Сводные таблицы создаются с помощью мастера сводных таблиц (команда **Данные | Сводная таблица**) по следующей методике:

1. Выбрать место для сводной таблицы, т. е. установить указатель ячейки в необходимое место на рабочем листе.
2. Выполнить команду **Данные | Сводная таблица**.



3. Задать исходный диапазон данных, выполнив шаги 1 и 2 мастера (рис. 54 и 55). После нажатия кнопки Далее в окне мастера, приведенном на рис. 55, откроется окно 3-го шага мастера (рис. 56).
4. Прежде чем указать местоположение будущей таблицы (рис. 56), необходимо нажать кнопку **Макет** и в открывшемся окне (рис. 57) сформировать макет сводной таблицы (т. е. задать страницу, строки, столбцы, итоговые и вычисляемые поля сводной таблицы).

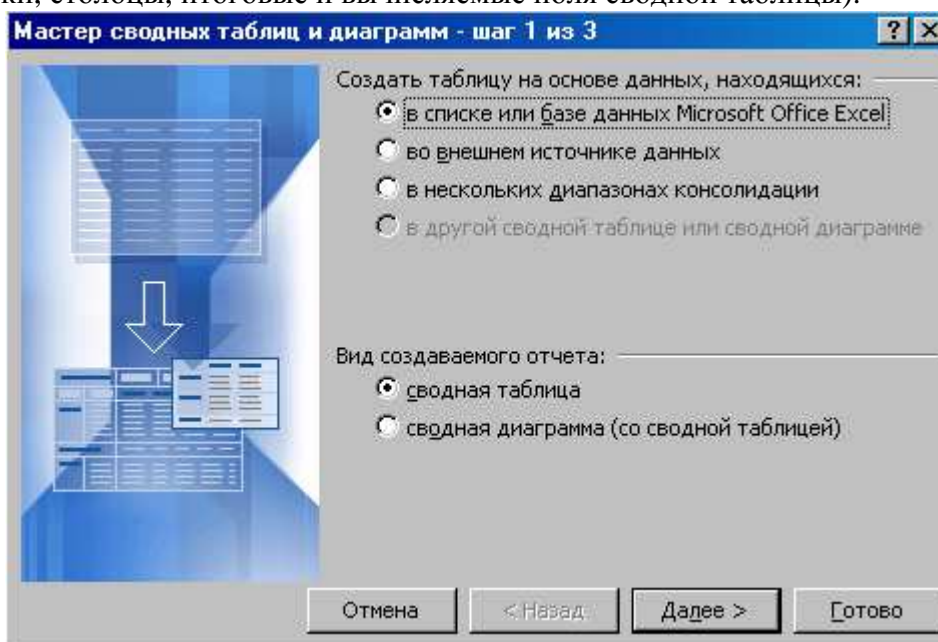


Рис. 54 Определение местоположения данных для сводной таблицы

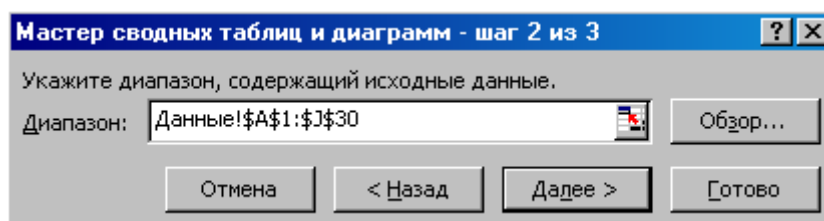


Рис. 55. Диапазон данных для сводной таблицы

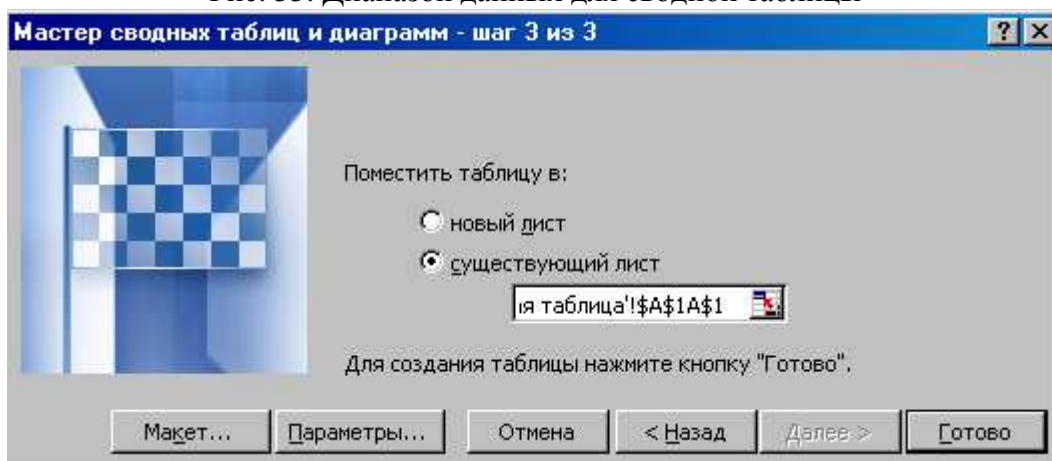


Рис. 56. Указание местоположения будущей сводной таблицы

5. Для определения необходимой операции для полей, помещенных в область **Данные**, либо задания вычисляемого поля дважды щелкнуть левой кнопкой мыши на поле, помещенном в область **Данные** (рис. 57), и выбрать необходимые действия в окне **Вычисление поля сводной таблицы** (рис. 58).
6. Нажать кнопку **Параметры** (рис. 56) и в открывшемся окне (рис. 59) установить необходимые параметры сводной таблицы.

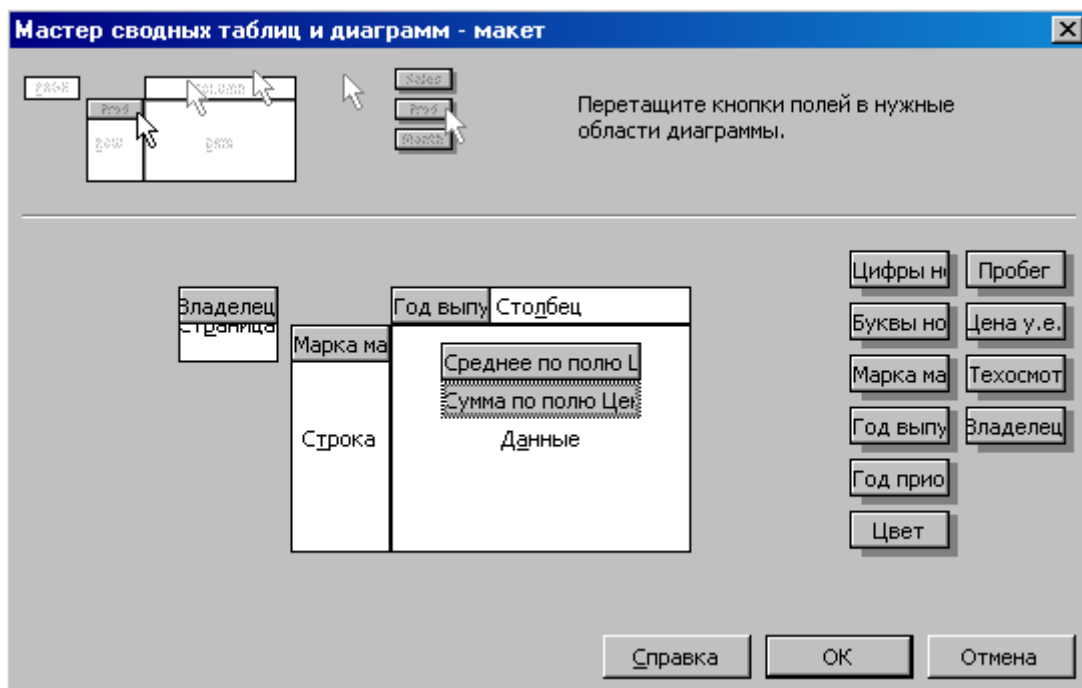


Рис. 57. Формирование макета сводной таблицы

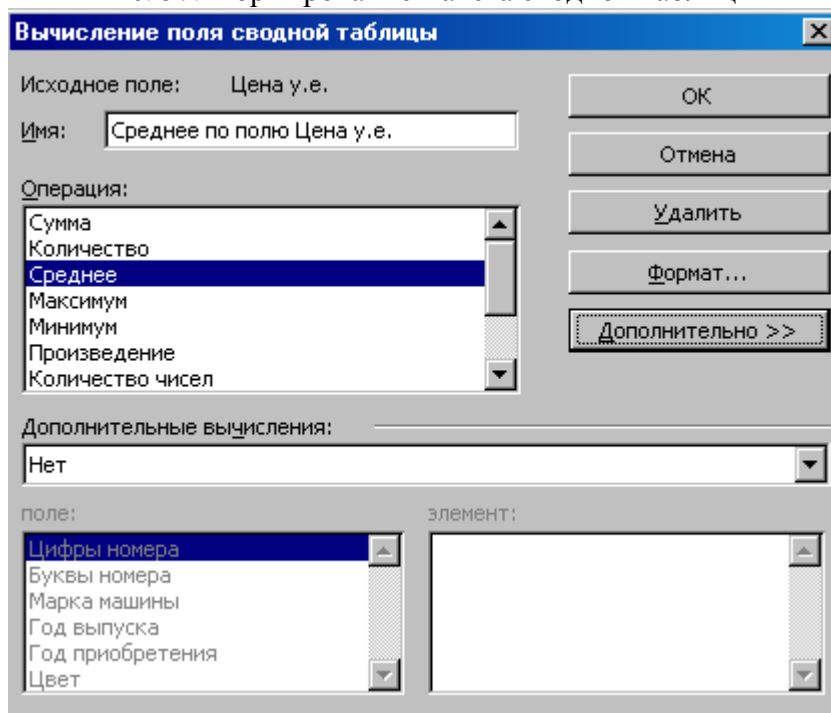


Рис. 58. Окно Вычисление поля сводной таблицы

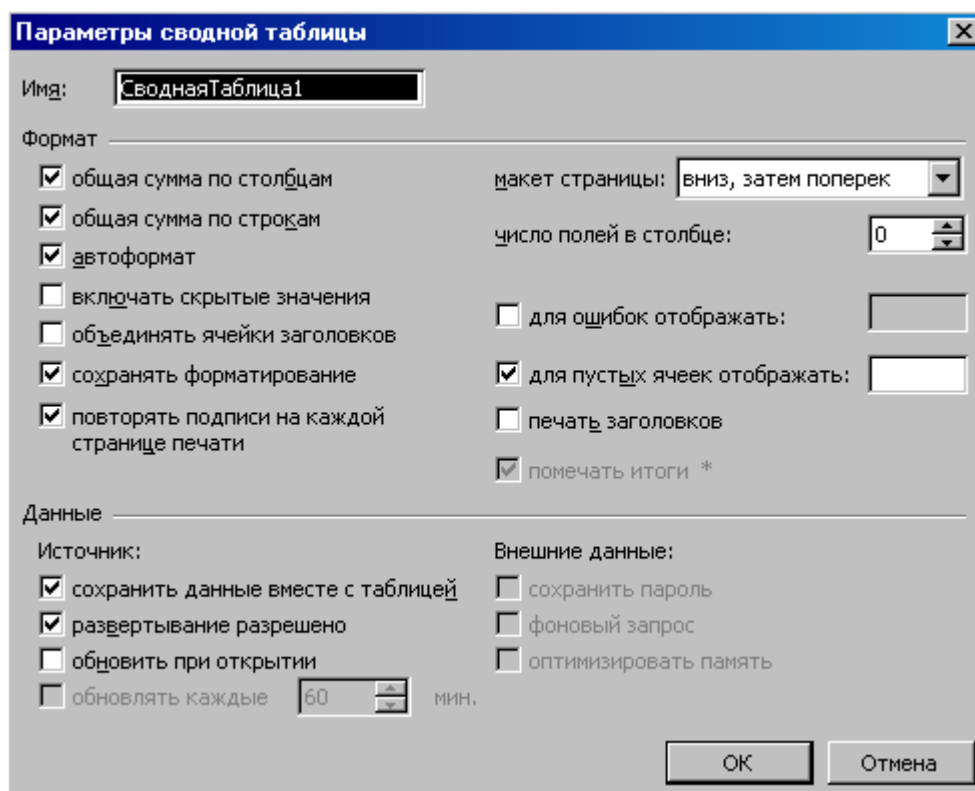


Рис. 59. Установка параметров сводной таблицы

7. После проведения всех подготовительных операций нажать кнопку **Готово** (рис. 56). При создании, редактировании и работе со сводными таблицами необходимо учитывать следующее:

- местоположение сводной таблицы — желательно располагать ее на отдельном листе, т. к. при обновлении, группировках сводной таблицы информация, заключающаяся на рабочих листах рядом со сводной таблицей, может оказаться скрытой;
- местонахождение исходных данных – список MS Excel, внешний источник, диапазоны консолидации, находящиеся в другой сводной таблице;
- необходимость при создании структуры сводной таблицы определить:
  - 1) поля, находящиеся в строках и столбцах таблицы;
  - 2) поля, по которым подводятся итоги (с выбором необходимой операции);
  - 3) поля для страниц, что позволяет представить информацию в трехмерном виде.
- сводная таблица – это средство только для отображения данных. Поэтому в самой таблице данные редактировать нельзя. Для изменения данных в сводной таблице необходимо внести изменения в источник данных, а затем обновить сводную (кнопкой **Обновить данные** на панели инструментов **Сводные таблицы** (рис. 60);

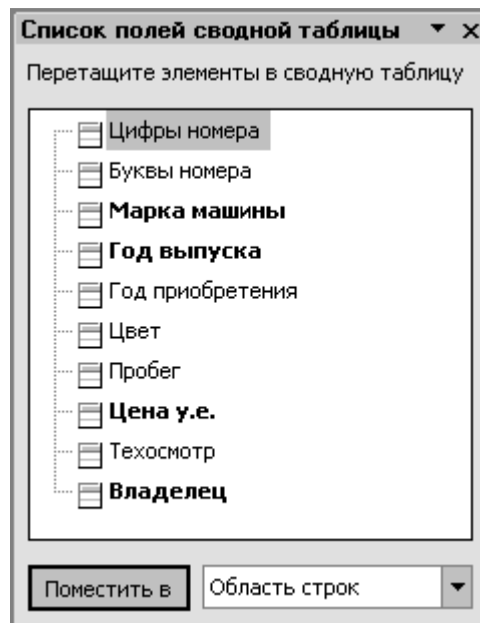


Рис. 60. Панель инструментов **Сводные таблицы**

- в сводных таблицах можно изменять названия полей, что не влечет изменений в полях исходных данных. Манипулирование элементами сводной таблицы можно также осуществлять мышью.
- сводные таблицы допускают возможность группировки элементов полей по различным уровням иерархии путем объединения в группы. Для этой цели в меню **Данные | Группа и структура** существуют две кнопки **Группировать** и **Разгруппировать**. Группы эти можно переименовывать по желанию.
- возможность построения диаграмм на основе сводных таблиц.

**Задание** Создать сводную таблицу, выводящую итоги по средней цене и суммарному пробегу машин, а также использующую в качестве страницы фамилию владельца, в качестве строк — марку машины, в качестве столбцов — год выпуска.

**Решение**

1. Выделите подготовленный диапазон данных либо установите в него указатель ячейки.
2. Выполните команду **Данные | Сводная таблица**.
3. Работая с мастером сводных таблиц, определите все необходимые элементы сводной таблицы.
4. Выполните группировку по полю **Год выпуска** — после создания сводной таблицы поочередно выделите мышью необходимые года выпуска: 1978—1990, 1991—1995, 1996—1999 и воспользуйтесь командой **Группировать** контекстного меню **Группа и структура** либо соответствующей командой меню **Данные | Группа и структура**.
5. Отформатируйте сводную таблицу, вызвав кнопкой **Формат отчета** панели инструментов **Сводные таблицы** диалоговое окно **Автоформат** с вариантами форматирования.
6. Подготовленная сводная таблица представлена на рис. 70.

	A	B	C	D	E	F
1	Владелец	(Все)				
2						
3			Год выпуска2	Год выпуска		
4			1996-1999	2000-2003	2004-2007	Общий итог
5	Марка маши	Данные				
6	Ауди	Среднее по полю Цена у.е.	13900,0	9600,0		12466,7
7		Сумма по полю Цена у.е.	27800,0	9600,0		37400,0
8	БМВ	Среднее по полю Цена у.е.		8250,0	15500,0	11875,0
9		Сумма по полю Цена у.е.		16500,0	31000,0	47500,0
10	Волга	Среднее по полю Цена у.е.	3350,0	2200,0		2966,7
11		Сумма по полю Цена у.е.	6700,0	2200,0		8900,0
12	Мазда	Среднее по полю Цена у.е.		8000,0	12000,0	10000,0
13		Сумма по полю Цена у.е.		8000,0	12000,0	20000,0
14	Мерседес	Среднее по полю Цена у.е.	19666,7		9950,0	15780,0
15		Сумма по полю Цена у.е.	59000,0		19900,0	78900,0
16	Пижо	Среднее по полю Цена у.е.	7500,0		16000,0	10333,3
17		Сумма по полю Цена у.е.	15000,0		16000,0	31000,0
18	Рено	Среднее по полю Цена у.е.		10800,0		10800,0
19		Сумма по полю Цена у.е.		32400,0		32400,0
20	Фольксваген	Среднее по полю Цена у.е.	5900,0		9500,0	7100,0
21		Сумма по полю Цена у.е.	11800,0		9500,0	21300,0
22	Форд	Среднее по полю Цена у.е.		6700,0	8000,0	7566,7
23		Сумма по полю Цена у.е.		6700,0	16000,0	22700,0
24	Итог Среднее по полю Цена у.е.		10936,4	8377,8	11600,0	10348,3
25	Итог Сумма по полю Цена у.е.		120300,0	75400,0	104400,0	300100,0

Рис. 70. Пример сводной таблицы

### Лабораторная работа Создание базы данных

#### **Основы технологии создания базы данных**

Access представляет собой систему управления базой данных (СУБД) офисного типа (т. н. хBase СУБД) и может представлять интерес для начинающего пользователя благодаря сравнительной простоте теоретических концепций и практической работы по проектированию баз данных. Кроме того, Access является разработкой фирмы Microsoft и представляет собой развивающуюся систему. Все это обуславливает широкое распространение этого популярного Windows-приложения.

Начинающий пользователь должен знать, что работа с Access выполняется на этапе реализации (рабочего проектирования) системы базы данных. В таком контексте Access представляет собой инструментальную среду, в которой реализуются результаты предшествующих этапов проектирования. Иными словами, не с Access начинается разработка базы данных, и начинающий пользователь должен знать, в какой момент потребуются соответствующие знания и навыки.

Основным интегрированным объектом Access является база данных.

**База данных (БД)** – это организованная на машинном носителе совокупность взаимосвязанных данных, которая представляет сведения об объектах определенной предметной области (ПО), их свойствах и связях между ними.

Используемым типом структуры данных является *реляционная модель данных (РМД)*. Термин «реляционная» происходит из теории множеств теоретической основы баз данных, где обозначает специфический тип структуры данных, называемый *отношением* (отношение, по-английски *relation*, определяет название модели). Как тип структуры данных отношение допускает наглядное представление данных в виде двумерной таблицы.

Основным элементом структуры БД в Access является *таблица*, с помощью которой моделируются объекты и связи между ними. Следовательно, БД в Access представляет собой совокупность взаимосвязанных двумерных таблиц.

*Структура реляционной таблицы* задается её заголовком и определяется составом и последовательностью *полей-атрибутов*, моделирующих свойства объектов и связей.

Содержание таблицы заключено в её строках. Каждая строка содержит данные об одном экземпляре объекта или связи и называется *записью*.

Для однозначного определения каждой записи (экземпляра объекта) таблица должна иметь уникальный ключ (*первичный ключ*). Ключ может состоять из одного или нескольких полей.

Таблицы в БД взаимосвязаны. Связь каждой пары таблиц, в которой одна является «отцом», а другая – «сыном», обеспечивается ключом связи (*внешним ключом*). Внешний ключ – это первичный ключ таблицы-отца, мигрировавший в таблицу-сына. Связи между таблицами могут быть двух типов: «один к одному» (1:1) или «один ко многим» (1:N).

Структура реляционной БД задается *схемой данных*. Схема данных включает в себя определение структуры каждой таблицы и определение связей между таблицами. Схема данных в Access создается специально для использования связей между таблицами при обработке данных из разных таблиц, а также для обеспечения целостности БД.

Access предоставляет два способа создания новой базы данных: «с нуля» или под управлением Мастера таблиц.

В нотации Access база данных представляет собой составной объект, которому соответствует один единственный файл с расширением **.mdb**.

При создании базы данных возможны разные варианты диалога, отличающиеся к тому же в разных версиях Access элементами интерфейса.

В версии Access 2003, например, создание БД возможно по следующему алгоритму

1. После «запуска» Access в диалоговом окне выбрать «Новая база данных» и нажать ОК.
2. В диалоговом окне «Файл новой базы данных» в соответствующих полях выбрать имя папки (каталога), ввести имя файла БД и нажать экранную кнопку **Создать**.

**Задание 1.** В своей рабочей папке создайте файл базы данных с именем «УЧЕБА».

Файл БД может содержать спецификации до 7 типов объектов-компонентов.

1. Таблицы, которые содержат хранимые данные.
2. Запросы, предназначенные для выборки или изменения данных одной или нескольких связанных таблиц.
3. Формы – экранные формы, предназначенные для ввода и просмотра взаимосвязанных данных в удобном для пользователя БД виде.
4. Отчеты – шаблоны печатных документов для вывода данных в удобном для пользователя БД виде.
5. Страницы – средства доступа к данным через Internet.
6. Макросы – средства автоматизации работы с БД.
7. Модули – компоненты прикладной программы автоматизации работы с БД, составленные на языке Visual Basic.

## **Объекты базы данных**

Таблицы и запросы представляют собой сами данные в виде реальных таблиц, т.е. хранимых данных, или в виде виртуальных таблиц, которые представляются пользователю БД как реально существующие.


Формы и отчеты представляют собой средства разработки интерфейса конечного пользователя БД. Страницы – средства доступа к данным через Internet.


Макросы и модули – это средства автоматизации работы с БД в эргономической среде пользователя.

Работа с БД может быть прервана в любой момент путем закрытия окна

БД или файла БД (по команде **Файл | Закрывать**) и продолжена путем открытия файла БД (по команде **Файл | Открыть**).

Продолжение работы с БД всегда начинается с окна БД (рис. 1). Для работы с тем или иным объектом БД нужно выбрать соответствующую закладку и нажать экранную кнопку:

 **Открыть** – для просмотра результата проектирования объекта;

 **Конструктор** – для изменения структуры объекта;

 **Создать** – для создания объекта «с нуля».

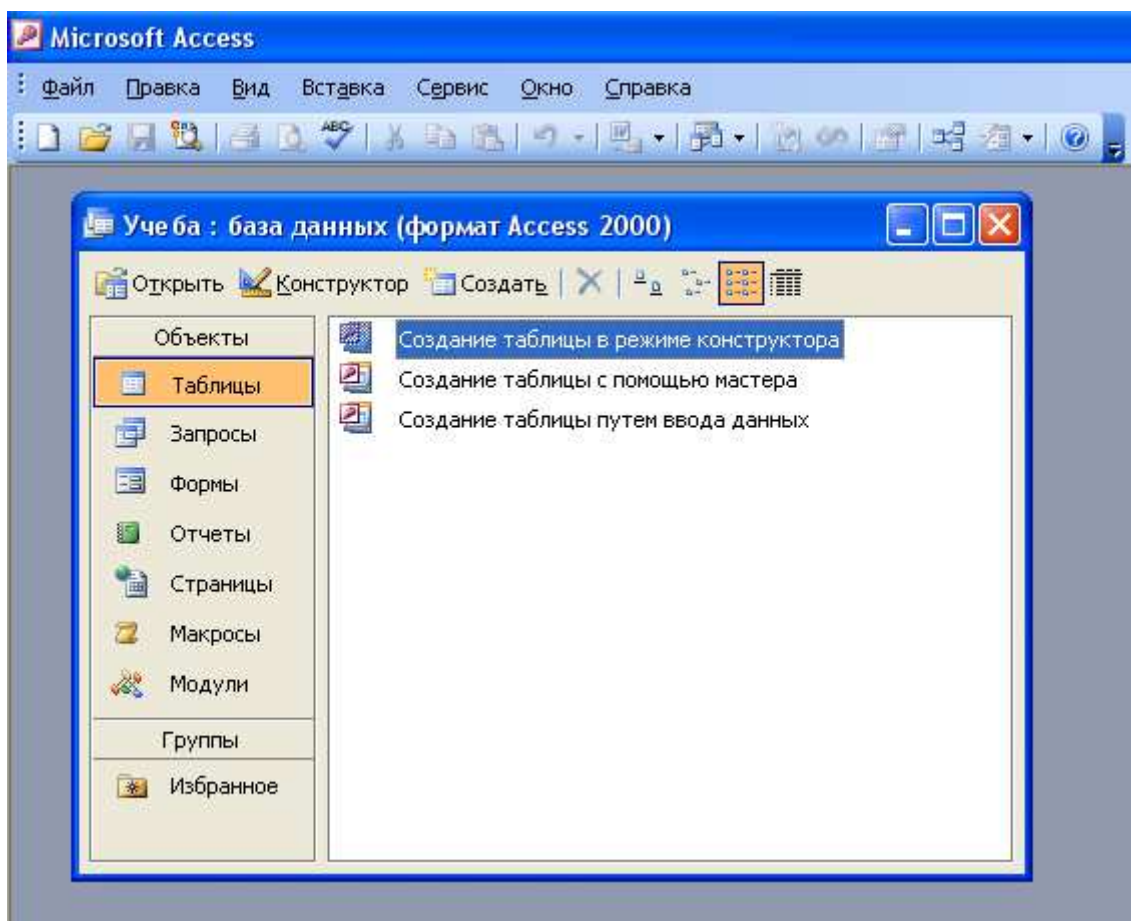


Рис. 1. Окно базы данных Access

## ПРОЕКТИРОВАНИЕ ТАБЛИЦ

### *Разработка структуры таблицы*

#### **Основы технологии формирования структуры таблицы**

Таблицы БД могут быть созданы одним из следующих способов:

- 1) в среде Access в режиме конструирования «с нуля»;
- 2) как результат запроса того или иного типа к уже созданным таблицам;
- 3) в результате импорта таблиц в форматах других сред.

Рассмотрим более подробно основной первый способ.

Создание таблицы «с нуля» выполняется в два этапа. Сначала разрабатывается её структура. Затем выполняется наполнение таблицы необходимой информацией.

При нажатии кнопки открывается окно «Новая таблица» с предложением выбрать способ создания новой таблицы. Наиболее актуальными являются:

- **Мастер таблицы** – для формирования структуры таблицы из полей заданных тематических таблиц;

- **Конструктор** – для создания таблицы «с нуля» (основной способ).

При работе под управлением **Конструктора** формирование структуры новой таблицы выполняется в диалоговом окне (рис. 2). Для каждого поля в соответствующем столбце предлагается указать:

- 1) **имя поля** (набирается вручную в столбце «Поле», можно по-русски);
- 2) **тип данных** (выбирается из списка);
- 3) **описание-комментарий** (не обязательно);
- 4) **свойства** (в зависимости от типа данных).

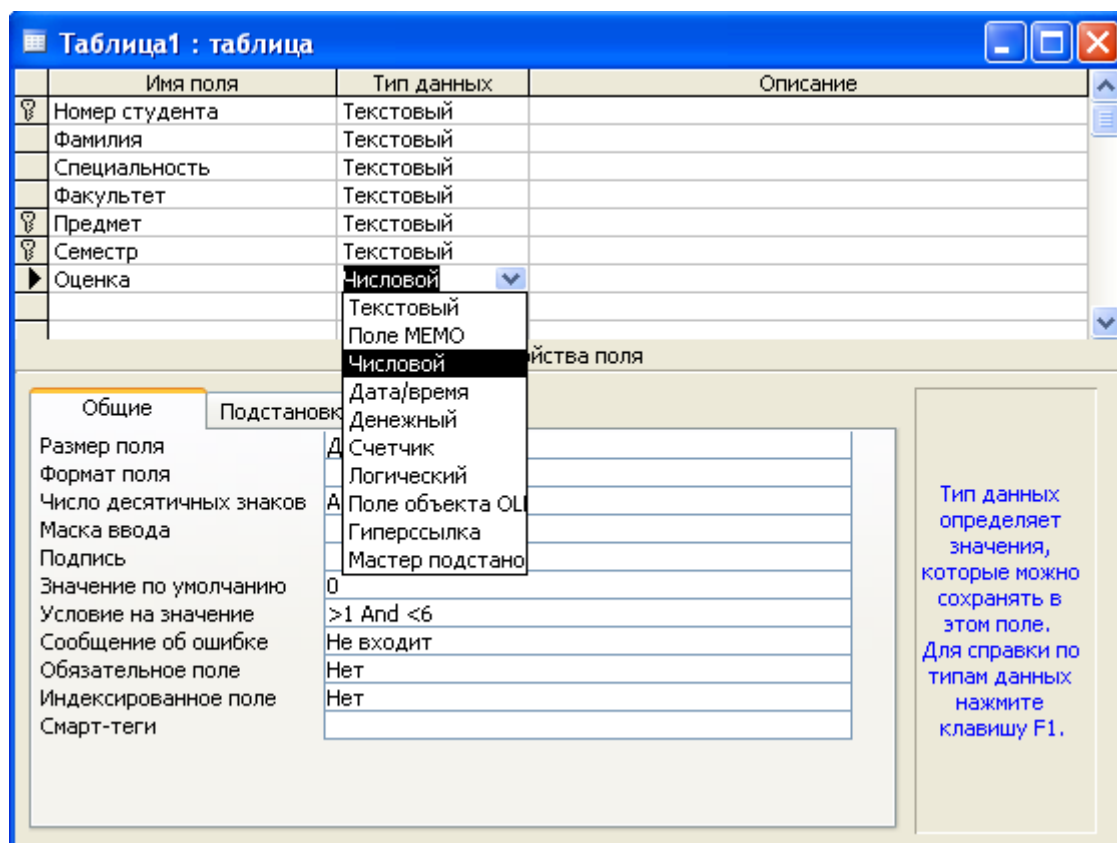


Рис. 2 Окно формирования структуры таблицы

Многообразие типов данных, predeterminedенных в Access (см. таблицу 1), определяет разнообразие информации, которую можно хранить в БД.

Таблица 1

### Типы данных

Типы данных	Диапазон значений	Размер в байтах
1	2	3
Текстовый	0 ÷ 255	1
Числовой (байт)	0 ÷ 255	1
Числовой (целое)	-32768 ÷ 32767	2
Числовой (длинное целое)	-2147483468 ÷ 2147483467	4
Числовой (с плав. точкой)	-3.402823 1038 ÷ 3.402823 1038	4
Числовой (двойной точности)	-1.797693134862232 10308 ÷ 1.797693134862232 10308	8



Логический	Да / Нет	
Дата/Время 100 ÷ 9999 гг.		8

Продолжение таблицы 1

1	2	3
Денежный	15,4	8
Счетчик		4
Поле MEMO	64000	
OLE-объект		109
Гиперссылка	64000	

Большинство типов данных в комментариях не нуждаются. Поясним некоторые из них.

Тип «Счетчик» может быть присвоен не более чем одному полю.

Данные этого типа являются уникальными номерами записей и не могут быть изменены или удалены.

Тип «Денежный» используется для описания данных, над которыми выполняются финансовые операции с высокой точностью вычисления. Точность представления данных этого типа составляет до 15 знаков в целой части и 4 – в дробной.

Тип «поле MEMO» используется для полей, содержащих пространную текстовую информацию.

Тип «поле объекта OLE» присваивается полям-объектам других приложений Windows (рисунок, фотография, звуковая запись, анимация и т. д.).

Тип «Гиперссылка» используется для полей, содержащих адрес удаленного источника данных.

В нижней части окна на закладке «Общие» могут быть заданы свойства полей, которые зависят от типа данных:

- 1) *размер поля* (для числовых полей возможна детализация);
- 2) *формат поля* (внешняя форма представления данных) при помощи специальных символов форматирования;
- 3) *маска ввода* в виде символов, управляющих вводом данных;
- 4) *подпись* – псевдоним поля;
- 5) *значение по умолчанию*;
- 6) *условие на значение* в виде ограничений на допустимые значения поля;
- 7) *сообщение об ошибке*, сопровождающее нарушение условия на значение;
- 8) *обязательное поле* – поле, которое не может принимать не определенное значение (т.н. Null-значение);
- 9) *индексированное поле* – поле, по которому предполагается упорядочение строк таблицы.

Несколько слов об управлении вводом и отображением данных таблиц. В Access существует два свойства полей, использование которых приводит к похожим результатам: свойство *формат поля* и свойство *маска ввода*.

Свойство *формат поля* используется для отображения данных и влияет только на отображение значения, но никак не влияет на значение, хранимое в таблице. Формат отображения применяется только после сохранения введенных данных. Если вводом данных необходимо управлять, в дополнение к формату отображения данных или вместо него используется маска ввода. Если требуется, чтобы данные отображались так, как они были введены, свойство *формат поля* можно не устанавливать.

Маска ввода обеспечивает соответствие данных определенному формату, а также заданному типу значений, вводимых в каждую позицию. Маска ввода является удобным средством контроля вводимых значений.

Если для поля определены как формат отображения, так и маска ввода, то при добавлении и редактировании данных используется маска ввода, а формат поля определяет отображение данных после сохранения записи. При этом необходимо следить за тем, чтобы маска и формат не противоречили друг другу. Управление вводом осуществляется при помощи символов маскирования, которые выполняют специфические функции (см. таблицу 2).

Таблица 2

### Управляющие символы маски ввода

СИМВОЛ	Описание
0	Цифра (от 0 до 9, ввод обязателен; символы плюс [+] и минус [-] не допускаются)
9	Цифра или пробел (ввод не обязателен; символы плюс и минус не допускаются)
#	Цифра или пробел (ввод необязателен; пустые символы преобразуются в пробелы, допускаются символы плюс и минус)
L	Буква (от A до Z или от A до Я, ввод обязателен)
?	Буква (от A до Z или от A до Я, ввод необязателен)
A	Буква или цифра (ввод обязателен)
a	Буква или цифра (ввод необязателен)
&	Любой символ или пробел (ввод обязателен)
C	Любой символ или пробел (ввод необязателен)
.,:; - /	Десятичный разделитель и разделители тысяч, значений дат и времени. (Отображаемый символ зависит от настроек языка и стандартов на панели управления Windows)
<	Указывает перевод всех следующих символов на нижний регистр
>	Указывает перевод всех следующих символов на верхний регистр
!	Указывает заполнение маски ввода справа налево, а не слева направо. Заполнение маски символами всегда происходит слева направо. Восклицательный знак в маске ввода можно помещать в любую позицию
\	Указывает ввод любого следующего символа в качестве текстовой константы. Используется для отображения всех перечисленных в данной таблице символов как текстовых констант (например, \A выводится как символ «А»)
Пароль	Значение <i>Пароль</i> , заданное для свойства <i>Маска ввода (InputMask)</i> , создает поле для ввода пароля. Любой символ, введенный в поле, сохраняется как символ, но отображается как звездочка (*)

#### **Теоретические предпосылки создания таблиц базы данных**

Реализации проекта базы данных предшествует большая аналитическая работа, которая выполняется на этапах инфологического и даталогического проектирования. На основании анализа семантики данных предметной области разрабатывается ее информационная модель, которая затем отображается в компьютерную среду с учетом структурных и процедурных возможностей СУБД реляционного типа. Разработка структуры БД, в том числе разработка структуры каждой таблицы, установление зависимостей между атрибутами, назначение ключевых атрибутов, установление связей между таблицами, является результатом продолжительной аналитической работы. Не нужно забывать, что Access – это только инструментальная среда для реализации информационной модели ПО и что обращаться к ней следует лишь тогда, когда проработаны особенности структуры данных ПО.

В качестве примера рассмотрим упрощенный вариант ПО, называемой «УЧЕБА». Допустим, что для спецификации этой ПО в упрощенном варианте достаточно следующих атрибутов-свойств:

*номер студента* - учётный номер студента, который однозначно идентифицирует его персону;

*фамилия* - фамилия и инициалы студента;

*специальность* - специальность, по которой обучается студент;

*факультет* - факультет, на котором учится студент;

*предмет* - учебная дисциплина, по которой студент в семестре сдает экзамен и получает оценку;

*семестр* - учебный семестр, по итогам которого студент сдает экзамен и получает оценку;

*оценка* - оценка за экзамен.

В реляционном представлении такой ПО можно поставить в соответствие «универсальное» отношение со схемой:

***Учёба (номер студента, фамилия, специальность, факультет, предмет, семестр, оценка)***

здесь подчеркнуты ключевые атрибуты.

Допустим также, что по условиям функционирования ПО УЧЕБА справедливы следующие семантические условия и соответствующие им функциональные (однозначные) зависимости (ФЗ) между атрибутами:

1. Учетный номер студента, а не его фамилия, является уникальной характеристикой студента (т.е. допускаются однофамильцы):

*номер студента -> фамилия*

2. Каждый студент учится только по одной специальности:

*номер студента -> специальность*

3. Каждый студент учится только на одном факультете:

*номер студента -> факультет*

4. По каждой специальности готовят специалистов только на одном факультете:

*специальность -> факультет*

5. Каждый студент (номер студента) по учебному предмету в семестр может получить лишь одну итоговую оценку:

*(номер студента, предмет, семестр) -> оценка*

Схема ФЗ ПО УЧЕБА приведена на рис. 4.

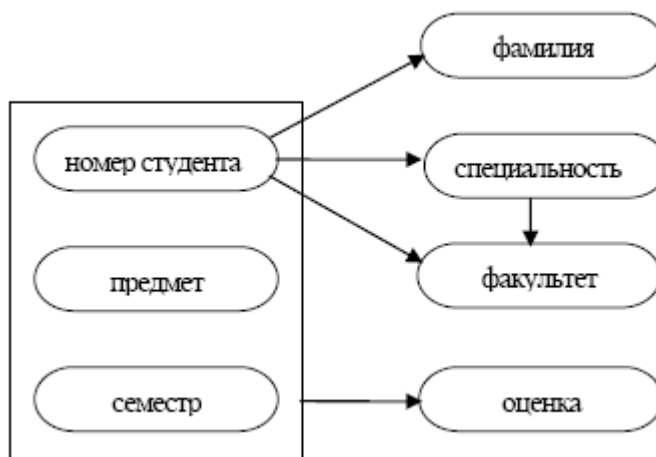
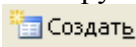
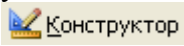
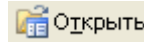


Рис. 4. Схема функциональных зависимостей ПО УЧЕБА

## Основы технологии работы с таблицами

Работа с таблицей в Access может выполняться в двух режимах:

1. Режим **Конструктор** используется при создании новой таблицы или при изменении структуры уже созданной таблицы. Начало работы в окне БД – по экранным кнопкам  или  (рис. 1) на закладке **Таблицы**.

2. **Режим таблицы** используется для добавления, удаления, изменения и просмотра данных в таблице. Начало работы в окне БД – по экранной кнопке  (рис. 1) на закладке **Таблицы**. Переход из одного режима в другой может выполняться:

- 1) по командам **Вид / Конструктор** и **Вид / Режим таблицы** главного меню;
- 2) нажатием экранных кнопок на панели инструментов конструктора таблиц:



– режим конструктора;



– режим таблицы.

**Задание 2.** После анализа ПО можно приступить к реализации проекта в среде Access:

1. В БД УЧЕБА создайте в режиме **Конструктора** таблицу с именем «Учёба». При разработке её структуры воспользуйтесь спецификацией полей атрибутов (см. таблицу 3).
2. По завершении спецификации структуры новой таблицы выполните команду **Файл | Сохранить как...** и присвойте таблице имя «Учеба».

Таблица 3

**Спецификация полей-атрибутов отношения «Учёба»**

Наименование	Тип данных	Размер <sup>1</sup>	Маска ввода	Ограничения
номер студента	Текстовый	6	000000	Ключевой
фамилия	Текстовый	15	>L<????????? \ >L.L.;0	
специальность	Текстовый	6	000000	
факультет	Текстовый	2	<CC	Ключевой
предмет	Текстовый	20	<????????????????????	
семестр	Текстовый	2	99;0	Ключевой
оценка	Числовой	целое		>1 And <6 <sup>2</sup>

3. Задайте в окне разработки структуры таблицы ключевые поля по следующему алгоритму:

- 1) удерживая клавишу **Ctrl** в нажатом состоянии, отметьте мышью в колонке слева от названия полей все ключевые поля (они должны выделиться цветом);
- 2) выполните команду **Правка / Ключевое поле** главного меню (в колонке слева от имени поля должно появиться изображение ключа).

По окончании формирования спецификация структуры таблицы «Учёба» должна выглядеть так, как показано на рис. 3.

## НАПОЛНЕНИЕ И РЕДАКТИРОВАНИЕ ТАБЛИЦЫ

Пока таблица не заполнена данными, можно смело редактировать её структуру в режиме **Конструктора**:

1. Добавить новое поле:
  - 1) выделить мышью в колонке слева от имени поля строку таблицы структуры;
  - 2) выполнить команду **Вставка / Строки**.
2. Удалить поле:
  - 1) выделить мышью в колонке слева от имени поля строку таблицы структуры;
  - 2) выполнить команду **Правка / Удалить строки**.
3. Переименовать поле.
4. Переопределить тип поля.

### 5. Определить ключ.

Если таблица уже заполнена данными, редактирование её структуры может повлечь за собой потерю данных. Например, если при изменении типа данных конкретное значение невозможно преобразовать к новому типу данных, то оно удаляется и заменяется во всех содержащих это значение полях на пустое значение. Например, если поле преобразуется из типа «Текстовый» в тип «Числовой», то будет удалено любое значение, содержащее одну или несколько букв. При этом выводится предупреждение о числе удаляемых значений, и пользователь имеет возможность отменить преобразование.

Наиболее часто встречающиеся преобразования типов данных можно представить следующими четырьмя категориями:

- 1) преобразование в текст других типов данных;
- 2) преобразование текстовых данных в числовые, денежные, логические форматы или в формат дата/время;
- 3) преобразование денежного типа данных в числовой, текстового в поле МЕМО и наоборот;
- 4) изменение настройки свойства **Размер поля** для числовых полей.

**Задание 3.** Наполните таблицу «Учёба» содержательной информацией (табл. 4):

Таблица 4

**Информация для ввода в таблицу «Учёба»**

Номер студента	Фамилия	Специальность	Факультет	Предмет	Семестр	Оценка
00001	Иванов И.П.	Менеджмент организации	Экономический	Информатика	1	4
00002	Лукин И.Р.	Маркетинг	Экономический	История	3	3
01235	Сергеев В.И.	Бухгалтерский учет	Экономический	Маркетинг	5	5
02351	Петров Т.А.	Маркетинг	Экономический	Философия	4	4
01112	Сидоров П.А.	Информатика	Технологический	Программиров	3	4
02314	Петрова О.А.	Электроснабжение	Технологический	Электротехник	2	3
14123	Сидоренко О.Р.	Электроснабжение	Технологический	Информатика	1	5
00021	Бондаренко А.А.	Маркетинг	Экономический	Философия	2	3
						0

1) перейдите в **Режим таблицы**. Обратите внимание на контекстное изменение главного меню. В нем появились пункты «Формат» (для изменения размеров полей таблицы) и «Записи» (для работы с записями таблицы);

2) заполните шаблон таблицы «Учёба» данными (табл. 4). Обратите внимание, что расположение записей в таблице при вводе определяется не хронологией ввода, а условиями упорядочения «по ключу» (в порядке возрастания ключа);

3) по завершении ввода сохраните результаты по команде **Файл / Сохранить** главного меню.

В **Режиме таблицы** можно выполнить простейшие интерактивные (без запоминания условий) запросы-выборки:

1. Поиск записи по заданному значению поля (выполняется по команде **Правка / Найти**).
2. Сортировка записей по значению поля (выполняется по команде **Записи | Сортировка**).
3. Выборка по условию (выполняется по команде **Записи / Фильтр**).

## Лабораторная работа Формирование запросов

### Спецификация запросов

База данных представляет собой информационную модель ПО и предназначена для удовлетворения информационных запросов пользователей, работающих в этой ПО. После того как БД будет наполнена содержательной информацией, к ней можно обращаться с запросами. *Запрос* представляет собой спецификацию условий манипулирования данными, в результате которого создается то или иное представление о хранимой в БД информации. БД является объектом коллективного пользования, и каждый запрос выражает индивидуальное, локальное представление пользователей о структуре информации моделируемой ПО.

Запросы к БД можно разделить на две категории.

1. **Запросы-выборки** осуществляют только извлечение информации из одной или нескольких связанных таблиц и представление её в виде виртуальной (не существующей на самом деле) таблицы. Структура такого рода таблицы определяется локальным представлением о ПО конечного пользователя БД в контексте запроса.

2. **Корректирующие запросы** или запросы на добавление, удаление и изменение данных в БД. Это весьма значительные по своим последствиям запросы, так как их реализация приводит к изменению хранимой в БД информации. Такого рода запросы соответствуют функциям администрирования БД.

БД одной и той же ПО может быть представлена в виде одной таблицы («универсального отношения») или в виде совокупности связанных таблиц.

Для спецификации запросов необходимы соответствующие языковые средства – *языки запросов*. Языки запросов – это языки высокого уровня, не процедурные (как Паскаль, Си, Бейсик и др.), а декларативные. Это значит, что в запросе достаточно указать, что нужно сделать, а реализация запроса скрыта от пользователя.

В Access (как и во многих других СУБД) используются языки запросов двух типов:

1. **QBE** (Query By Example) – запрос по образцу. Запрос формируется в режиме диалога путем заполнения таблицы специального вида. QBE- технология дает пользователю визуальное представление достаточно сложного запроса.

2. **SQL** (Structured Queries Language) – язык структурированных запросов. Запрос формируется в терминах команд этого языка (т. е. нужно знать язык SQL). В настоящее время SQL-технология запросов получила широкое распространение в системах БД на основе распределенных систем, в компьютерных сетях.

## ПРОЕКТИРОВАНИЕ СТРУКТУРЫ ЗАПРОСА

### *Основы технологии проектирования запросов*

Основной технологией формирования запросов в Access является QBE- технология с возможностью отображения запроса в стиле SQL.

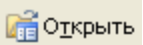
Работа с запросами в Access может выполняться в трех режимах:

1. Режим **Конструктора** (основной) используется при создании нового запроса или при изменении структуры уже созданного запроса в стиле QBE.

Начало работы в окне БД – по экранным кнопкам  **Создать** или  **Конструктор** (рис. 1) на закладке **Запросы**.

2. Режим **SQL** используется для просмотра уже созданного QBE- запроса, а также для создания нового запроса или при изменении структуры уже созданного в стиле QBE-запроса.

3. Режим **таблицы** используется для просмотра результатов запроса. Начало работы в окне БД –

по экранной кнопке  **Открыть** (рис. 1) на закладке **Запросы**.

Переход из одного режима в другой может выполняться:

1) по командам **Вид / Конструктор**, **Вид / Режим SQL** и **Вид / Режим таблицы** главного меню;

2) нажатием экранных кнопок панели инструментов конструктора запросов:



– режим конструктора;



– режим таблицы;



– режим SQL.

### **Основной алгоритм построения запросов**

Формирование запроса выполняется примерно по следующему алгоритму:

1. В окне БД выбрать закладку **Запрос** и нажать экранную кнопку **Создать**.

2. В окне «Новый запрос» выбрать вариант **Конструктор**.

3. В окне «Добавление таблицы» выбрать по закладке источник данных к запросу: **Таблицы / Запросы | Таблицы и Запросы**.

4. Выбрать в списке конкретную таблицу или запрос и нажать экранную кнопку **Добавить** и в конце **Заккрыть**.

5. В окне конструктора оформить условия спецификации конкретного запроса (как это делается, рассмотрим позднее на конкретных примерах типов запросов). Обратите внимание, что в контекстном меню проявился новый пункт меню «Запрос».

6. Посмотреть результат запроса по команде **Запрос→Запуск**. На экране появится окно просмотра данных в **Режиме таблицы**.

7. Вернуться в режим **Конструктора** и принять решение, удовлетворяют ли вас результаты запроса или нет. При отрицательном исходе можно продолжить формирование запроса.

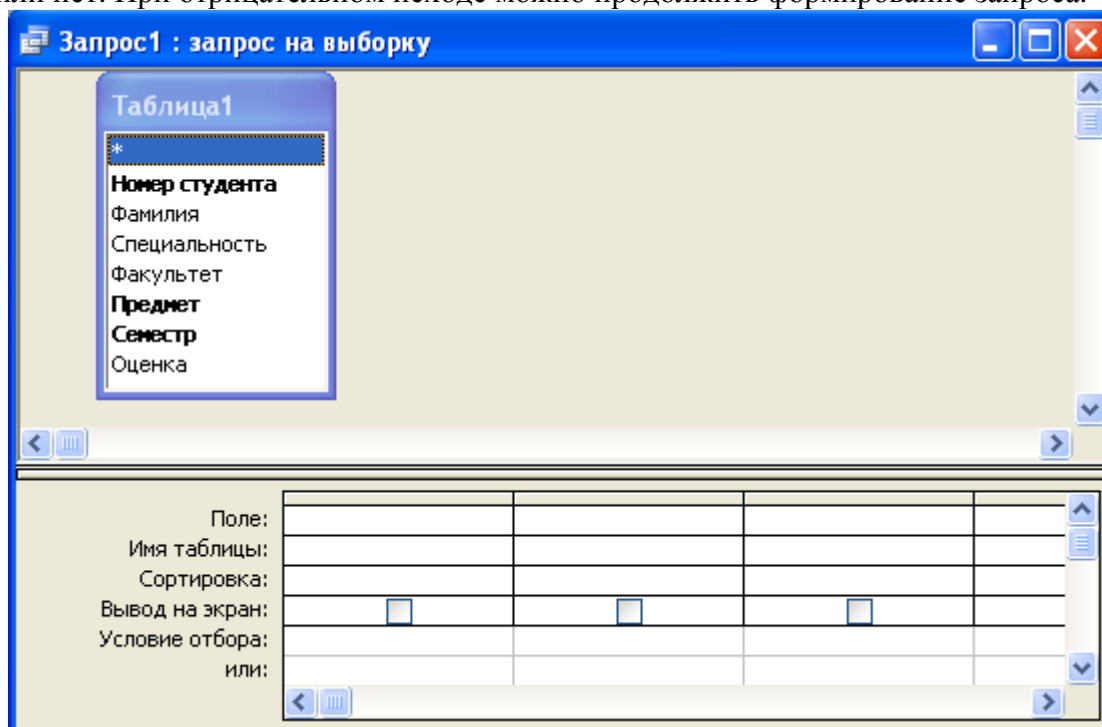



Рис. 5. Окно конструктора QBE-запросов

8. Если результаты запроса вас удовлетворяют, запрос можно сохранить по команде **Файл / Заккрыть** или **Файл | Сохранить как**, указав в диалоговом окне имя запроса, под которым он будет представлен в списке на закладке **Запросы** текущей БД.

9. Для воспроизведения результата запроса необходимо выбрать запрос по имени в списке в окне

БД на закладке **Запросы** и нажать экранную кнопку **Открыть**.

10. Для изменения структуры запроса необходимо выбрать запрос по имени в списке в окне БД на закладке **Запросы** и нажать экранную кнопку  **Конструктор**.

## ЗАПРОСЫ ВЫБОРКИ

При сохранении запросов рекомендуется присваивать им содержательные имена, которые указывали бы на смысл запроса.

### **Вывод всех полей**

**Задание 4.** Сформируйте запрос-выборку всех полей таблицы «Учёба» БД УЧЕБА:

- 1) перенесите мышью в строку «Поле» таблицы запроса имена *всех* полей из списка в верхней части таблицы;
- 2) включите щелчком мыши по квадратикам в строке «Вывод на экран» режим просмотра соответствующих полей – спецификация запроса примет вид таблицы, приведенной на рис. 6;
- 3) завершите результаты формирования запроса так. Результаты запроса представлены на рис. 7;
- 4) перейдите в SQL-режим по команде **Вид / Режим SQL**, чтобы получить представление о структуре SQL-запроса;
- 5) сохраните запрос под именем «Все поля».

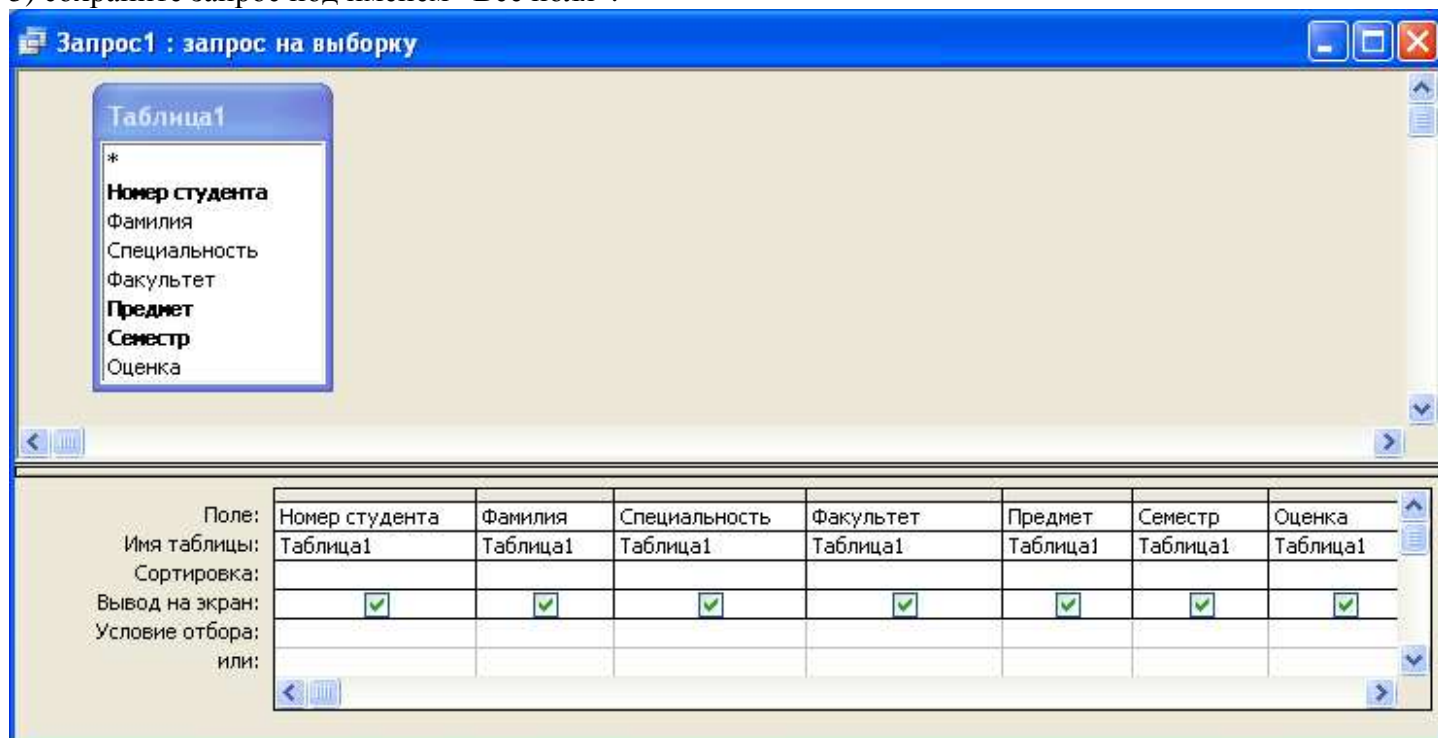


Рис. 6. Структура QBE - запроса «Все поля»



Номер студента	Фамилия	Специальность	Факультет	Предмет	Семестр	Оценка
00001	Иванов И.П.	Менеджмент организации	Экономический	Информатика	1	4
00002	Лукин И.Р.	Маркетинг	Экономический	История	3	3
00021	Бондаренко А.А.	Маркетинг	Экономический	Философия	2	3
01112	Сидоров П.А.	Информатика	Технологический	Программиров	3	4
01235	Сергеев В.И.	Бухгалтерский учет	Экономический	Маркетинг	5	5
02314	Петрова О.А.	Электроснабжение	Технологический	Электротехник	2	3
02351	Петров Т.А.	Маркетинг	Экономический	Философия	4	4
14123	Сидоренко О.Р.	Электроснабжение	Технологический	Информатика	1	5
						0

Запись: 9 из 9

Рис. 7. Результаты запроса «Все поля»

### Выборка полей

В структуре запроса могут быть представлены не все поля, а только некоторые (вертикальная выборка). В результате в итоговой таблице могут присутствовать дубли строк. Это происходит по понятной причине: строки-дубли образуются потому, что поля, по которым они различались в исходной таблице, остались за пределами выборки.

**Задание 5.** Сформируйте запрос-выборку некоторых полей таблицы «Учёба» БД УЧЕБА.

- 1) сформируйте запрос, в котором присутствовали бы следующие поля: «номер студента», «фамилия», «специальность», «факультет» (технология построения запроса такая же, как в примере «Все поля», отличается только количеством полей);
- 2) просмотрите результаты запроса. Обратите внимание на присутствие в итоговой таблице дублирующих записей;
- 3) назовите запрос «Дубли».

### Выборка полей без дублей записей (проекция)

Дубли записей в таблице являются её недостатком потому, что не несут никакой полезной информации. Избавиться от дублей можно, если настроить свойство запроса, которое запрещает совпадение значений данных в выбранных полях. Итоговая таблица по такому запросу представляет собой результат реляционной операции *проекции* исходной таблицы на выбранные поля-атрибуты.

в запрос, в котором присутствовали бы следующие поля: «номер студента», «фамилия», «специальность», «факультет» (технология построения запроса такая же, как в предыдущем примере);

- 2) в режиме **Конструктора** выполните команду **Вид / Свойства**;
- 3) в диалоговом окне «Свойства запроса» переключите мышью свойство «Уникальные значения» в состояние «Да» (рис. 8);

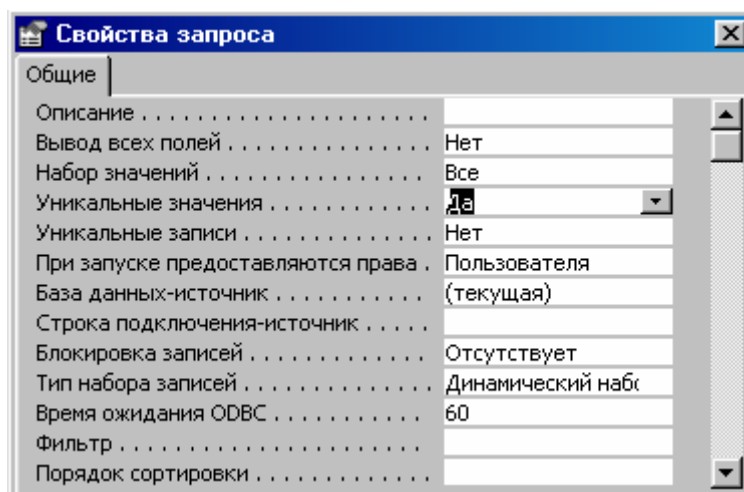


Рис. 8. Спецификация свойств запроса «без дублей записей»

- 4) завершите результаты формирования запроса;
- 5) назовите запрос «Проекция».

### Запросы на основе операции селекции

Очень распространенным типом запросов являются запросы выборки записей-строк по условию (горизонтальная выборка). Итоговая таблица в таком случае представляет собой результат реляционной операции *селекция*.

Условия отбора записываются в строке «Условие отбора» структуры запроса и могут состоять из литералов, констант и идентификаторов, соединенных операторами. Условия отбора можно записывать «по памяти» или при помощи специального инструмента, так называемого **Построителя выражений**.

Доступ к этому средству осуществляется в контексте построения условия при нажатии правой клавиши мыши и выборе пункта **Построить** контекстного меню (рис. 9).

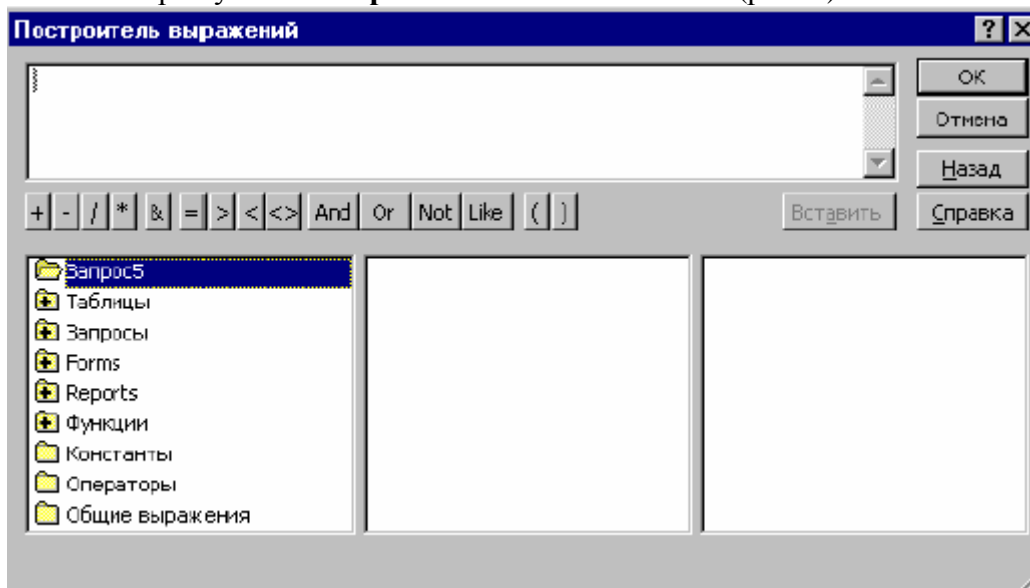


Рис. 9. Окно построителя выражений условий

### Выборка по значению поля

**Задание 7.** Для выбора всех данных о студентах, обучающихся на факультете «экономический»:

- 1) выберите все поля в структуру запроса, как в примере «Все поля»;
- 2) в поле «факультет» в строке «Условие отбора» наберите вручную значение «экономический» (рис. 10);

- 3) завершите результаты формирования запроса;
- 4) назовите запрос «Селекция».

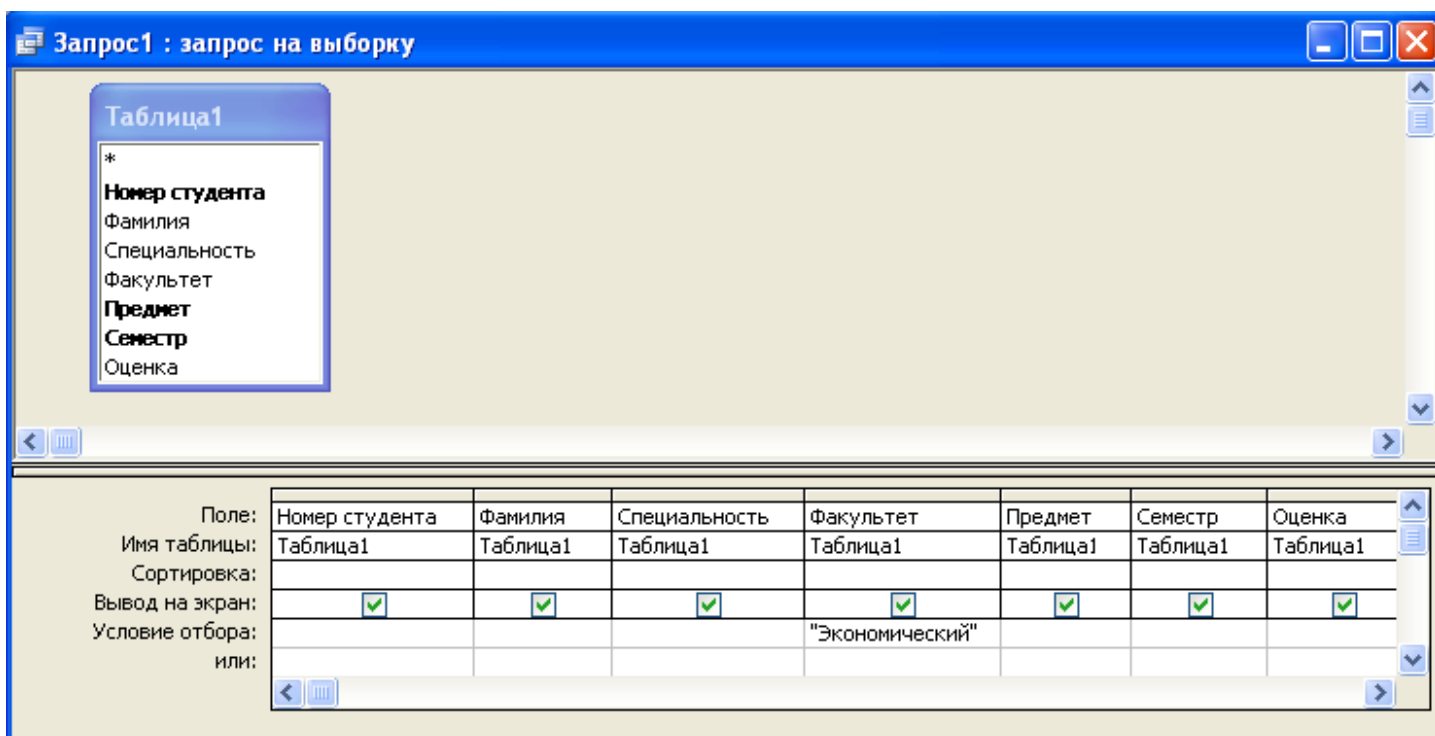


Рис. 10. Структура запроса-выборки по значению поля

### Параметрический запрос

Значение поля в условии запроса можно вводить не только непосредственно в шаблон структуры запроса, как в предыдущем примере, но и динамически в диалоге в процессе выполнения запроса. Для этой цели используется *параметрический запрос*.

**Задание 8.** Сформируйте параметрический запрос всех данных о результатах сдачи экзаменов по предмету, заданному его названием:

- 1) выберите все поля в структуру запроса, как в примере «Все поля»;
- 2) в поле «предмет» в строке «Условие отбора» наберите в квадратных скобках текст подсказки к вводу конкретного значения (рис. 11);

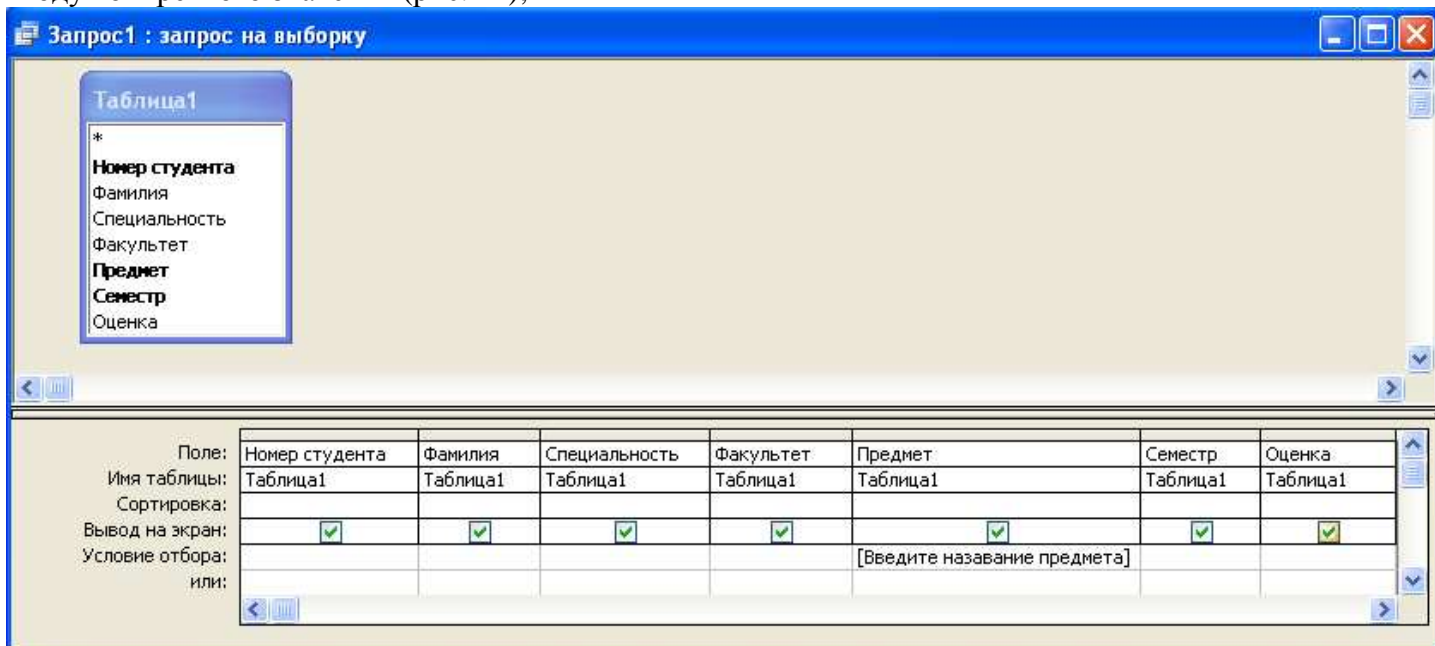


Рис. 11. Структура параметрического запроса

- 3) завершите результаты формирования запроса;
- 4) в поле диалогового окна, которое появится при выполнении запроса, введите конкретное значение, например «математика» (рис. 12);
- 5) сохраните запрос под именем «Параметрический».

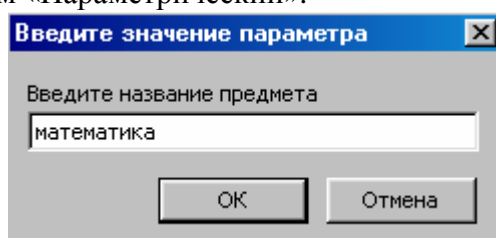


Рис. 12. Диалоговое окно параметрического запроса

### **Выборка по условию с операторами сравнения**

В спецификации условий выборки можно использовать операторы *сравнения* (=, <, >, <=, >=). Для этого в шаблоне структуры запроса в строке «Условие запроса» в соответствующем поле указывается оператор сравнения.

**Задание 9.** Сформируйте запрос на выборку всех данных о студентах-младшекурсниках (1–3 курсы) может быть формализован условием «<7» по полю «семестр»:

- 1) выберите все поля в структуру запроса, как в примере «Все поля»;
- 2) в поле «семестр» в строке «Условие отбора» наберите «<7»;
- 3) завершите результаты формирования запроса;
- 4) назовите запрос «Меньше».

### **Выборка по условию с логическими операторами**

В спецификации условий выборки можно использовать *логические операторы* (Not, And, Or и т. д.). Речь идет о реализации сложных условий запроса. Для спецификации условия с **And** операнды логического выражения должны располагаться в строке «Условие запроса» на одном уровне в шаблоне структуры запроса. Для спецификации условия с **Or** операнды логического выражения должны располагаться в строке «Условие запроса» и последующих строках «или» на разных уровнях в шаблоне структуры запроса.

**Задание 10.** Выполните пример на тему And. Сформируйте запрос на выборку всех данных о результатах сдачи экзаменов студентами факультета «Технологический» на оценки 4 и 5 (т.е. более 3).

- 1) перенесите все поля в структуру запроса, как в примере «Все поля»;
- 2) в поле «факультет» в строке «Условие отбора» наберите «Технологический», а в поле «оценка» **в той же строке** наберите «>3» (рис. 13);
- 3) завершите результаты формирования запроса;
- 4) назовите запрос «And».

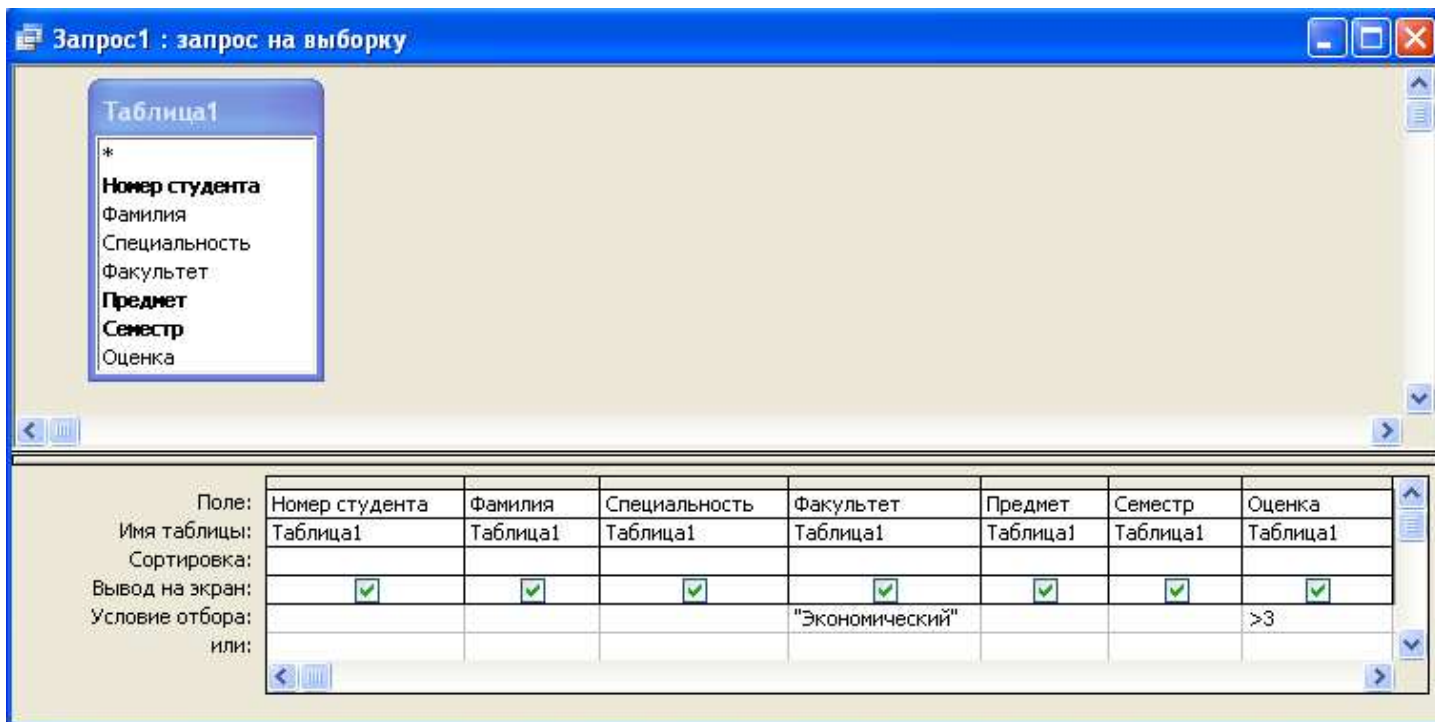


Рис. 13. Структура запроса с And

**Задание 11.** Выполните пример на тему Or. Сформируйте запрос на выборку всех данных о результатах сдачи экзаменов студентами, которые или учатся на факультете «Экономический», или получили оценки 4 и 5:

- 1) перенесите все поля в структуру запроса, как в примере «Все поля»;
- 2) в поле «факультет» в строке «Условие отбора» наберите «**Экономический**», а в поле «оценка» в строке *или* наберите «>3»;
- 3) завершите результаты формирования запроса;
- 4) назовите запрос «Or».

### **Выборка по условию с оператором Between...And**

Оператор **Between** определяет принадлежность значения поля таблицы указанному диапазону значений и имеет следующий синтаксис применения:

**Between «значение1» And «значение2»**, здесь «значение1» и «значение2» – выражения, задающие границы диапазона. Порядок следования операндов «значение1» и «значение2» значения не имеет.

Если значение поля попадает в диапазон, задаваемый выражениями «значение1» и «значение2» (включительно), оператор **Between...And** возвращает значение «истина»; в противном случае возвращается значение «ложь». Допускается проверка с помощью логического оператора **Not** обратного условия (т.е. того, что выражение не принадлежит диапазону значение1 – значение2).

**Задание 12.** Сформируйте запрос на выборку всех данных о результатах сдачи экзаменов студентами, которые сдали экзамены на 4 или 5 (**Between 4 And 5**):

- 1) перенесите все поля в структуру запроса, как в примере «Все поля»;
- 2) в поле «оценка» в строке «Условие отбора» наберите **Between 4 And 5** (рис. 14);
- 3) завершите результаты формирования запроса так, как это обсуждалось в алгоритме формирования запроса на стр. 18;
- 4) назовите запрос «Between».

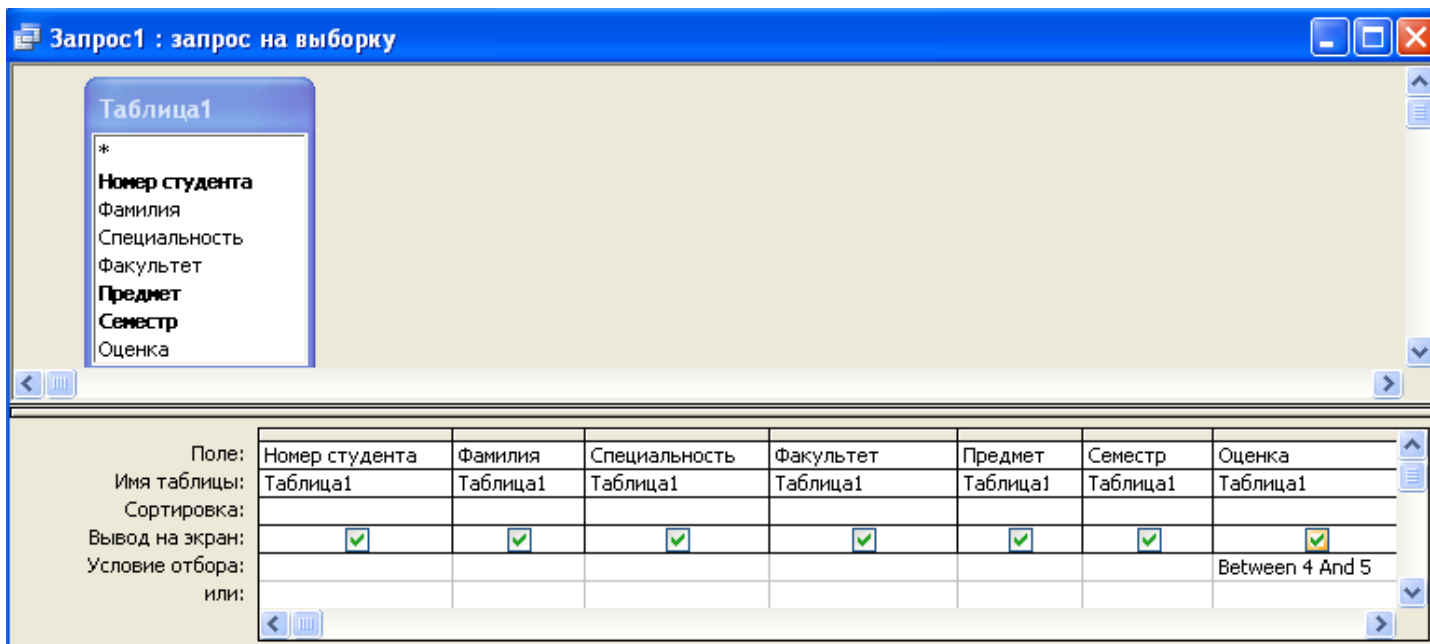


Рис. 14. Структура запроса с Between

Границы диапазона можно задать параметрически по образцу **Between [от] And [до]**.

### **Выборка по условию с оператором Like**

Оператор **Like** имеет синтаксис **Like «шаблон»** и используется для поиска в поле текстового типа значений, совпадающих с указанным образцом. Допускается как точное указание аргумента «шаблон» (например, Like «Иванов И.И.»), так и использование поддерживаемых операционной системой символов шаблона, определяющих диапазон значений (например, Like «И\*»).

При подстановке в выражение оператор Like позволяет сравнивать значение поля со строковым значением. Например, при вводе в строку «Условие отбора» таблицы запроса выражения Like «И\*» в запросе будут выведены все поля, содержимое которых начинается с буквы «И».

**Контрольное задание 13.** Сформируйте запрос на выборку всех данных о результатах сдачи экзаменов студентами, чьи фамилии заканчиваются на «ов» (Петров, Иванов и т.д.):

- 1) перенесите все поля в структуру запроса, как в примере «Все поля»;
- 2) в поле «фамилия» в строке «Условие отбора» наберите **Like «\*ов\*»** (после «ов» пробел);
- 3) завершите результаты формирования запроса;
- 4) назовите запрос «Like».

Запрос с Like, как любой запрос, может быть задан параметрически.

В запросе с параметром допускается вывод для пользователя приглашения ввести образец поиска.

**Задание 14.** Сформируйте параметрический запрос на выборку всех данных о студентах, чьи фамилии начинаются с первых букв (одной, двух, сколько нужно по контексту поиска), задаваемых в диалоге:

- 1) перенесите все поля в структуру запроса, как в примере «Все поля»;
- 2) в поле «фамилия» в строке «Условие отбора» наберите **Like [Введите первые буквы фамилии:]&«\*»** (рис. 15);
- 3) завершите результаты формирования запроса.

При выполнении запроса будет открыто окно диалога с надписью над полем ввода «Введите первые буквы фамилии». Если пользователь введет в поле буквы «Ив», то в запросе будет выполняться поиск по шаблону «Ив\*» (т.е. будут найдены все фамилии, начинающиеся с букв «Ив»);

5) назовите запрос «Like параметрический».

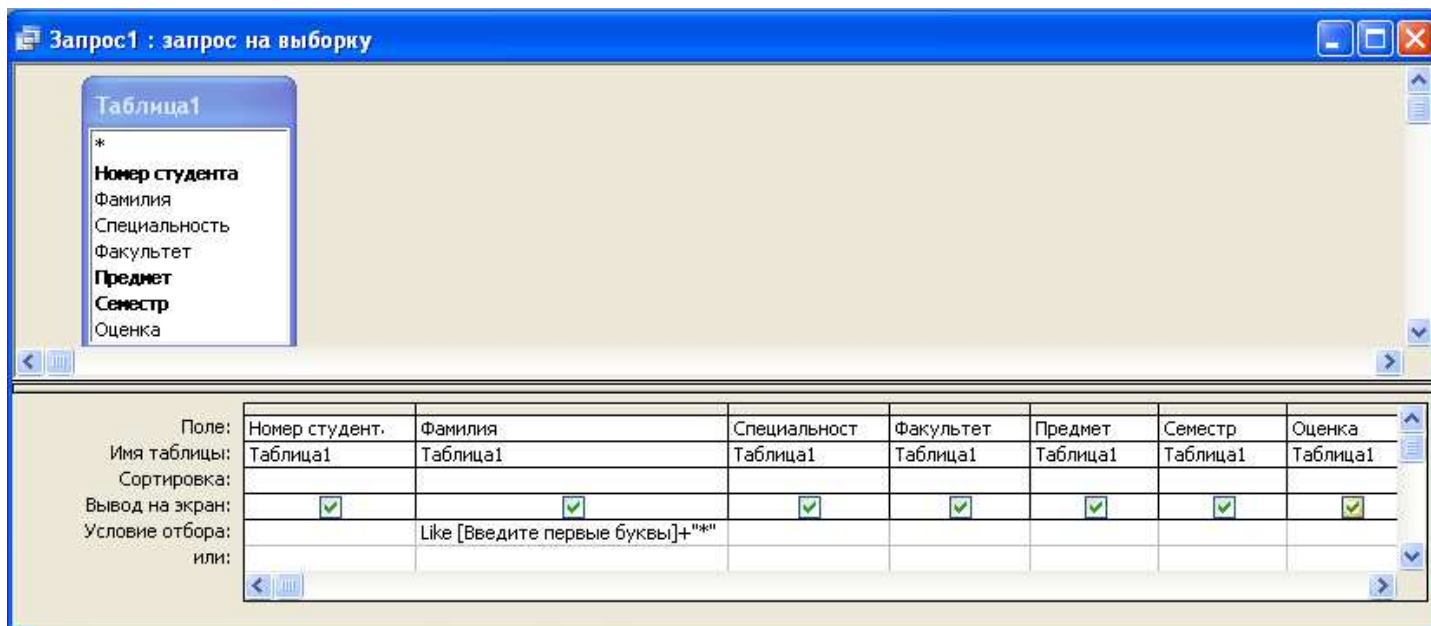


Рис. 15. Структура запроса с Like

#### Выборка по условию с оператором In

Оператор **In** имеет синтаксис **In**(«значение1»;«значение2»; ...) и проверяет значение поля таблицы на совпадение с любым из элементов указанного списка значений. Если содержимое поля совпадает со значением одного из элементов списка, оператор **In** возвращает значение «истина»; в противном случае возвращается «ложь». Допускается проверка с помощью логического оператора **Not** обратного условия (т.е. того, что выражение не принадлежит списку значений).

**Задание 15.** Сформируйте запрос на выборку всех данных о результатах сдачи экзаменов по базовым дисциплинам (физика, математика и др.):

- 1) перенесите все поля в структуру запроса, как в примере «Все поля»;
- 2) в поле «специальность» в строке «Условие отбора» наберите **In** («физика»;«математика») (рис. 16);
- 3) завершите результаты формирования запроса;
- 4) сохраните запрос под именем «In».

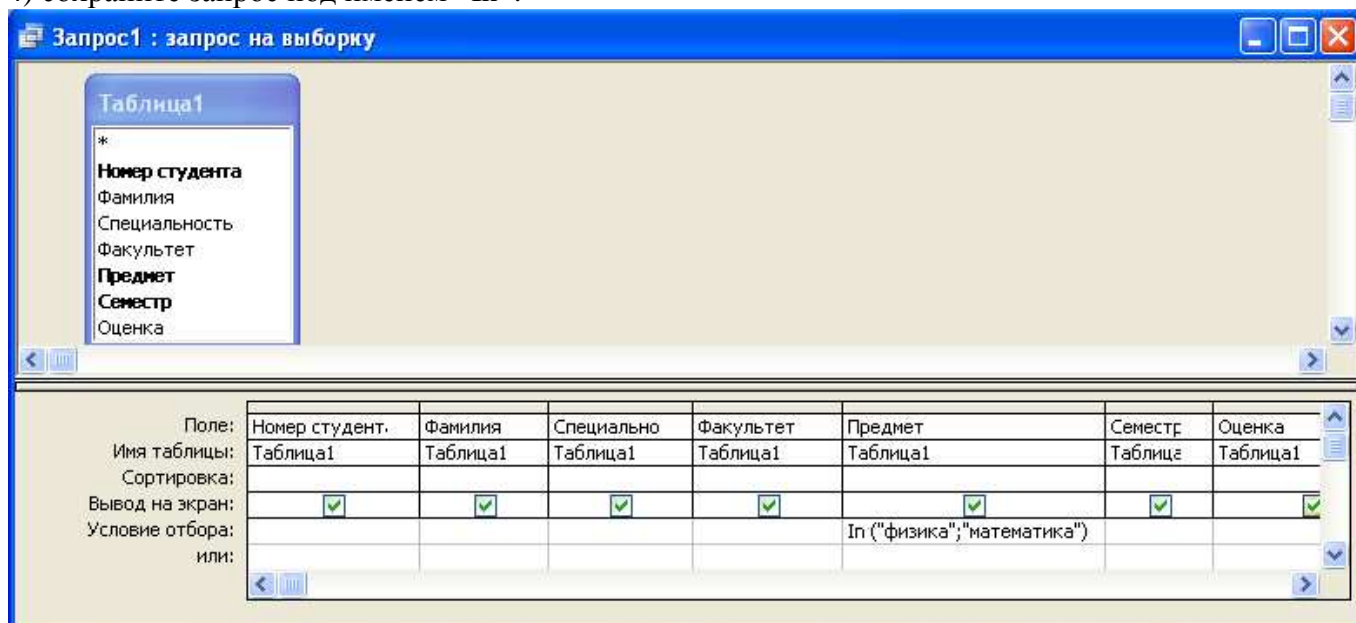


Рис. 16. Структура запроса с In

### **Выборка с вычисляемыми полями**

В структуре запроса наряду с полями реальных таблиц могут быть представлены данные, которые отсутствуют в реальных таблицах, но могут быть выведены из хранимых данных при помощи арифметических, логических и других операторов, а также встроенных функций. Типичными примерами такого рода данных являются вычисляемые (виртуальные) и агрегированные данные (количество, сумма, среднее, минимальное, максимальное значения и др.).

**Задание 16.** Сформируйте запрос-выборку с вычисляемыми данными из БД УЧЕБА, представленной одной таблицей «Учёба». Допустим, что по условиям функционирования нашей модельной ПО «Учёба» две первые цифры номера студента указывают на год его зачисления в вуз. Допустим также, что датой зачисления студента является 1 сентября года приема. Добавим в структуру запроса на выборку всех данных после поля «номер студента» вычисляемое поле «дата зачисления», построенное по следующей формуле:

**DateValue(«20»+Mid([Учеба]![номер студента];1;2)+«01.09»)**

Здесь функция **DateValue(«строка символов»)** преобразует строковое представление даты в данное типа дата.

Функция **Mid(«строка символов»; «начало»; «длина»)** вырезает из «строки символов», начиная с позиции «начало» (целое число), подстроку символов длиной «длина» (целое число) символов.

Оператор + выполняет сцепление (конкатенацию) строк символов.

Построение формулы для вычисления виртуального поля удобно производить под управлением **Построителя выражений**.

Сформируйте запрос по следующему алгоритму:

- 1) перенесите все поля в структуру запроса, как в примере «Все поля»;
- 2) для добавления нового поля выделите поле, *слева* от которого будет выполнена вставка. В нашем случае это поле «фамилия». (Для выбора столбца следует поместить курсор в любую его ячейку или нажать кнопку в области маркировки полей в верхней части таблицы запроса);
- 3) выполните команду **Вставка / Столбцы**. (Для удаления поля следует выделить соответствующий столбец и выполнить команду **Правка | Удалить**);
- 4) в строке «Поле» наберите формулу для вычисляемого поля. Формулу можно вводить вручную или конструировать при помощи **Построителя выражений** (рис. 17) по принципу «снаружи внутрь»;
- 5) в списке «Функции» («Встроенные функции») в категории «Дата/время» найдите функцию **DateValue («stringexpr»)**;
- 6) аргумент функции «stringexpr» замените на сцепление трех строк символов: «20»+ Mid («stringexpr»; «start»; «length»)+ «01.09»;
- 7) функцию Mid найдете в списке «Функции» («Встроенные функции») в категории «Текстовые»: **Mid («stringexpr»; «start»; «length»)**;
- 8) аргумент «start» замените на «1» (с первой позиции), аргумент «length»— на «2» (подстрока длиной 2 символа);
- 9) аргумент «stringexpr» найдите в списке «Таблицы» («Учеба», поле «номер студента») **[Учеба]![номер студента]**;



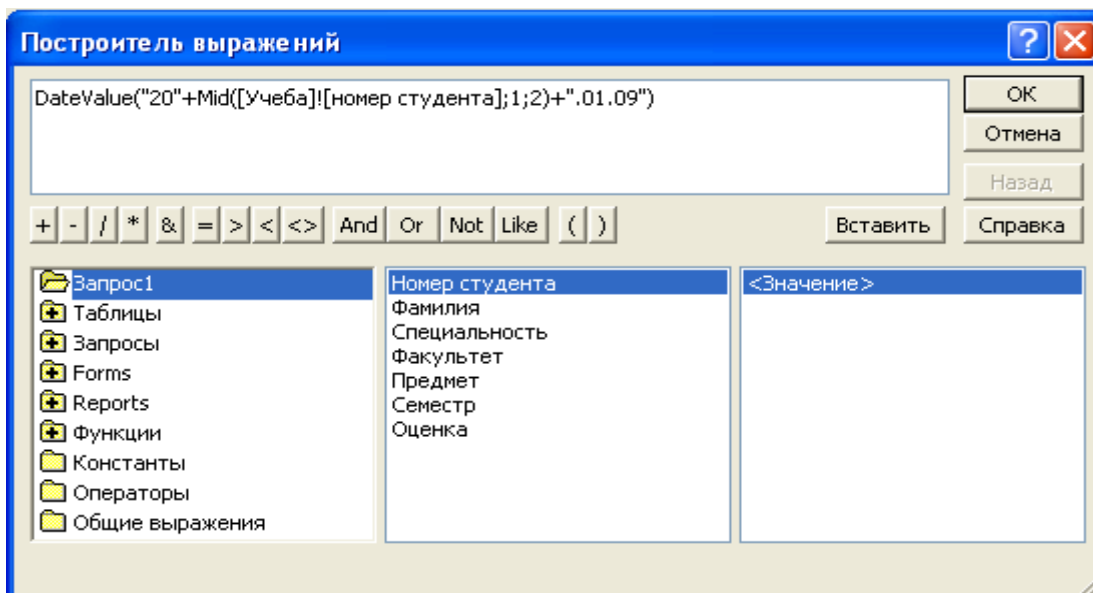


Рис. 17. Построение выражения для вычисляемого поля

- 10) в таблице результата запроса новое поле будет иметь имя «Выражение1». Чтобы присвоить ему содержательное имя, установите курсор в ячейку «Поле» и нажмите правую клавишу мыши;
- 11) в контекстном меню выполните команду **Свойства** и введите в диалоговом окне «Свойства поля» в строку «Подпись» имя поля «дата зачисления»;
- 12) завершите результаты формирования запроса;
- 13) результат запроса будет иметь вид, как на [рис. 18](#);
- 14) сохраните запрос под именем «Вычисляемое поле».

Рис. 18. Результат запроса с вычисляемым полем

### Сортировка записей

Изначально записи-строки таблицы БД оказываются отсортированными по возрастанию ключа. Если по смыслу запроса требуется рассортировать записи по другому критерию, то формируют запрос на *сортировку*. Сортировка может выполняться по возрастанию или по убыванию значений поля. Сортировка может выполняться по нескольким полям. В этом случае порядок следования полей в структуре запроса определяет порядок сортировки (вложенная сортировка). Сначала выполняется сортировка по первому полю. В пределах внешней сортировки выполняется сортировка по второму полю и т. д.

**Задание 17.** Постройте запрос на сортировку результатов сдачи экзаменов в таблице «Учеба» по факультетам (по полю «факультет» по возрастанию), а в пределах факультета по специальностям (по полю «специальность» по убыванию):

- 1) перенесите все поля в структуру запроса, как в предыдущем примере;
- 2) в структуре запроса поменяйте местами поля «факультет» (первое поле) и «специальность» (второе поле);
- 3) в поле «факультет» в строке «Сортировка» мышью включите список способов сортировки и выберите «По возрастанию» (по алфавиту);
- 4) в поле «специальность» в строке «Сортировка» мышью включите список способов сортировки и выберите «По убыванию»;
- 5) завершите результаты формирования запроса;
- 6) сохраните запрос под именем «Сортировка».

### ***Выборка с группированием данных и вычислением функций агрегации***

Для просмотра интегральных характеристик данных по подмножествам записей удобно выполнять группирование данных. Для вычисления итоговых по группе характеристик (количество, сумма, среднее, минимальное, максимальное значения и др.) применяются *групповые функции*. Группировка данных может выполняться по нескольким полям, образуя группы и подгруппы. Так же как и в случае сортировки, порядок следования полей определяет порядок формирования групп и подгрупп. Особенность запроса на группировку данных состоит в том, что в структуре запроса можно использовать только те поля, по которым выполняется группировка, а также вычисляемые поля с функциями агрегации.

**Задание 18.** Допустим, что для каждого студента из БД УЧЕБА, представленной одной таблицей «Учёба», требуется подсчитать количество сданных экзаменов и средний балл по результатам сдачи всех экзаменов. Основанием для группирования является поле «номер студента» и «фамилия» (если мы хотим увидеть это поле в структуре запроса). Группировка по полю «фамилия» будет фиктивной, поскольку с каждым номером студента связана только одна фамилия. Вычисление групповых характеристик (количество и среднее значение) будет выполняться по полю «оценка».

Сформируйте запрос на выборку с группированием данных и вычислением групповых характеристик:

- 1) перенесите в структуру запроса поля «номер студента», «фамилия», «оценка», «оценка»;
- 2) выполните команду **Вид | Групповые операции** – в структуре запроса появится строка «Групповая операция». Группирование данных будет выполняться по полю «номер студента»; поле «фамилия» не создаст вложенной группы, так как уникальной характеристикой студента является его учетный номер. Поэтому значением групповой операции для этих полей будет значение по умолчанию «Группировка»;
- 3) в строке «Групповая операция» для первого поля «оценка» щелкните мышью и выберите из списка функцию подсчета количества Count, для второго поля – функцию вычисления среднего значения Avg (рис. 19);
- 4) в таблице результата запроса полям, по которым вычисляются групповые характеристики, присвойте содержательные имена;
- 5) установите курсор в ячейку «Поле» и нажмите правую клавишу мыши;
- 6) в контекстном меню выполните команду **Свойства** и введите в диалоговом окне «Свойства поля» в строку «Подпись» имена вычисляемых полей «количество экзаменов» и «средний балл». Результаты запроса будут иметь вид, как на рис. 20;
- 7) завершите результаты формирования запроса;

8) сохраните запрос под именем «Группировка».

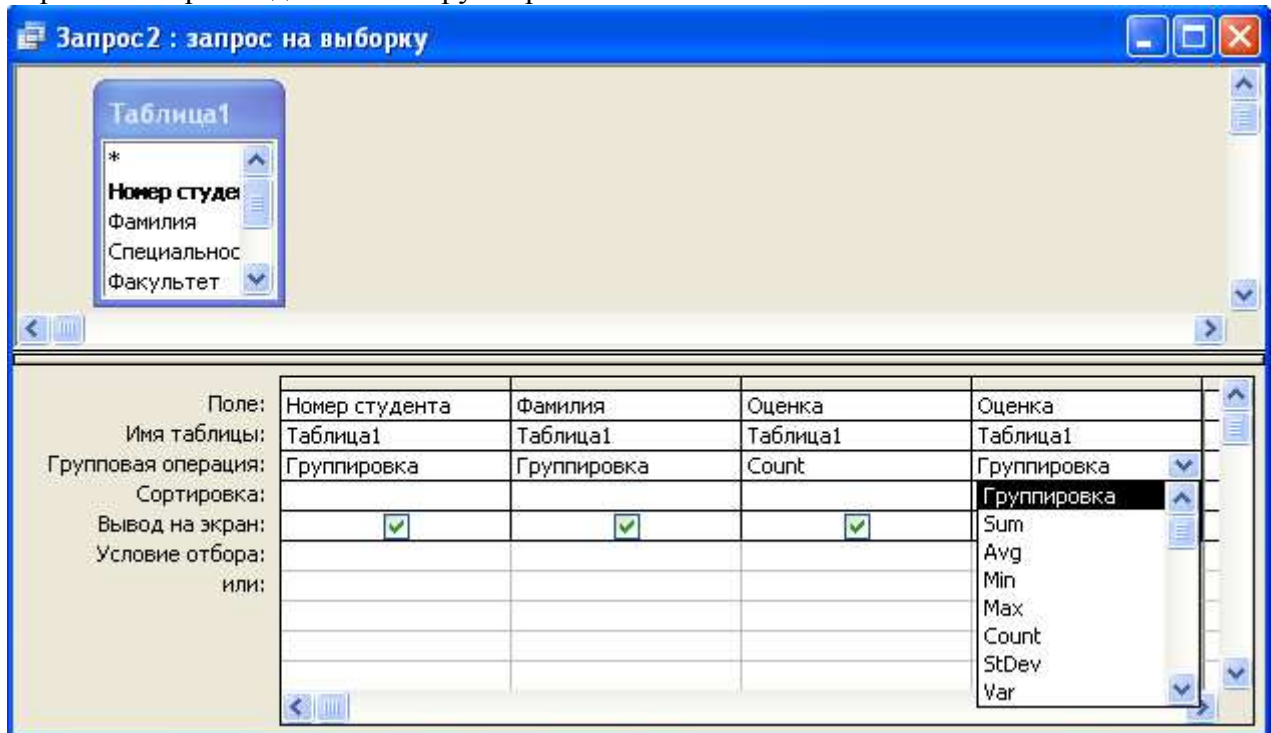


Рис. 19. Структура запроса на группирование данных

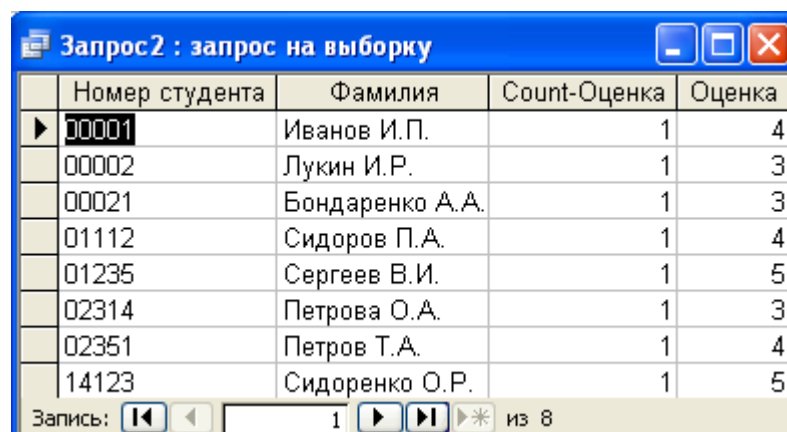


Рис. 20. Результаты запроса на группирование данных

### Лабораторная работа Закрепление изученного материала

Задания для самостоятельной работы

В предлагаемых ниже вариантах задач выполнить следующие задания:

1. Создать базу данных в соответствии с вариантом.
2. Определить ключевые поля.
2. Заполнить таблицу данными. Количество записей должно быть не менее 20.
3. Продумать и выполнить следующие типы запросов-выборок:
  - 1) **проекция** на подмножество однородных атрибутов;
  - 2) **параметрический запрос** на примере **селекции** по условию на основе операций сравнения (равно, не равно, больше, меньше, больше или равно, меньше или равно);
  - 3) запрос-выборка, селекция по сложному условию на основе оператора конъюнкции **And**;
  - 4) запрос-выборка, селекция по сложному условию на основе оператора дизъюнкции **Or**;
  - 5) запрос-выборка, селекция по условию на основе оператора **Between...And**;

- б) запрос-выборка, селекция по условию на основе оператора **Like**;
- 7) запрос-выборка, селекция по условию на основе оператора **In**;
- 8) запрос-выборка, содержащий **вычисляемое поле**;
- 9) запрос-выборка с **сортировкой** по двум-трем полям;
- 10) запрос-выборка с **группировкой** данных и вычислением статистических функций **Count, Sum, Avg, Min, Max** и др.

Самостоятельные упражнения выполнить на основе одного из нижеследующих примеров. Номер варианта выбирается по последней цифре в зачетке. При создании таблиц описание свойств полей задать самостоятельно. Предусмотреть соответствующие маски ввода.

По данной лабораторной работе сформировать отчет с подробным описанием всех действий.

Вариант I. База данных **Поставки** (номер поставщика, фамилия поставщика, город, код детали, город, код детали, количество за месяц).

Вариант II. База данных **Больница** (номер пациента, фамилия пациента, адрес, номер хирурга, фамилия хирурга, дата операции, тип операции).

Вариант III. База данных **Библиотека** (номер абонента, фамилия абонента, номер книги, автор, название, год издания, дата выдачи, дата возврата).

Вариант IV. База данных **Аренда** (номер клиента, фамилия клиента, номер объекта, плата, фамилия собственника, дата начала, дата конца).

Вариант V. База данных **Кадры** (номер сотрудника, фамилия сотрудника, дата рождения, место работы, должность, прием, увольнение, учебное заведение, образование)

### **Лабораторная работа Проблемы манипулирования данными и запросы на создание таблиц**

#### **Проблемы выполнения корректирующих запросов**

Корректирующие запросы на добавление, обновление или удаление данных приводят к изменению содержания базы данных. Если они выполняются некорректно, то в БД будет храниться противоречивая информация.

При выполнении корректирующих запросов могут встретиться проблемы, которые являются следствием недостатков структуры данных «универсального» отношения. Это так называемые аномалии добавления, обновления и удаления. Такими дефектами обладает и таблица «Учеба».

Например, нельзя ввести в таблицу данные о зачисленном в вуз студенте до тех пор, пока он не сдаст первый экзамен и не будут определены ключевые атрибуты «предмет» и «семестр». Это пример *аномалии добавления*. Другой пример: при изменении фамилии студента необходимо обновить поле «фамилия» не один, а много раз во всех строках-записях, содержащих информацию о результатах сдачи экзаменов этим студентом. При этом есть вероятность ошибочного ввода в разные строки разных данных. Это пример *аномалии обновления*. А при удалении единственной строки, содержащей информацию о сдаче экзаменов студентом, приведет к потере информации и о самом студенте. Это пример *аномалии удаления*.

Аномалии манипулирования данными являются следствием избыточного дублирования данных, присущего «универсальному» отношению. Дело в том, что в «универсальном» отношении присутствуют семантически разнородные данные и попытки манипулирования данными одного рода влекут за собой проблемы в отношении данных другого рода.

Решением проблемы аномалии манипулирования данными является корректная декомпозиция «универсального» отношения на его проекции (подмножества, содержащие семантически однородные данные).

В реляционной алгебре операция построения из одной таблицы- отношения другой, являющейся её подмножеством, называется *проекцией* исходной таблицы на подмножество её столбцов-атрибутов.

### Запрос на создание таблицы

Запрос на создание новой таблицы как проекции исходной рассмотрим в контексте проблемы декомпозиции «универсального» отношения. Выполним декомпозицию «универсального» отношения со схемой:

**Учёба (номер студента, фамилия, специальность, факультет, предмет, семестр, оценка)**

на ее проекции со схемами:

**Кафедра (специальность, факультет)**

**Экзамен (номер студента, фамилия, факультет, предмет, семестр, оценка)**

Рассмотрим, как в Access реализуется операция проекция–создание одной таблицы как проекции другой. Алгоритм выполнения этой процедуры состоит из трёх этапов:

1. Построение запроса-выборки столбцов-атрибутов без дублей строк (Задание 6 на стр. 21]. Тем самым создается проекция – подмножество данных исходной таблицы, которое определит структуру и наполнение новой таблицы.
2. Выполнение запроса на создание новой таблицы.
3. Определение ключа вновь созданной таблицы .

**Задание 1.** На основе исходной таблицы «Учёба» со схемой отношения:

**Учёба (номер студента, фамилия, специальность, факультет, предмет, семестр, оценка)**

создайте две новые таблицы со схемами:

**Кафедра (специальность, факультет)**

**Экзамен (номер студента, фамилия, факультет, предмет, семестр, оценка)**

Алгоритм построения и выполнения запроса на создание новой таблицы рассмотрим на примере отношения «Кафедра»:

- 1) в режиме **Конструктора** запросов в качестве таблицы-источника данных к запросу выберите таблицу «Учёба»;
- 2) сформируйте запрос-выборку, в котором присутствовали бы следующие поля таблицы «Кафедра»: **специальность, факультет**;
- 3) в режиме **Конструктора** выполните команду **Вид | Свойства**;
- 4) в диалоговом окне «Свойства запроса» переключите мышью свойство «Уникальные значения» в состояние «Да»;
- 5) завершите результаты формирования запроса (на этом первая часть алгоритма заканчивается);
- 6) в режиме **Конструктора** выполните команду **Запрос / Создание таблицы**;
- 7) в диалоговом окне «Свойства запроса» введите в поле «Таблица» имя новой таблицы «**Кафедра**» (рис. 21);
- 8) в режиме **Конструктора** выполните команду **Запрос | Запуск**. В новую таблицу будет скопировано соответствующее количество записей
- 9) перейдите в режим **Конструктора** и завершите работу с запросом (можно без сохранения структуры запроса, так как результатом запроса является созданная таблица). На этом заканчивается вторая часть алгоритма;

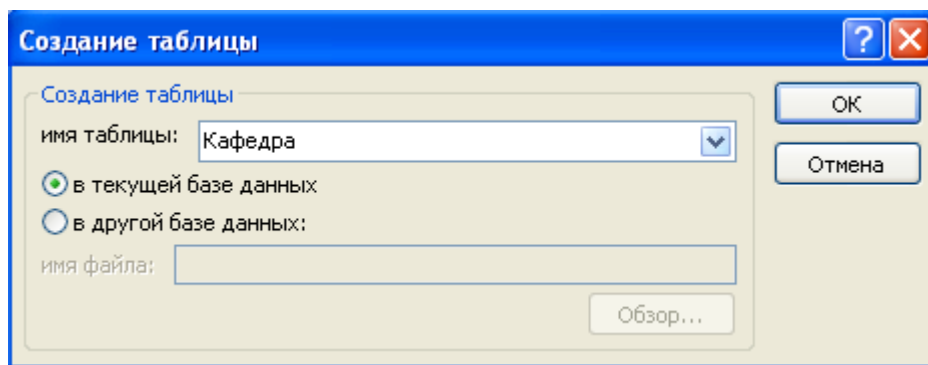


Рис. 21. Окно наименования новой таблицы

- 10) в окне БД выберите закладку **Таблицы** и убедитесь, что новая таблица «Кафедра» представлена в списке таблиц БД; это является признаком успешного выполнения запроса на создание таблицы;
- 11) выберите новую таблицу «Кафедра» в списке и нажмите экранную кнопку;
- 12) в режиме **Конструктора** таблиц в окне структуры таблицы определите ключевые атрибуты (в таблице «Кафедра» это поле «специальность»);
- 13) сохраните результаты спецификации новой таблицы «Кафедра» по команде **Файл | Сохранить**. На этом заканчивается третья часть алгоритма;
- 14) просмотрите в **Режиме таблицы** наполнение новой таблицы «Кафедра» (рис. 22);
- 15) по рассмотренному алгоритму создайте вторую таблицу «Экзамен».

Специальность	Факультет
Бухгалтерский учет	Экономический
Информатика	Технологический
Маркетинг	Экономический
Менеджмент организации	Экономический
Электроснабжение	Технологический

Рис. 22 Наполнение таблицы «Кафедра»

### Создание связей между таблицами

В результате декомпозиции «универсального» отношения «Учеба» мы получили представление той же самой предметной области в виде совокупности двух отношений – таблиц «Кафедра» и «Экзамен». Теперь установим связь между новыми таблицами.

Правила установления связей между таблицами в Access таковы, что разрешены только *бинарные* (между двумя таблицами) связи. Связи могут быть двух типов «один к одному» и «один ко многим». Причём одна таблица выступает в роли *старшей* (отцовской), например таблица «Кафедра» в отношении «Кафедра» : «Экзамен», а другая – в роли младшей (сыновьей), например таблица «Экзамен» в отношении «Кафедра» : «Экзамен». Однако одна и та же таблица в разных отношениях может выступать в роли старшей и младшей одновременно. Связь между таблицами устанавливается по общим полям-атрибутам (поле «факультет» в нашем случае). Связь между таблицами Access устанавливается при помощи инструмента, называемого «**Схемой данных**».

**Задание 2.** Для установления связи между таблицами «Кафедра» и «Экзамен»:

- 1) выполните команду **Сервис | Схема данных** – появится окно макета схемы данных БД;
- 2) в контекстном меню выполните команду **Связи / Добавить таблицу**;
- 3) в окне «Добавление таблицы» выберите в списке таблицу «Кафедра» и нажмите экранную кнопку **Добавить**;

4) в окне «Добавление таблицы» выберите в списке таблицу «Экзамен» и нажмите экранную кнопку **Добавить**;

5) для установления связи между таблицами «Кафедра» и «Экзамен» захватите мышью и перенесите поле «факультет» таблицы «Кафедра» на поле «факультет» таблицы «Экзамен» – раскроется окно «Изменение связей» (рис. 23);

б) не настраивая свойства связи, нажмите экранную кнопку.

Схема базы данных примет вид, аналогичный представленному на рис. 24.

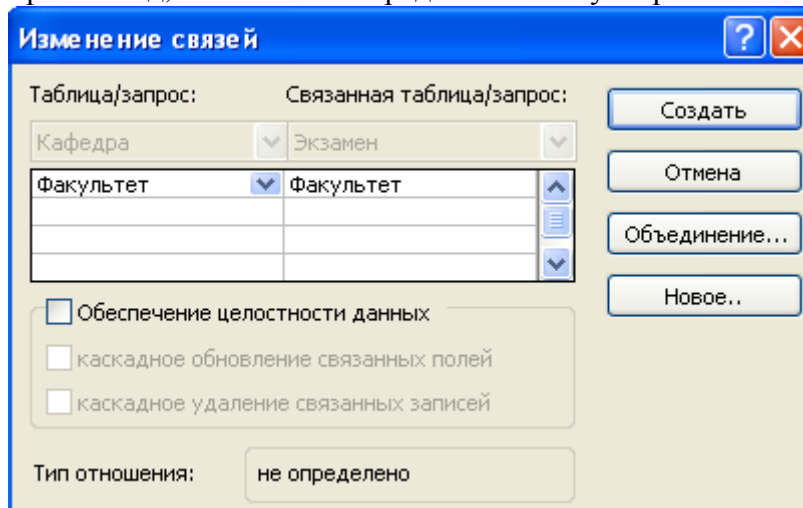


Рис. 23. Окно спецификации связей в схеме данных

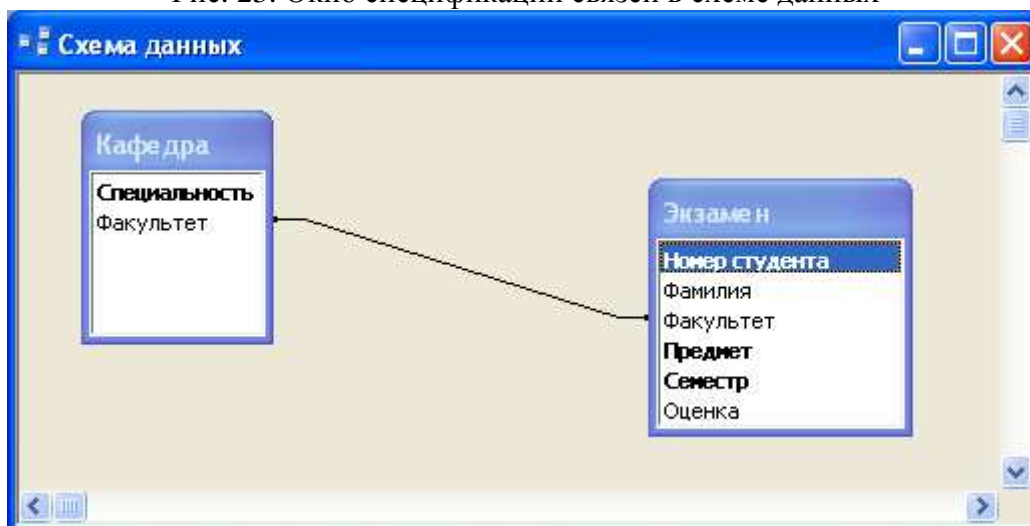


Рис. 24. Схема данных двух таблиц

## ДЕКОМПОЗИЦИЯ ОТНОШЕНИЙ И ЗАПРОСЫ НА СОЕДИНЕНИЕ ТАБЛИЦ

### Корректность декомпозиции отношений

Итак, структура информации в БД одной и той же предметной области (ПО) УЧЕБА может быть представлена двумя способами:

1) в виде схемы одного так называемого «универсального» отношения (в нашем примере в виде таблицы «Учёба»);

2) в виде комбинации схем нескольких отношений (в нашем примере в виде таблиц «Кафедра» и «Экзамен»).

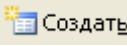
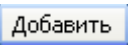
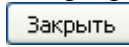
Эквивалентность того и другого представления обеспечивается правилами корректной декомпозиции, при которой отсутствуют искажения хранимых данных и сохраняются все избыточные функциональные зависимости между атрибутами. Корректность декомпозиции можно проверить, выполнив так называемую операцию *естественного соединения* компонентов

декомпозиции. Декомпозиция считается корректной с точки зрения сохранения данных, если при помощи естественного соединения компонентов декомпозиции можно восстановить исходное отношение.

### Запрос на соединение таблиц

Естественное соединение – это реляционная операция, которая порождает итоговую таблицу как соединение исходных таблиц по условию равенства одноименных (одноименных) полей-атрибутов без дублей одноименных столбцов [1–5].

**Задание 3.** Создайте запрос на соединение двух независимых таблиц «Кафедра» и «Экзамен»:

- 1) в окне БД выберите закладку **Запрос** и нажмите экранную кнопку  ;
- 2) в окне «Новый запрос» выберите вариант **Конструктор**;
- 3) в окне «Добавление таблицы» выберите по закладке источник данных к запросу: **Таблицы / Запросы | Таблицы и Запросы**; в данном случае это **Таблицы**;
- 4) выберите в списке таблицы «Кафедра» и «Экзамен» и нажмите экранную кнопку  ;
- 5) нажмите экранную кнопку  – появится основное окно конструктора запросов в стиле QBE (рис. 25). Обратите внимание на связи между одноименными атрибутами разных таблиц в верхней части бланка запроса;
- 6) перенесите в структуру запроса в нижней части из разных таблиц **разноименные поля** (рис. 25);
- 7) выполните запрос по команде **Запрос / Запуск** – результат запроса будет выглядеть, как на рис. 26;
- 8) завершите результаты формирования запроса так, как это обсуждалось ранее;
- 9) сохраните результат под именем «Естественное соединение».

Обратите внимание на важный результат: итоговая таблица (рис. 26) по запросу на соединение двух таблиц не совпадает с исходной таблицей «Учеба» по числу строк. В ней присутствуют лишние строки, которые содержат семантически некорректные данные, например о том, что студенты учатся по двум специальностям. Из полученного результата следует важный вывод: декомпозиция таблицы-отношения «Учёба» выполнена некорректно.

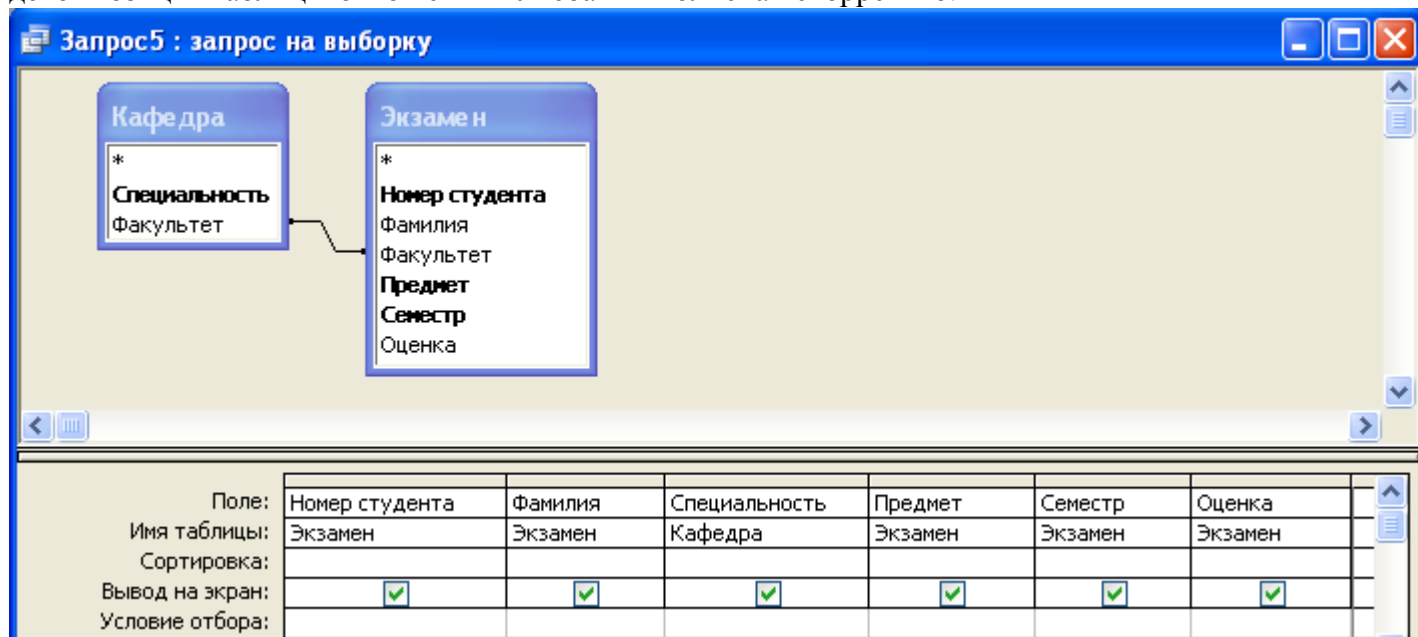


Рис. 25. Структура запроса на соединение таблиц «Кафедра» и «Экзамен»



Запрос5 : запрос на выборку

Номер студент	Фамилия	Специальность	Предмет	Семестр	Оценка
01112	Сидоров П.А.	Электроснабжение	Программиров	3	4
02314	Петрова О.А.	Информатика	Электротехник	2	3
02314	Петрова О.А.	Электроснабжение	Электротехник	2	3
14123	Сидоренко О.Ф	Информатика	Информатика	1	5
14123	Сидоренко О.Ф	Электроснабжение	Информатика	1	5
00001	Иванов И.П.	Бухгалтерский учет	Информатика	1	4
00001	Иванов И.П.	Маркетинг	Информатика	1	4
00001	Иванов И.П.	Менеджмент организации	Информатика	1	4
00002	Лукин И.Р.	Бухгалтерский учет	История	3	3
00002	Лукин И.Р.	Маркетинг	История	3	3
00002	Лукин И.Р.	Менеджмент организации	История	3	3
00021	Бондаренко А.	Бухгалтерский учет	Философия	2	3

Запись: 12 из 21

Рис. 6. Результат запроса на соединение таблиц «Кафедра» и «Экзамен»

### Корректная декомпозиция

Декомпозиция исходного отношения (таблицы) на его проекции считается корректной, если выполняются два условия:

1. Сохранение исходных данных в результате естественного соединения компонентов декомпозиции.
2. Сохранение избыточных функциональных зависимостей между атрибутами исходного отношения.

Из теории БД, занимающейся проблемой оптимизации структуры данных, следует, что для реализации корректирующих запросов на добавление, удаление и изменение данных предпочтительнее представление БД в виде не одной таблицы («универсального» отношения), а в виде совокупности так называемых *нормализованных таблиц-отношений*. Проблемы нормализации отношений здесь не обсуждаются. Достаточно понять, что в основе нормализации таблиц-отношений лежит процедура декомпозиции ненормализованного «универсального» отношения на её нормализованные компоненты, которые объединяют семантически однородные данные.

**Задание 4.** Выполните декомпозицию «универсального» отношения со схемой:

**Учёба** ( номер студента, фамилия, специальность, факультет, предмет, семестр, оценка)  
на ее проекции со схемами:

**Кафедра** (специальность, факультет)

**Студент** (номер студента, фамилия, специальность)

**Сессия** (номер студента, предмет, семестр, оценка)

Таблица «Кафедра» уже создана, поэтому:

- 1) создайте остальные таблицы по алгоритму, описанному на стр. 26;
- 2) установите связи между таблицами так, как обсуждалось на стр. 28;
- 3) проверьте корректность декомпозиции по новой схеме как результат операции естественного соединения трех таблиц (см. стр. 29).

Итоговые таблицы «Студент» и «Сессия» должны выглядеть так, как показано на рис. 27 и 28.

Схема данных нового варианта БД на основе трех таблиц должна выглядеть, как на рис. 29.

Структура запроса на естественное соединение таблиц «Кафедра», «Студент» и «Сессия» представлена на рис. 30.

Результат запроса на естественное соединение таблиц «Кафедра», «Студент» и «Сессия» приведен на рис. 31.

Студент : таблица			
номер студента	фамилия	специальность	
016510	Иванов И.И.	с2	
021008	Иванов И.И.	с1	
028101	Андреев А.А.	с3	
029812	Петров П.П.	с1	
031512	Николаева Н.Н.	с2	
*			

Запись: 2 из 5

Рис. 27. Наполнение таблицы «Студент»

Сессия : таблица				
номер студента	предмет	семестр	оценка	
016510	экономика	9	3	
021008	математика	2	4	
021008	физика	2	4	
028101	программиров	5	3	
029812	кибернетика	4	4	
031512	математика	1	5	
031512	математика	2	5	
031512	физика	1	5	
*				

Запись: 1 из 8

Рис. 28. Наполнение таблицы «Сессия»

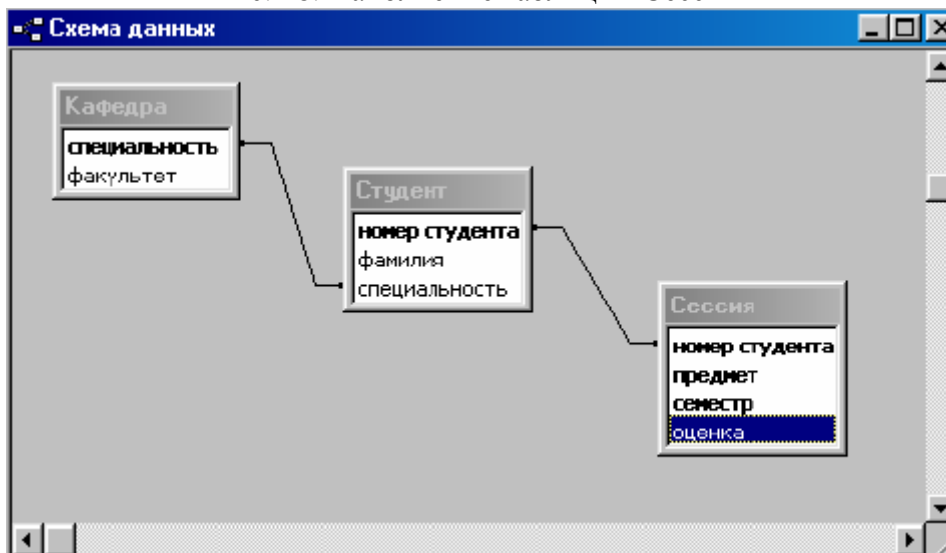


Рис. 29. Схема данных трех таблиц

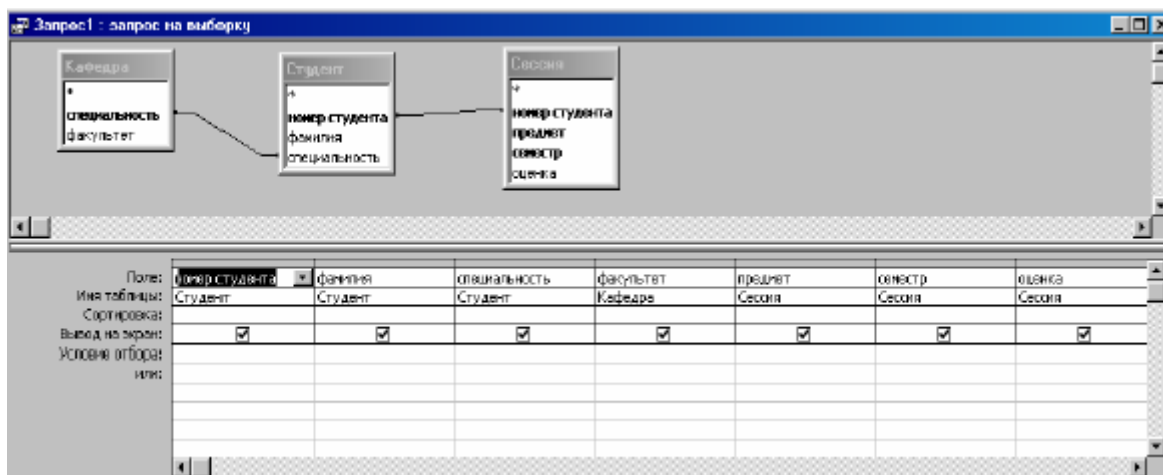


Рис. 30. Структура запроса на соединение трех таблиц

номер студента	фамилия	специальность	факультет	предмет	семестр	оценка
016510	Иванов И.И.	с2	ф2	экономика	9	3
021008	Иванов И.И.	с1	ф1	математика	2	4
021008	Иванов И.И.	с1	ф1	физика	2	4
028101	Андреев А.А.	с3	ф2	программирование	5	3
029812	Петров П.П.	с1	ф1	кибернетика	4	4
031512	Николаева Н.Н.	с2	ф2	математика	1	5
031512	Николаева Н.Н.	с2	ф2	математика	2	5
031512	Николаева Н.Н.	с2	ф2	физика	1	5
*						

Рис. 31. Результат запроса на естественное соединение трех таблиц

Обратите внимание на важнейший результат: результирующая таблица (рис. 11) по запросу на соединение трех таблиц с точностью до порядка следования строк и столбцов (что не существенно) совпадает с исходной таблицей «Учёба» в [6, стр. 18], из которой новые таблицы были получены как её проекции.

Из полученного результата следуют важные выводы:

1. Декомпозиция таблицы-отношения «Учёба» на её компоненты- проекции «Кафедра», «Студент» и «Сессия» выполнена корректно, т. е. Без потери данных.
2. Представления предметной области (ПО) «Учёба» в виде одноименного «универсального» отношения и в виде комбинации нормализованных отношений- таблиц «Кафедра», «Студент» и «Сессия» эквивалентны. Отныне наша база данных УЧЕБА будет представлена не одной таблицей «Учеба», а комбинацией нормализованных связанных таблиц «Кафедра», «Студент» и «Сессия». Все простые запросы-выборки, которые были выполнены ранее на примере одной таблицы «Учеба», теперь с равным успехом могут быть воспроизведены на основе связанных таблиц. Рассмотрим в качестве примера запрос на группировку данных.

**Задание 5.** Допустим, что для каждого студента из БД УЧЕБА, представленной теперь тремя таблицами, требуется подсчитать количество сданных экзаменов и средний балл по результатам сдачи всех экзаменов.

Сформируйте запрос на выборку с группированием данных из связанных таблиц «Студент» и «Сессия» и вычислением групповых характеристик:

- 1) в окне БД на закладке **Запрос** приступите к созданию нового запроса (нажмите экранную кнопку);

- 2) в окне «Новый запрос» выберите вариант **Конструктор**;
- 3) в окне «Добавление таблицы» по закладке **Таблицы** выделите в списке таблицы «Студент» и «Сессия» и нажмите экранные кнопки и;
- 4) в окне конструктора запросов выберите в структуру запроса поля «номер студента», «фамилия» (из таблицы «Студент»), «оценка», «оценка» (из таблицы «Сессия»);
- 5) выполните команду **Вид | Групповые операции**; в структуре запроса в строке «Групповая операция» для полей «номер студента» и «фамилия» включите настройку «Группировка»;
- 6) для первого поля «оценка» в строке «Групповая операция» выберите из списка функцию подсчета количества Count, для второго поля – функцию вычисления среднего значения Avg (рис. 32);
- 7) в таблице результата запроса полям, по которым вычисляются групповые характеристики, присвойте содержательные имена «количество экзаменов» и «средний балл»;
- 8) результаты запроса будут иметь вид, как на рис. 33;
- 9) сохраните запрос под именем «Группировка2»

номер студента	фамилия	количество экзаменов	средний балл
016510	Иванов И.И.	1	3
021008	Иванов И.И.	2	4
028101	Андреев А.А.	1	3
029812	Петров П.П.	1	4
031512	Николаева Н.Н.	3	5

Рис. 32. Структура запроса на группирование данных из связанных таблиц

Поле:	номер студента	фамилия	оценка	оценка
Имя таблицы:	Студент	Студент	Сессия	Сессия
Групповая операция:	Группировка	Группировка	Count	Avg
Сортировка:				
Вывод на экран:	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>
Условие отбора:				
или:				

Рис. 33. Результаты запроса на группирование данных из связанных таблиц

Сравните полученные результаты с результатами аналогичного запроса к одной таблице. Совпадение очевидно.

Однако следует заметить, что выборка данных из связанных таблиц, особенно если их много и они содержат большое количество строк, требует для своего выполнения больше компьютерного времени, чем выборка из одной таблицы.

Таким образом, для запросов-выборок желательно иметь все данные в одном месте, в одной таблице. Что же касается корректирующих запросов, то там ситуация прямо противоположная. Эффективность корректирующих запросов и качество структуры базы данных требуют представление БД в виде совокупности связанных таблиц. Решение проблемы требует взвешенного подхода с учетом всех «за» и «против».

## КОРРЕКТИРУЮЩИЕ ЗАПРОСЫ

### Выполнение корректирующих запросов

В результате корректной декомпозиции отношения «Учеба» мы развели семантически однородные данные по разным таблицам. Тем самым мы решили проблему аномалий манипулирования данными и создали предпосылки для корректного выполнения корректирующих запросов. Корректирующие запросы бывают трех типов:

- 1) запросы на добавление данных;
- 2) запросы на удаление данных;
- 3) запросы на обновление данных.

Алгоритм выполнения этих запросов состоит из двух этапов [7–10]:

1. Построение запроса-выборки строк-записей, подлежащих добавлению, удалению или обновлению. Тем самым создается подмножество корректируемых данных.
2. Выполнение корректирующего запроса.

Продемонстрируем эти типы запросов на примере таблицы «Студент».

### **Запрос на удаление**

Задание 6. Создайте запрос на удаление из таблицы «Студент» данных о студенте, заданном его фамилией.

- 1) в качестве таблицы-источника данных к запросу выберите таблицу «Студент»;
- 2) сформируйте запрос-выборку всех полей по условию, задаваемому параметрически; в поле «фамилия» в строке «Условие отбора» наберите в квадратных скобках текст подсказки к вводу конкретного значения [Фамилия студента];
- 3) выполните запрос по команде **Запрос | Запуск**; в диалоговом окне, которое появится при реализации запроса, введите в кавычках конкретное значение, например «**Андреев А.А.**»;
- 4) просмотрите результаты запроса в **Режиме таблицы** и примите решение, удовлетворяют ли вас результаты выборки или нет;
- 5) вернитесь в режим **Конструктора**. На этом первая часть алгоритма заканчивается; выборка данных на удаление подготовлена;
- 6) в режиме **Конструктора** выполните команду **Запрос | Удаление** – в бланке структуры запроса появится строка «Удаление» (рис. 34);

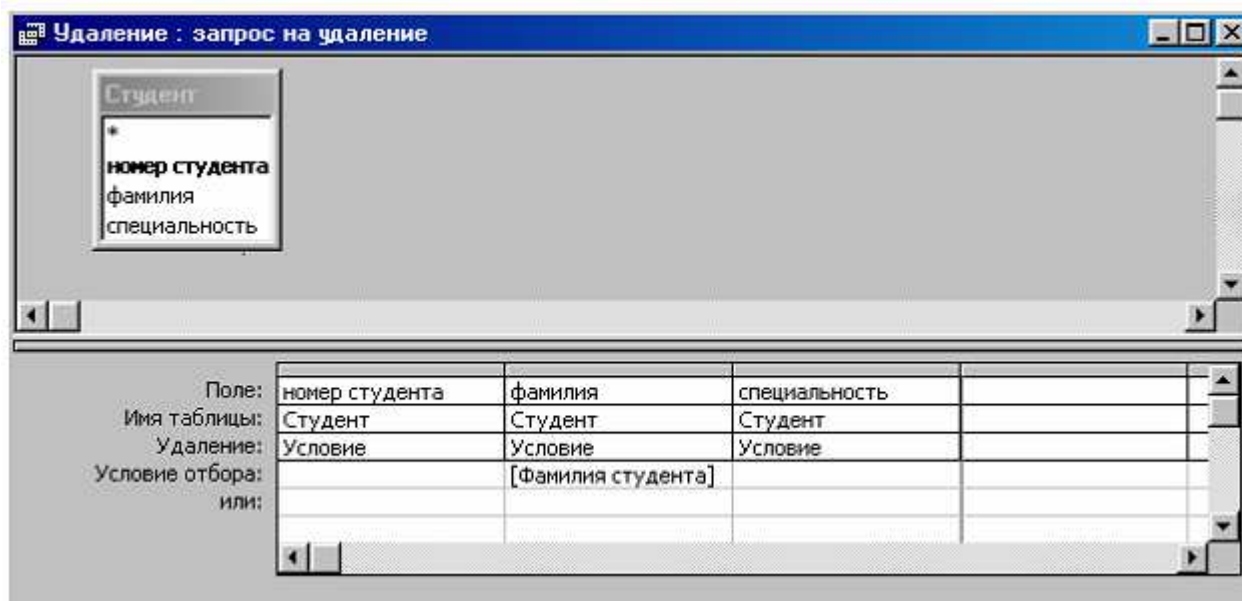


Рис. 34. Структура запроса на удаление данных из таблицы

- 7) выполните запрос по команде **Запрос | Запуск**;
- 8) завершите результаты формирования запроса. На этом заканчивается вторая часть алгоритма;
- 9) в окне БД на закладке Таблицы выберите таблицу «Студент» и посмотрите её наполнение. Убедитесь, что удаление записи действительно произошло.

### Запрос на добавление

**Задание 7.** Речь идет о добавлении данных в одну таблицу из другой. Создайте запрос на добавление данных о студенте, заданном его фамилией из таблицы «Учёба» в таблицу «Студент»:

- 1) в качестве таблицы-источника данных выберите таблицу «Учёба»;
  - 2) сформируйте запрос-выборку полей, которые должны быть добавлены («номер студента», «фамилия», «специальность») по условию, задаваемому параметрически; в поле «фамилия» в строке «Условие отбора» наберите в квадратных скобках текст подсказки к вводу конкретного значения [Фамилия студента];
  - 3) чтобы избавиться от дублей строк в выборке для добавления в таблицу «Студент» дополните условие выборки требованием «без дублей записей». Для этого в режиме **Конструктора** выполните команду **Вид | Свойства**;
  - 4) в диалоговом окне «Свойства запроса» переключите мышью свойство «Уникальные значения» в состояние «Да»;
- выполните запрос по команде **Запрос | Запуск**; в диалоговом окне, которое появится при реализации запроса, введите в кавычках конкретное значение, например «Андреев А.А.»;
- просмотрите результаты запроса в **Режиме таблицы** и примите решение, удовлетворяют ли вас результаты выборки или нет;

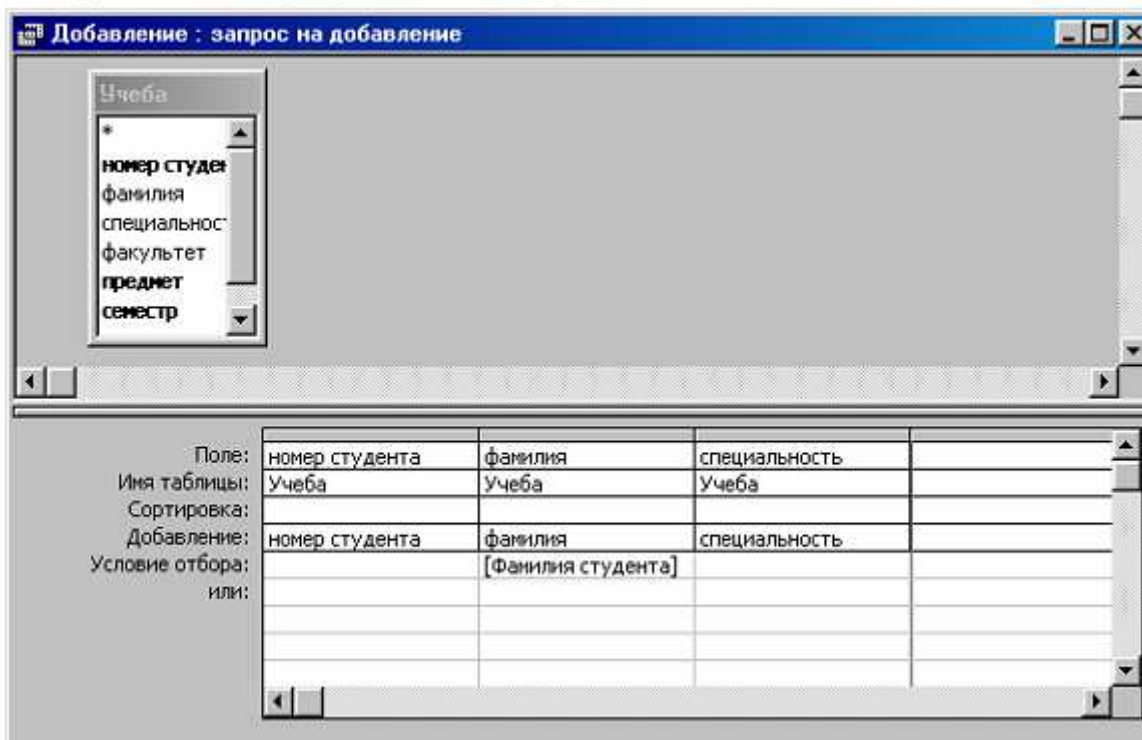


Рис. 15. Структура запроса на добавление данных в таблицу

- 5) вернитесь в режим **Конструктора**. На этом первая часть алгоритма заканчивается; выборка данных на добавление подготовлена.
- 6) в режиме Конструктора выполните команду **Запрос | Добавление**;
- 7) в диалоговом окне «**Добавление**» в поле «имя таблицы» выберите из списка таблицу – приёмник данных «**Студент**»;
- 8) в бланке структуры запроса появится строка «Добавление» (рис. 15);
- 9) выполните запрос по команде **Запрос | Запуск**;
- 10) завершите результаты формирования запроса. На этом заканчивается вторая часть алгоритма;
- 11) в окне БД на закладке Таблицы выберите таблицу «**Студент**» и убедитесь, что добавление записи действительно произошло.

### Запрос на обновление

**Задание 8.** Создайте запрос на замену в таблице «Студент», например, фамилии «Николаева Н.Н.» на «Сергеева Н.Н.»:

- 1) в качестве таблицы-источника данных выберите таблицу «**Студент**»;
- 2) сформируйте параметрический запрос-выборку всех полей по условию «фамилия» = [Старая фамилия]: в поле «фамилия» в строке «Условие отбора» наберите [Старая фамилия];
- 3) выполните запрос по команде **Запрос | Запуск**;
- 4) в диалоге задайте в кавычках старую фамилию, например «**Николаева Н.Н.**»
- 5) просмотрите результаты запроса в Режиме таблицы и примите решение, удовлетворяют ли вас результаты выборки или нет;
- 6) вернитесь в режим **Конструктора**. На этом первая часть алгоритма заканчивается; выборка данных на обновление подготовлена.
- 7) в режиме **Конструктора** выполните команду **Запрос | Обновление**; в бланке структуры запроса появится строка «Обновление» (рис. 36);
- 8) в строку «Обновление» в поле «фамилия» введите подсказку для ввода [Новая фамилия];
- 9) выполните запрос по команде **Запрос | Запуск**;
- 10) в диалоге задайте новую фамилию, например «**Сергеева Н.Н.**»;

- 11) завершите результаты формирования запроса. На этом заканчивается вторая часть алгоритма;
- 12) в окне БД на закладке Таблицы выберите таблицу «Студент» и убедитесь, что обновление записи действительно произошло.

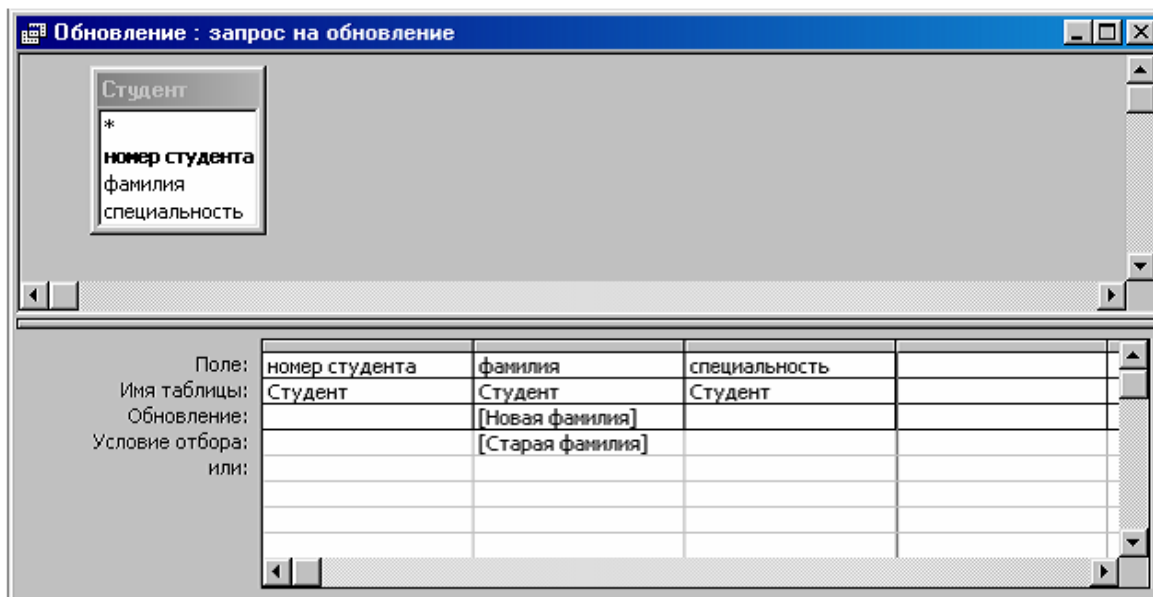


Рис.

Рис. 36. Структура запроса на обновление данных

## ОБЕСПЕЧЕНИЕ ЦЕЛОСТНОСТИ ДАННЫХ

### Ограничения целостности данных

В отношении базы данных важно хорошо уяснить себе следующее. База данных – это не просто архив, хранилище полезной информации. База данных – главным образом информационная модель ПО, для которой она создана.

Поэтому важнейшим условием ее успешной эксплуатации является адекватность хранимых в ней данных актуальному состоянию ПО.

Целостность БД – это синоним её системности. Для успешного исследования ПО должна рассматриваться как система. Точно так же БД – её информационная модель – должна проектироваться как системный объект. Это значит, что структура БД определяется не только составом образующих её информационных элементов, но и характером связей между ними. Связи соответствуют зависимостям между компонентами ПО. Всякие зависимости представляют собой ограничения на возможные отношения элементов системы. В теории БД эти ограничения называются ограничениями целостности данных.

При проектировании БД особое значение имеют так называемые семантические ограничения целостности (или семантические условия).

Семантические ограничения присущи самой ПО и их учёт основан на анализе смыслового содержания данных. Семантические ограничения целостности не выводятся. Это свойства данных, которые выполняются или не выполняются для рассматриваемого отношения элементов данных.

Семантические ограничения целостности различают:

- 1) ограничения целостности атрибутов (доменов);
- 2) ограничения целостности отношений (таблиц);
- 3) ограничения целостности ссылок между отношениями (ограничения целостности базы данных).

Ограничением целостности атрибута является определение множества элементов данных – домена, на котором определен атрибут. Ограничения такого рода задаются при определении структуры таблицы в виде свойств образующих ее полей, таких как тип данных, размер поля, маска ввода, значение по умолчанию, условие на значение и др. Например, понятие «оценка» предполагает, что это данное «числового» и никакого другого типа со значениями в



диапазоне от 2 до 5 или что «фамилия» – это строка символов определенной длины, содержащая фамилию и инициалы и т. д.

Ограничение целостности атрибутов проверяется СУБД всегда и притом немедленно. Любая попытка введения некорректного значения атрибута будет отвергнута.

Ограничения целостности отношения базируются на понятиях функциональных зависимостей атрибутов, потенциальных ключей, неопределенных так называемых Null-значений атрибутов и др.

Зависимости одних атрибутов от других являются очень распространенным типом ограничений. Особое значение имеют зависимости от так называемых потенциальных ключей. Строки-кортежи отношения соответствуют реальным экземплярам объектов, а реальные объекты различимы, идентифицируемы. Это значит, что среди атрибутов отношения всегда найдется такое их подмножество, которое однозначно определяет любой другой атрибут отношения, а значит, и всё отношение в целом. Например, понятие «номер студента» в роли потенциального, первичного ключа как ограничение означает, что не может быть двух и более персон с одинаковыми учётными номерами.

Значение Null используется в реляционной модели при решении проблемы отсутствующей информации. Null – это не одно и то же, что пробелы или числовые нули. Значения Null могут использоваться как значения атрибутов, для которых данные не известны. Null-значение для заданного атрибута может быть разрешено или запрещено. Значение Not Null и означает, что (обязательное поле) является ограничением для ключевых полей ни при каких условиях ключевые атрибуты не могут принимать неопределенные значения.

Ограничения целостности отношений также задаются при спецификации структуры каждой таблицы при определении ее ключевых или индексированных полей. Ограничения целостности отношений также являются безотлагательными, они проверяются всегда и притом немедленно. Любая попытка пропуска значений или ввода дублирующих значений ключевых полей пресекается СУБД немедленно.

В случае, когда база данных моделируется не одним, а несколькими (хотя бы и двумя) связанными отношениями, возникает проблема ограничения целостности базы данных. Ограничение целостности базы данных в реляционном представлении определяется как ограничение ссылочной целостности, или ограничение, основанное на связях между отношениями-объектами.

Связи между отношениями могут быть типа «один к одному» (1:1) или «один ко многим» (1:N).

Связь каждой пары отношений-объектов, в которой одно отношение является «отцом», а другое – «сыном», обеспечивается ключом связи, или внешним ключом.

Внешний ключ – это первичный ключ отношения-отца, переданный «по наследству» отношению-сыну.

Ограничения ссылочной целостности моделируют в структуре данных два рода зависимостей: неидентифицирующие и идентифицирующие. Если внешний ключ используется только для моделирования связи и не входит в состав потенциального ключа отношения-сына, мы имеем пример «слабой», или неидентифицирующей зависимости. Если внешний ключ используется не только для моделирования связи, но и для однозначной идентификации отношения-сына (входит в состав потенциального ключа отношения -сына), мы имеем случай «сильной», или идентифицирующей зависимости.

Роль связей как ограничений целостности проявляется в том, что они налагают ограничения на выполнение корректирующих операций (добавление, удаление и обновление), выполняемых над строками-кортежами связанных таблиц. Корректирующие операции должны выполняться таким образом, чтобы связи между таблицами всегда оставались согласованными и соответствовали семантике связей между объектами ПО.

Ограничения целостности базы данных, как правило, не проверяются немедленно. Проверка откладывается до завершения транзакции –логического элемента работы с базой данных. Это делается для того, чтобы были выполнены обновления различных связанных отношений при переходе из одного целостного состояния в другое.

Примерами ограничений целостности ссылок в нашем случае являются, например такие:

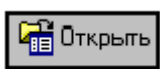
1. Отношение между таблицами «Кафедра» и «Студент» должно быть таково, что студент не может быть принят на несуществующую специальность.
2. Отношение между таблицами «Студент» и «Сессия» должно быть таково, что персона, не зачисленная как студент, не может быть допущена к сдаче экзамена.

### Контроль целостности данных

Ограничения целостности атрибутов и отношений уже были учтены нами при создании таблиц. Рассмотрим, как обстоит дело с ограничениями ссылочной целостности.

**Задание 9.** Для проверки корректности процедур манипулирования данными связанных таблиц попытайтесь добавить вручную в таблицу «Студент» бесконтрольную запись такого содержания: «000000», «Одинцов О.О.», «с0» (речь идет о приёме студента на несуществующую специальность):

- 1) в окне БД на закладке Таблица выберите из списка таблицу «Студент» и нажмите экранную



кнопку ;

- 2) введите вручную новую запись: «000000», «Одинцов О.О.», «с0»;

- 3) сохраните результат по команде **Файл | Сохранить** и **Файл | Заккрыть**.

Вы выполнили некорректную с точки зрения семантики ПО операцию:

зачислили студента на несуществующую специальность. Система БД никак на это не отреагировала. Это плохо. Наша БД не защищена от некорректных действий пользователя. Причина в том, что нет корректных связей между таблицами и ничто не мешает вводить запрещенные данные (например, номер специальности, не зарегистрированной в БД, в таблице «Кафедра», как в нашем примере).

### Семантика связей между таблицами

В нашем примере представлены связи обоих типов. Между таблицами «Кафедра» и «Студент» имеется неидентифицирующая связь. Роль внешнего ключа таблицы «Студент» играет атрибут «специальность». Семантика этой связи такова «каждый студент обучается по определенной специальности, но персона студента однозначно определяется его собственным учетным номером». Между таблицами «Студент» и «Сессия» имеется идентифицирующая связь. Внешним ключом таблицы «Сессия» является атрибут «номер студента». Семантика этой связи такова «результаты сдачи экзамена по данному предмету в данный семестр однозначно определяются учетным номером студента».

СУБД Access замечательна тем, что установка средств контроля целостности данных в связанных таблицах в значительной степени автоматизирована.

Познакомимся с этими возможностями подробнее.

Во-первых, корректность ввода значений внешнего ключа «специальность» в таблице «Студент» может быть обеспечена настройкой свойства подстановка этого поля. Свойство подстановка позволяет связать поле с элементом управления «поле со списком» и вводить данные под управлением списка «разрешенных» значений. Источником «разрешенных» значений для поля «специальность» таблицы «Студент» является отцовская таблица «Кафедра».

**Задание 10.** Для обеспечения контроля ввода корректных значений внешнего ключа «специальность» в таблице «Студент» настройте его свойство подстановка (создайте на основе таблицы «Кафедра» поле со списком разрешенных значений):

- 1) в окне БД на закладке Таблицы откройте таблицу «Студент» в режиме **Конструктор**;
- 2) выделите поле «специальность»;

- 3) в нижней части диалогового окна, на закладке «Подстановка» настройте свойства элемента управления «Поле со списком» в соответствии с образцом, приведенным на рис. 37;
- 4) по завершении модификации структуры таблицы «Студент» сохраните результаты по команде **Файл | Сохранить** и закройте диалоговое окно;
- 5) откройте таблицу «Студент» в Режиме таблицы, для чего нажмите экранную кнопку



- б) в любой строке таблицы щелчком мыши в поле «специальность» вызовите список «разрешенных» значений (рис. 38);
- 7) попробуйте ввести данные в обход элемента управления, убедитесь в тщетности этой попытки.

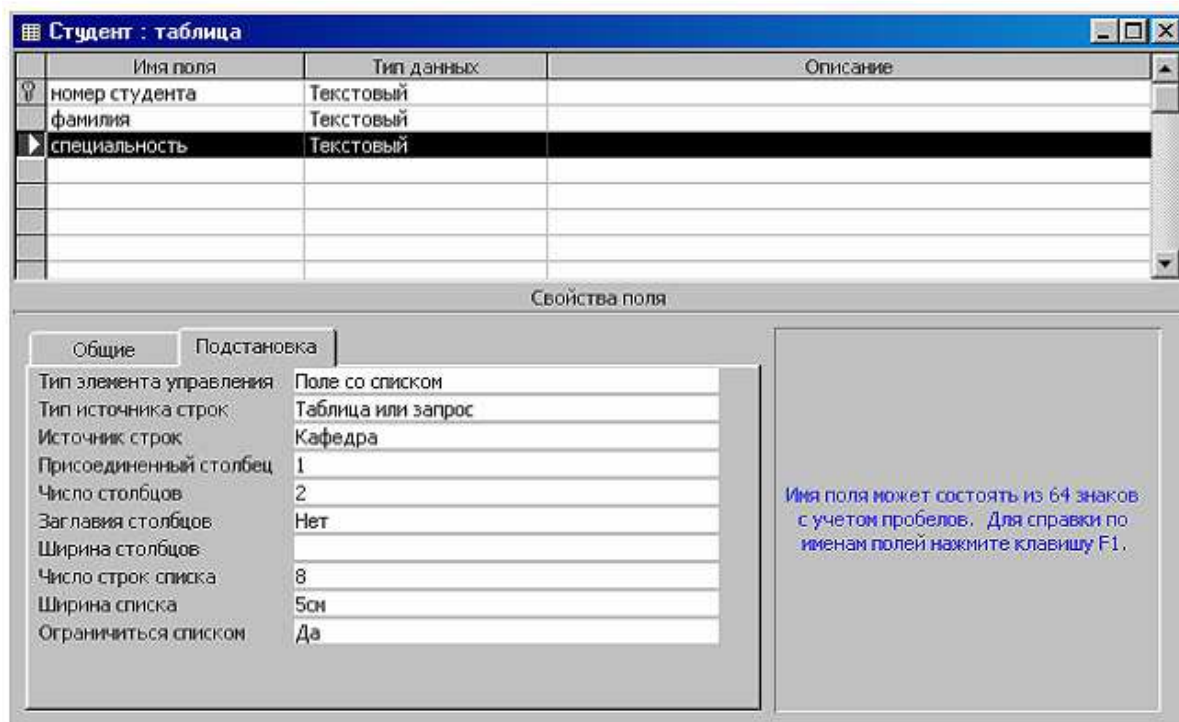


Рис. 37. Настройка свойств элемента управления для поля специальность

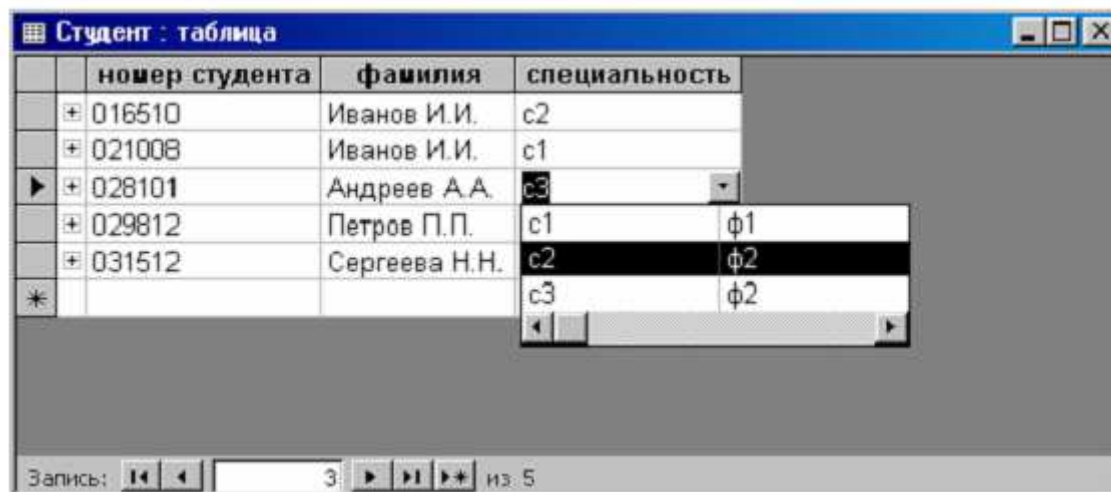


Рис. 38. Ввод данных под управлением поля со списком

Проблема ввода корректных значений внешнего ключа актуальна и для таблицы «Сессия». Внешний ключ «номер студента» в таблице «Сессия» также должен принимать только «разрешенные» значения. Источником разрешенных номеров студентов является таблица

«Студент». По аналогии с рассмотренным примером настройте свойство подстановка поля «номер студента» таблицы «Сессия» (создайте на основе таблицы «Студент» поле со списком разрешенных значений).

Более универсальное средство контроля ссылочной целостности обеспечивает настройка свойств связей между таблицами.

**Задание 11.** Для обеспечения целостности и корректности данных в БД установите связи между таблицами, т.е. создайте корректную схему данных БД:

- 1) выполните команду **Сервис | Схема данных** – появится макет схемы данных БД (рис. 39);
- 2) для установления контроля целостности связи между таблицами «Кафедра» и «Студент» щелкните правой клавишей мыши по изображению связи и выберите в контекстном меню **«Изменить связь»** – раскроется окно «Изменение связей» (рис. 39);
- 3) в окне «Изменение связей» мышью «включите» настройку **«Обеспечение целостности данных»** и нажмите экранную кнопку ОК. Попытка установления связи увенчается неудачей (рис. 40). Причина в том, что в таблице «Студент» нарушено условие целостности данных – введены данные о студенте с незарегистрированной в БД специальностью;

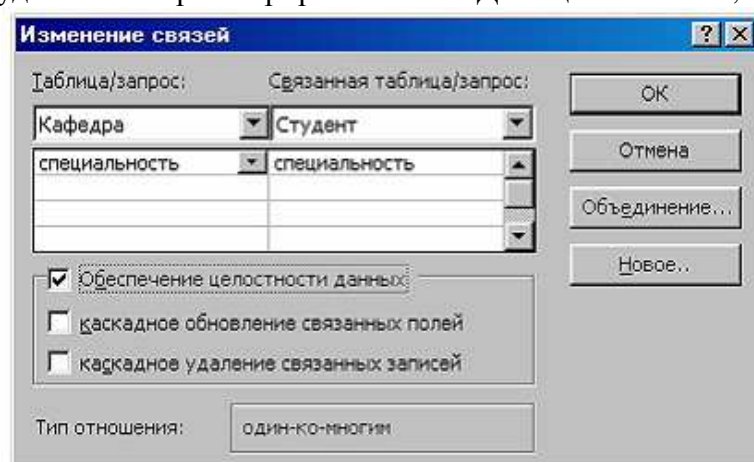


Рис. 39. Окно спецификации связей в схеме данных

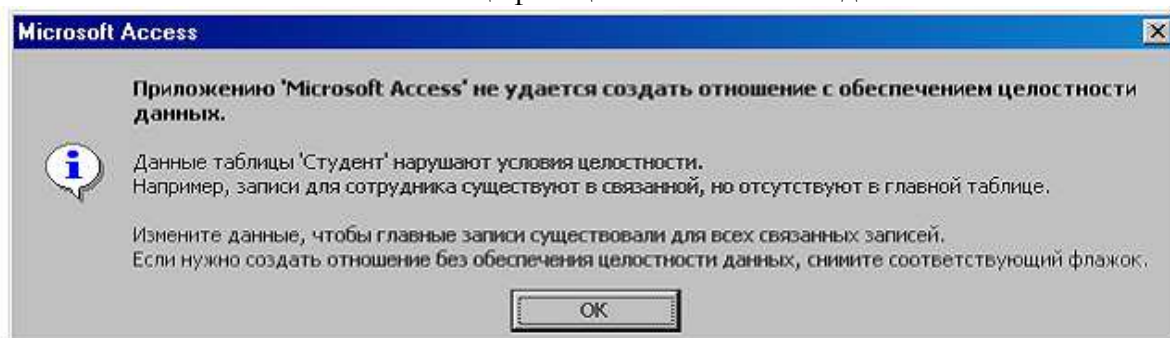


Рис. 40. Сообщение о неудачной попытке установления связи

- 4) для установления контроля целостности связи между таблицами «Студент» и «Сессия» щелкните правой клавишей мыши по изображению связи и выберите в контекстном меню **«Изменить связь»** – раскроется окно «Изменение связей»;

5) в окне «Изменение связей» мышью «включите» состояние **«Обеспечение целостности данных»** и нажмите экранную кнопку ОК – связь будет установлена. Корректная связь в схеме данных выделяется жирной линией и метками кардинальности связи **1** и **∞** (рис. 41);

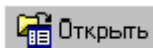
б) сохраните результат создания связей между таблицами по команде **Файл | Сохранить**. Ничего, что попытка установления связи с обеспечением целостности данных между таблицами «Кафедра» и «Студент» оказалась неудачной. Схему данных (макет), как и любой объект Access, можно отредактировать в новых условиях.

Из полученных результатов важно уяснить себе следующее:

- 1) методы создания связи с обеспечением целостности данных;
- 2) причины неудачи создания связей;
- 3) невозможность установления контроля целостности некорректных данных.

**Задание 12.** Причина неудачи создания связи между таблицами «Кафедра» и «Студент» в предыдущем задании состоит в том, что в таблице «Студент» нарушено условие целостности данных: введены данные о студенте с незарегистрированной в БД специальностью. Исправьте этот дефект:

1) в окне БД на закладке **Таблица** выберите из списка таблицу «Студент» и нажмите экранную



кнопку

- 2) удалите вручную запись: «000000 Одинцов О.О. с0»;
- 3) сохраните результат по команде **Файл | Сохранить** и **Файл | Закрывать**;
- 4) вернитесь к настройке свойств связей по команде **Сервис | Схема данных** – появится макет схемы данных;
- 5) щелкните мышью по связи «Кафедра»:«Студент» – раскроется окно «Изменение связей» (рис. 39);
- 6) в окне «Изменение связей» мышью «включите» состояние «**Обеспечение целостности данных**» и нажмите экранную кнопку ОК. Макет схемы данных примет вид, как на рис. 41;
- 7) сохраните результат создания связей между таблицами по команде **Файл | Сохранить**.

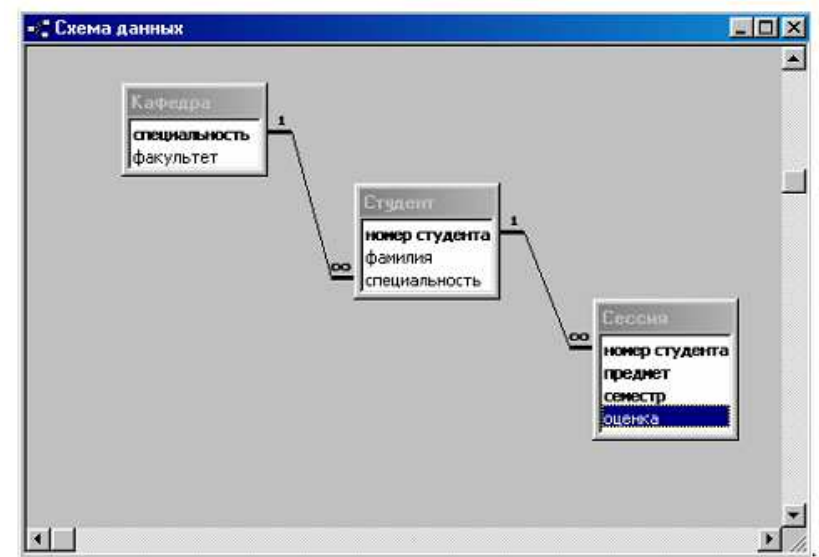


Рис. 41. Корректная схема базы данных УЧЕБА

Внимание! Важный результат: наша демонстрационная ПО УЧЕБА представлена БД в виде совокупности нормализованных таблиц «Кафедра», «Студент» и «Сессия», связанных корректными связями, которые обеспечивают контроль целостности данных.

### Манипулирование данными в связанных таблицах

Поскольку в нашей БД теперь предусмотрен контроль целостности данных, произвол в манипулировании данными не возможен. Правила ссылочной целостности регламентируют выполнение операций добавления, удаления и обновления над данными в связанных таблицах. Основными правилами являются «запрет» и «каскадирование». При отсутствии специальных настроек свойств связей действуют правила, устанавливаемые по умолчанию (правила запрета). Для неидентифицирующих и идентифицирующих связей действуют правила запрета. Запрещаются удаление и обновление записи из таблицы-отца, если в таблице-сыне есть

связанные с ней записи. Запрещаются добавление и обновление записи в таблице-сыне, если в таблице-отце соответствующие данные отсутствуют.

Для того чтобы при манипулировании данными одной из связанных таблиц выполнялись адекватные изменения других таблиц, могут быть «включены» каскадные обновление и удаление данных [7–10].

Какие правила обеспечения целостности данных выбрать, зависит от семантики данных моделируемой ПО. Рассмотрим эти проблемы на нашем примере БД УЧЕБА.

Конкретный студент (строка в таблице «Студент») не может быть зачислен без отнесения его к определенной специальности (внешний ключ «специальность» не может принимать Null-значение), следовательно, добавление в таблицу «Студент» новой строки, не имеющей соответствующей строки в таблице «Кафедра», должно быть запрещено (нельзя принимать студентов на несуществующую специальность). По этой же причине удаление соответствующей строки в таблице «Кафедра» должно быть запрещено, пока по этой специальности обучается хотя бы один студент. Альтернативное правило каскадирования означало бы, что при закрытии (удалении) специальности все ее студенты должны быть отчислены. Вряд ли такая семантика может быть оправдана.

В случае обновления строки в таблице «Кафедра» (например, изменении номера специальности) разумно допустить каскадирование, т.е. адекватное изменение этих же данных в таблице «Студент».

В случае идентифицирующей зависимости между таблицами «Студент» и «Сессия» разумно предположить запрет на добавление в таблицу «Сессия» новой строки, не имеющей соответствующей строки в таблице «Студент» (иначе будет не понятно, чей экзамен) и каскадирование для операций обновления и удаления в отношении строк таблицы «Студент». Действительно, изменение номера студента в таблице «Студент» должно быть распространено на связанные строки таблицы «Сессия». При удалении строки в таблице «Студент» (например, при отчислении студента) могут быть удалены все связанные строки в таблице «Сессия» (результаты сдачи всех экзаменов этим студентом). Каскадное удаление – чрезвычайно ответственная операция, потому что удаление одной строки старшей таблицы может привести к удалению нескольких тысяч строк в подчиненных таблицах.

**Задание 13.** Настройте контроль целостности в схеме данных на каскадные обновление и удаление данных:

- 1) раскройте макет схемы данных по команде **Сервис | Схема данных**;
- 2) для редактирования связи «Кафедра»:«Студент» щелкните мышью по связи – раскроется окно «Изменение связей» (рис. 42);
- 3) в окне «Изменение связей» мышью «включите» настройку **«Каскадное обновление связанных полей»** и нажмите кнопку **ОК** (рис. 42);
- 4) для редактирования связи «Студент»:«Сессия» щелкните мышью по связи. Раскроется окно «Изменение связей» (рис. 43);
- 5) в окне «Изменение связей» мышью «включите» состояния **«Каскадное обновление связанных полей»** и **«Каскадное удаление связанных полей»**; нажмите экранную кнопку **ОК** (рис. 43);

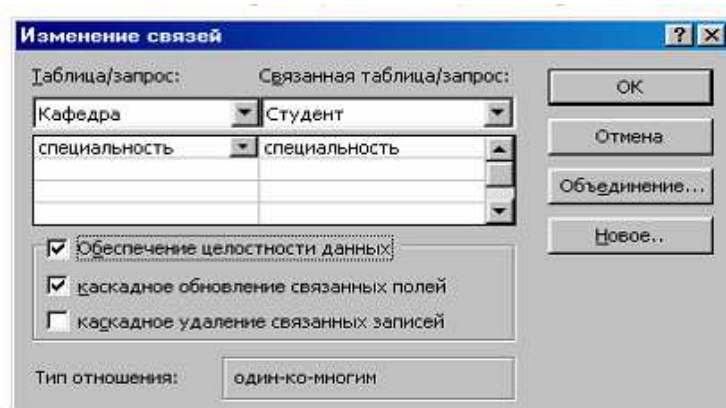


Рис. 41. Настройка свойств связи «Кафедра» : «Студент»

б) сохраните результаты редактирования связей между таблицами по команде **Файл | Сохранить**;

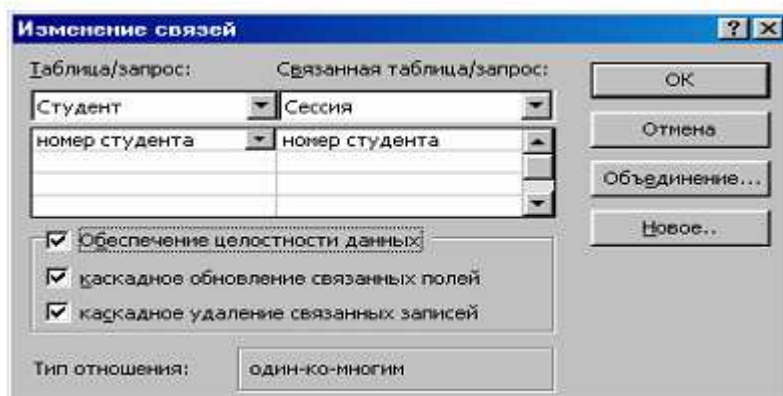



Рис. 42. Настройка свойств связи «Студент» : «Сессия»

**Задание 14.** Выполните простейшие манипуляции с данными в условиях контроля целостности данных:

- 1) попробуйте добавить в сыновью таблицу «Студент» запрещенные данные. В окне БД на закладке **Таблица** выберите из списка таблицу «Студент» и нажмите экранную кнопку  ;
- 2) добавьте ручную запись: «000000 Одинцов О.О. с0». При попытке сохранить некорректные данные появится диагностическое сообщение о невозможности ввода (рис. 44);

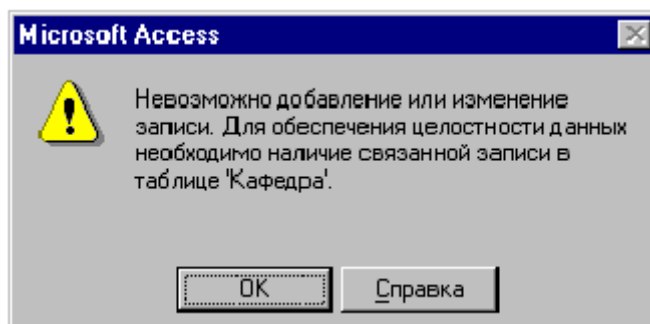



Рис. 43. Диагностическое сообщение о невозможности добавления записей

- 3) попробуйте теперь удалить из отцовской таблицы «Кафедра» запись, имеющую потомков в сыновьей таблице «Студент». В окне БД на закладке **Таблица** выберите из списка таблицу «Кафедра» и нажмите экранную кнопку  ;

4) удалите запись «с1 ф1». При попытке удаления появится диагностическое сообщение о невозможности этой операции (рис. 45);

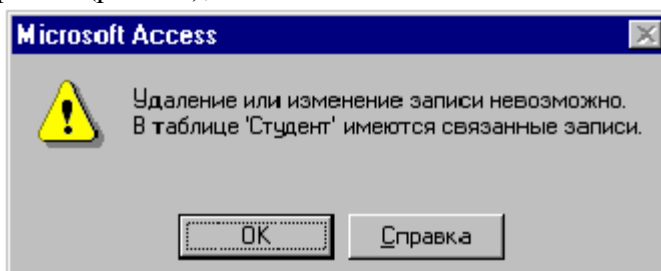


Рис. 44. Диагностическое сообщение о невозможности удаления связанных записей

5) попробуйте теперь изменить в отцовской таблице «Кафедра» запись, имеющую потомков в сыновней таблице «Студент»;

6) измените в таблице «Кафедра» номер специальности «с2» на «с4» в соответствующей записи;

7) посмотрите, как эта операция скажется на подчиненной таблице «Студент». В окне БД на закладке Таблица выберите из списка таблицу «Студент» и нажмите экранную кнопку

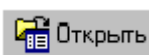


. Должно произойти каскадное обновление данных в подчиненной таблице;

8) попробуйте изменить в таблице «Студент» её независимые данные, например, «номер студента»;

9) измените в соответствующей строке «номер студента» (это Сергеева Н.Н.) «031512» на «040000» и сохраните результаты по команде **Файл | Сохранить**;

10) посмотрите, как эта операция скажется на подчиненной таблице «Сессия». В окне БД на закладке Таблица выберите из списка таблицу «Сессия» и нажмите экранную кнопку



. Должно произойти каскадное обновление данных в подчиненной таблице;

11) попробуйте удалить в таблице «Студент» хотя бы одну её запись.

В окне БД на закладке Таблица выберите из списка таблицу «Студент» и нажмите экранную



кнопку ;

12) удалите запись о студенте Андрееве А.А. в таблице «Студент» в соответствующей строке и сохраните результаты по команде **Файл | Сохранить**;

13) посмотрите, как эта операция скажется на подчиненной таблице

«Сессия». В окне БД на закладке Таблица выберите из списка соответствующую таблицу и



нажмите экранную кнопку . Зависимые записи (такая была одна о студенте Андрееве А.А. под номером 028101) должны исчезнуть. Должно произойти каскадное удаление данных в подчиненной таблице;

14) попробуйте добавить в таблицу «Сессия» некорректные данные. В окне БД на закладке



«Таблица» выберите из списка таблицу «Сессия» и нажмите экранную кнопку ;

15) добавьте вручную запись: «000000 физика 2 3». При попытке сохранить некорректные данные появится диагностическое сообщение о невозможности ввода;

16) попробуйте удалить запись в таблице «Сессия» «031512 математика 1 5»;

17) убедитесь, что эта операция прошла без проблем.



## ЗАКЛЮЧЕНИЕ

Собственно разработка структуры БД включает в себя:

1. Разработку структуры каждой таблицы, в том числе:

- 1) определение имен всех ее полей;
- 2) определение типов данных полей;
- 3) определение свойств полей;
- 4) определение ключевых полей.

2. Построение схемы базы данных, в том числе:

- 1) установление корректных связей между таблицами;
- 2) обеспечение контроля целостности связей;
- 3) установление правил целостности данных (запрет или разрешение каскадных преобразований).

Разработка структуры БД должна быть выполнена до наполнения ее содержательной информацией.