**Информационные технологии создания и обработки баз данных с помощью MS Access XP**

МЕТОДИЧЕСКИЕ УКАЗАНИЯ

по курсу «Информатика»

В методических указаниях излагается материал для освоения и приобретения навыков работы с СУБД MS ACCESS XP. Рассматриваются основные понятия, способы создания таблиц базы и схемы взаимосвязей между ними. Дается методика разработки и использования форм для ввода, корректировки и просмотра данных, применения диалоговых средств выборки и обработки данных, конструирования отчетов.

Рассматриваемая технология разнообразных средств СУБД MS ACCESS XP позволит успешно применять данный пакет для решения практических задач.

Предназначены для студентов очной формы обучения дипломированных специалистов по направлению 521500 Менеджмент, специальностям 061100 Менеджмент организации, 061500 Маркетинг, 060400 Финансы и кредит.

Ил. 52. Табл. 3. Библиогр.: 2 назв.

**Содержание**

Введение

1. Цель работы

2. Краткие сведения и основные понятия

3. Интерфейс Access

4. Начало работы в Microsoft Access

5. Создание таблиц данных

5.1. Создание таблицы в режиме конструктора

5.2. Создание таблицы в режиме таблицы

5.3. Создание таблицы с помощью мастера таблиц

6. Поиск, сортировка и фильтрация данных

6.1. Поиск и замена значений в полях

6.2. Сортировка записей

6.3. Использование фильтра

7. Построение реляционной базы данных и разработка инфологической модели

8. Разработка форм

8.1. Разработка формы с помощью мастера форм

8.2. Разработка многотабличных форм

9. Запросы

9.1. Разработка запроса в режиме конструктора

9.2. Разработка запросов на редактирование данных

9.2.1. Запросы на обновление данных

9.2.2. Запросы на удаление данных

9.2.2. Запросы на добавление данных

10. Разработка отчетов

10.1. Создание однотабличного отчета в режиме конструктора

10.2. Создание отчета на основе запроса

11. Задание к лабораторной работе

12. Используемая литература

**Введение**

Microsoft Access – это одно из приложений, входящих в состав пакета Microsoft Office. Приложение является мощной программой обработки баз данных, которую можно использовать для хранения, сортировки и управления практически любыми типами данных. В качестве примеров можно привести имена и адреса, информацию о продавцах и складах и даже любимые рецепты.

Microsoft Access является системой управления реляционными базами данных. *Реляционная база данных* (relational database) – это такая база данных (БД), записи которой можно сравнивать друг с другом и другими источниками данных и анализировать в целях создания динамического источника данных. Содержание такой БД может изменяться на основе результатов сортировки, запросов или других операций с данными. Хотя Microsoft Access является довольно мощной программой, ее легко использовать. БД определяется простым указанием полей данных, которые должны содержаться в базе (например, имя, адрес и индекс). Можно легко создавать формы, которые помогут вводить данные в базу, не задумываясь о работе программы управления этой БД, просматривать введенные данные в удобном виде, создавать запросы для отбора информации, удовлетворяющей определенным условиям и формировать отчеты.

# Цель работы

Изучение программы обработки баз данных Microsoft Access. Получение навыков создания баз данных, форм для работы с ними, межтабличных связей, умения выбрать определенные данные по запросам, а также конструирования различных видов отчетов.

# Краткие сведения и основные понятия

Microsoft Access – это система управления базами данных (СУБД). СУБД является универсальным программным инструментом создания и обслуживания БД, позволяет не только хранить большие массивы данных в определенном формате, но и обрабатывать их, представляя в удобном для пользователей виде. Access дает возможность автоматизировать часто выполняемые операции, разрабатывать удобные формы для ввода, корректировки и просмотра данных, составлять сложные отчеты. В Microsoft Access база данных включает в себя все объекты, связанные с данными, в том числе и те, которые предназначены для автоматизации работы с ними, такие, как таблицы, формы, запросы, отчеты, программный код.

СУБД Access включает в себя разнообразные и многочисленные относительно автономные программные средства, ориентированные на создание объектов базы данных и приложений пользователя.

*Средства графического конструирования* позволяют пользователю создавать объекты базы данных и объекты приложения с помощью многочисленных графических элементов, не прибегая к программированию. *Диалоговые средства* представлены разнообразными мастерами, которые в режиме ведения диалога с пользователем позволяют создавать объекты и выполнять разнообразные функции по реорганизации и преобразованию БД. Среди многочисленных средств следует выделить средства для создания:

* таблиц и схем баз данных;
* запросов выборки, отбирающих и объединяющих данные нескольких таблиц в виртуальную таблицу, которая может использоваться во многих задачах приложения;
* запросов на изменение баз данных;
* экранных форм, предназначенных для ввода, просмотра и обработки данных в диалоговом режиме;
* отчетов, предназначенных для просмотра и вывода на печать данных из базы и результатов их обработки в удобном для пользователя виде.

В СУБД поддерживаются различные модели данных.

*Модель данных* – это метод (принцип) логической организации данных, используемый СУБД. Наиболее известными являются иерархическая, сетевая и реляционная модели. В СУБД для ПЭВМ поддерживается преимущественно *реляционная модель*, которую отличает простота и единообразие представления данных простейшими *двумерными таблицами*.

Основной логической структурной единицей манипулирования данными является строка таблицы – запись. Структура *записи* определяется составом входящих в нее полей. Совокупность полей записи соответствует логически связанным реквизитам, характеризующим некоторую сущность предметной области. Типовыми функциями СУБД по манипулированию данными являются: выборка, добавление, удаление, изменение данных.

Microsoft Access называет объектами все, что может иметь имя, такими объектами являются таблицы, запросы, формы, макросы и модули. Рассмотрим список объектов Microsoft Access.

*Таблица* – объект, который используется для хранения информации. Таблица содержит *поля* (поле – столбец таблицы; поле содержит определенное свойство объекта; каждое поля имеет имя; внутри имени поля нельзя использовать пробелы; для связки между словами можно ставить знак подчеркивания;тип поля определяет множество значений, которые может принимать данное поле в различных записях; в реляционных базах данных используется четыре основных типа полей: числовой, символьный, дата, логический; логический тип соответствует полю, которое может принимать всего два значения: "да" – "нет" или "истина" – "ложь"), в которых хранятся различного рода данные, например, фамилия и адрес, и *записи* (запись – строка таблицы; одна запись содержит информацию об отдельном объекте, описываемом в БД).

*Запрос* – объект, который позволяет пользователю получить нужные данные из одной или нескольких таблиц. Для создания запроса можно использовать QBE (запрос по образцу) или инструкции SQL. Можно создавать запросы на выборку, удаление или добавление данных, можно создавать новые таблицы, используя данные одной или нескольких таблиц, которые уже существуют.

*Форма* – объект, предназначенный в основном для ввода данных, отображения их на экране или управления работой приложения.

*Отчет* – объект, предназначенный для создания документа, который впоследствии может быть распечатан или включен в документ другого приложения.

*Макрос* – объект, представляющий собой структурированное описание одного или нескольких действий, которые, по мнению разработчика, должен выполнить Access в ответ на определенное событие.

*Модуль* – объект, содержащий программы на Microsoft Access Basic, которые позволяют разбить процесс на более мелкие действия и обнаружить ошибки, которые невозможно обнаружить с помощью макросов. Модули могут быть независимыми объектами, содержащими функции, которые вызываются из любого места приложения, также модули могут быть «привязаны» к отдельным формам или отчетам для реакции на происходящие в них изменения.

# Интерфейс Access

Access имеет характерный для всех приложений Microsoft Windows удобный графический интерфейс, ориентированный на комфортную работу пользователя. Для работы с таблицами БД и другими объектами Access предоставляет многочисленные команды меню и контекстно-зависимые панели инструментов. В Access, как и в других приложениях Microsoft Office XP, часто выполняемые задачи вынесены в области, которые отображаются вместе с базой данных. В Access область задач **Создание файла** (New File) позволяет, не прерывая работы, быстро создавать новые документы или открывать файлы, область задач **Поиск** (Search) – выполнять поиск файлов, область задач **Буфер обмена** (ClipBoard) - выбирать объекты для вставки.

Пользователь имеет возможность переносить объекты БД и их элементы с помощью мыши. Например, любую таблицу или запрос можно перенести из окна БД в окно схемы данных. Для установления связи между объектами можно в схеме данных переместить поле из одной таблицы в другую. Для размещения подчиненной формы в главной достаточно перенести в нее ранее созданную форму или даже просто перенести таблицу-источник в конструируемую форму.

В Access предусмотрено широкое использование технологии, которая помогает пользователю ориентироваться в выборе необходимых действий и обеспечивает высокую производительность труда за счет автоматизации выполнения основных функций. Выдача Помощником контекстно-зависимой справочной информации помогает принять решение, как лучше выполнить то или иное действие или найти нужный инструмент в Access.

Справочная система Access построена на основе HTML. При вызове справки ее окно не заслоняет окна Access, которое автоматически изменяет размер, освобождая пространство для окна справки. В Access XP обеспечивается более удобный доступ к справочной системе через поле **Введите вопрос**, расположенное в строке меню.

Access располагает разнообразными диалоговыми средствами пользователя, которые позволяют создавать приложения для решения задач, не прибегая к разработке запросов на языке SQL или к программированию макросов или модулей на языке Microsoft Visual Basic for Applications. Для автоматизации процесса создания объектов базы данных (таблиц, запросов), схем базы данных и объектов приложения (форм, отчетов) используются специализированные диалоговые графические средства, называемые **Конструктор** (Design). Конструктор предоставляет пользователю набор инструментов, с помощью которых можно быстро создать и модифицировать объект. Предусмотрено автоматическое конструирование форм, запросов, отчетов, страниц и их элементов с помощью *программ-мастеров* и команд, начинающихся с приставки «авто».

Множество мастеров Access позволяет автоматизировать процесс создания таблиц БД, форм, запросов, отчетов, анализировать таблицы БД и выполнять многие другие работы. Практически для любых работ имеется *Мастер* (Wizard), который поможет их выполнить. Вот перечень некоторых из них: мастер подстановок, мастер запросов, мастера по созданию форм и отчетов, мастер кнопок, мастер анализа таблиц, мастер сводных таблиц, мастер баз данных и т.д.

# Начало работы в Microsoft Access

Чтобы начать работу с СУБД Access, необходимо после загрузки операционной системы запустить ее. Это можно сделать, например, так: в меню **Пуск** выбрать пункт **Программы** и в появившемся меню щелкнуть на пункте **Microsoft Access.** Появившееся после запуска окно Access представлено на Рис. 1. Для быстрого запуска Access удобно иметь ярлык этой программы на рабочем столе Windows. Создать ярлык можно разными способами. Например, выберите программу Microsoft Access, как при ее запуске. Нажмите правую кнопку мыши. В контекстном меню последовательно выберите команды **Отправить** и **Рабочий стол**. Ярлык будет создан и отобразится на рабочем столе и запуск можно выполнить двумя нажатиями мыши на ярлыке.

Рассмотрим основные элементы окна Access: строка заголовка, строка меню, панель инструментов, область задач.

В строке заголовка содержится кнопка системного меню, название приложения и кнопки, управляющие представлением окна на экране.

Строка меню содержит несколько пунктов, каждый из которых имеет собственное ниспадающее меню, открываемое щелчком мыши на этом пункте. Ниспадающее меню содержит команды Access. Для получения справки о назначении команды можно вызвать справку, набрав команду **Справка|Что это такое ?. Э**та команда позволяет получить сведения как о доступных, так и о недоступных командах в текущем режиме Access.

Строка заголовка

строка меню

панель инструментов

область задач

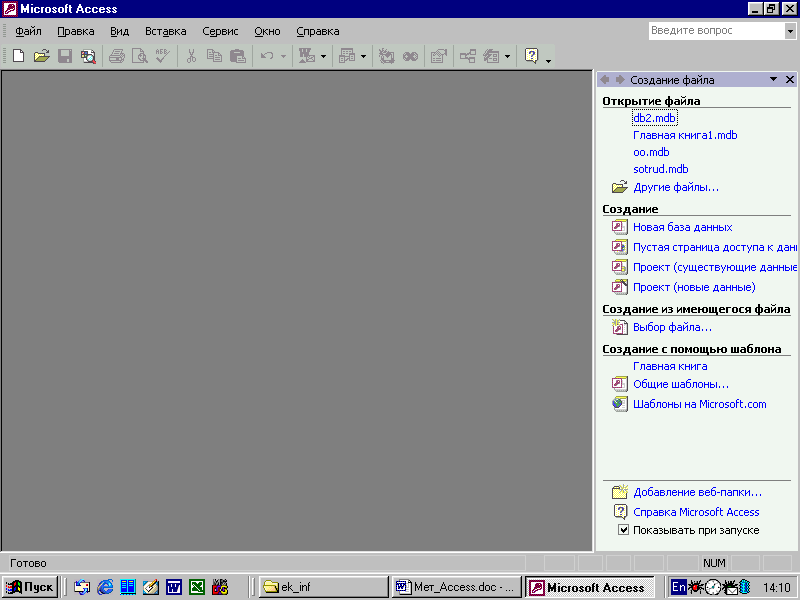


Рис. 1

Панель инструментов состоит из значков-кнопок, представляющих собой наиболее часто используемые команды меню. Кнопки панели позволяют, не обращаясь к командам, выполнить те же операции, но белее быстрым и удобным способом. Набор команд и их графических представлений на панели инструментов является основой интерфейса пользователя, который позволяет в диалоговом режиме выполнить основные работы в среде Access.

Панель **Область задач** отображается после запуска Access, она позволяет начать работу с базой данных (уже существующей или вновь создаваемой).

В отличие от многих других приложений (таких, как Word), пользователю нет необходимости специально сохранять базу данных. Сохранение выполняется автоматически после присвоения названия БД. Access записывает данные на диск всякий раз при вводе записей или внесении изменений в объекты БД. Именно поэтому в начале создания базы данных Access запрашивает ее название. Необходимо сохранять объекты внутри самой БД после их создания или редактирования (формы, отчеты, запросы и т.д.).

# Создание таблиц данных

Рассмотрим на примере таблицы данных *Преподаватели* ее создание в MS ACCESS. Ее структура представлена в Таблица 1.

Таблица 1.

|  |  |  |
| --- | --- | --- |
| Имя поля | Тип данных | Размер поля |
| Код преподавателя | Счетчик |  |
| Фамилия | Текстовый | 15 |
| Имя | Текстовый | 15 |
| Отчество | Текстовый | 15 |
| Дата рождения | Дата/время | Краткий |
| Должность | Текстовый | 9 |
| Дисциплина | Текстовый | 30 |
| Телефон | Текстовый | 9 |
| Оклад | Денежный |  |

При вводе данных в поле **Должность** должны вводиться только слова *Профессор, Доцент* или *Ассистент*. В противном случае на экране должно появляться сообщение о неверном вводе данных. Значение по умолчанию в этом поле сделаем *Доцент*.

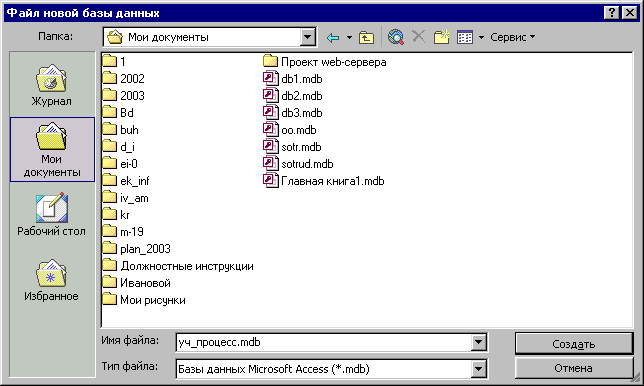


Рис. 2

Для создания новой базы данных после запуска Access на панели задач нужно выбрать пункт **Новая база данных**. Если панель задач не открыта, можно воспользоваться командами в строке меню **Файл – Создать**, после чего появится панель задач. После выбора пункта меню в появившемся окне (Рис. 2) нужно выбрать каталог, в котором будет сохранен файл БД, и задать имя файла новой БД. После нажатия кнопки **Создать** открывается окно новой БД (Рис. 3).

Access для создания новой таблицы БД предлагает три способа их создания: в режиме конструктора, с помощью мастера, путем ввода данных.

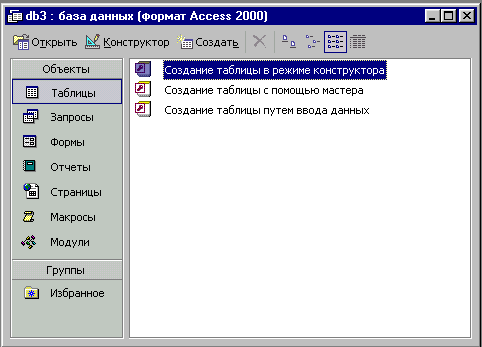


Рис. 3

## Создание таблицы в режиме конструктора

При выборе *режима конструктора* появляется окно **Таблица 1**, в котором определяется структура таблицы БД (Рис. 4).

При отображении таблицы в режиме **Конструктор** в окне можно выделить три основные области: панель инструментов **Конструктор таблиц,** области панели **Бланк** и **Свойства поля**.

Область панели инструментов используют для модификации содержимого этой области. Основные кнопки панели:

Область панели **Бланк** – основная часть окна **Режим конструктора таблиц**. Здесь можно задать имена полей, типы данных и их описание. Область состоит из столбца маркера текущей строки и ключевого поля (крайний слева), столбцов **Имя поля, Тип даных** и **Описание**. В столбце **Имя поля** можно изменять (или назначать) имена полей. Имена должны содержать не более 64 символов, цифр и пробелов. Кроме того, они должны быть уникальны для данной таблицы. **Типы данных**, которые можно хранить в данном поле, определяются в данном столбце. Недопустимо заносить в поле данные несовместимого с ним типа. Нельзя, например, размещать текст в числовом поле.

В столбце **Описание** хранится информация о поле.

Общие свойства для каждого поля задаются в области **Свойства полей**, эти свойства можно установить, чтобы задать метод сохранения и отображения данных этого поля. Поля, создаваемые в режиме **Конструктор**, используются другими объектами базы данных, работающими с таблицами, например, формами, отчетами и запросами. Свойства полей изменяют на вкладках **Общие** и **Подстановка.**

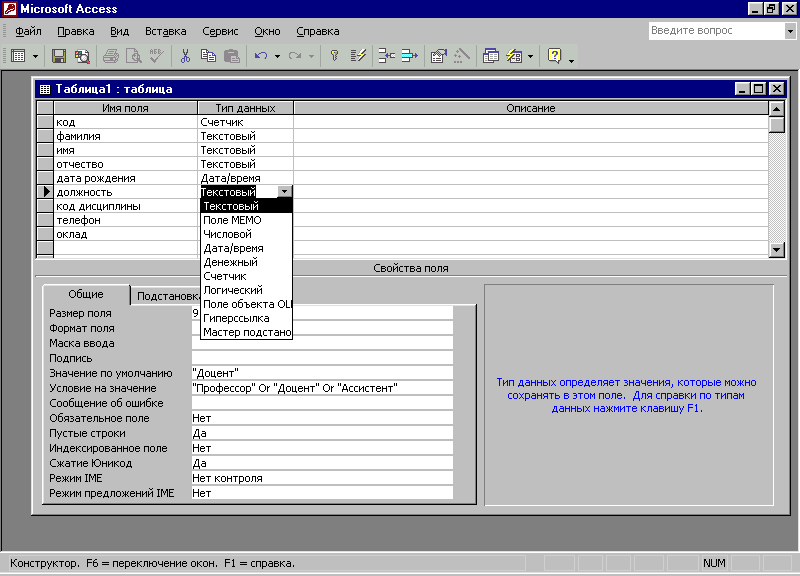


Рис. 4

На вкладке **Подстановка** задается свойство **Тип элемента управления**, которое определяет, в каком виде будет отображаться поле в таблице и в форме: в виде **Поля**, **Списка** или **Поля со списком**. Если для поля выбран тип элемента управления **Список** или **Поле со списком**, на вкладке **Подстановка** появляются дополнительные свойства.

Каждая таблица в реляционной БД должна иметь уникальный ключ, однозначно определяющий каждую запись. Это позволяет быстро найти и связать данные из разных таблиц в запросах, формах и отчетах. Ключевое поле должно содержать уникальные значения, такие как коды или инвентарные номера и не может содержать значения **Null**. Если для таблицы определен первичный ключ, то Access предотвращает дублирование ключа или ввод значения **Null** в эти поля. Ключ может быть простым или составным, включающим несколько полей (до 10). Для определения ключа выделите составляющие ключ поля и нажмите на панели инструментов **Конструктор таблиц** кнопку **Ключевое поле** или выполните команду меню **Правка/Ключевое поле**. Для ключевого поля автоматически строится индекс. Окно **Индексы** (Рис. 5) вызывается щелчком на кнопке просмотра и редактирования индексов **Индексы** на панели инструментов или выполнением команды меню **Вид/Индексы.**

В нашем случае после выбора создания таблицы в режиме конструктора в столбце **Имя поля** вводим имена полей из первого столбца таблицы Таблица 1., в столбце **Тип данных** выбираем типы в соответствии с значениями во втором столбце вышеуказанной таблицы, размеры полей из третьего столбца таблицы Таблица 1. указываем на вкладке **Общие** в строке **Размер поля**. Для ввода огра ничений на значение поля **Должность** необходимо сделать текущим имя поля Должность, далее на вкладке **Общие** в строке **Значение по умолчанию** ввести слово *Доцент*. Далее перейти на строку **Условие на значение** и щелкнуть по кнопке . В появившемся окне **Построитель выражений** набрать слова *Профессор, Доцент* и *Ассистент*, соединив их логическим выражением *OR* (или) (Рис. 6)

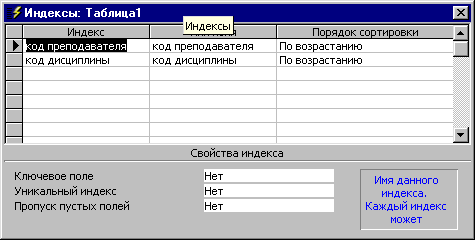


Рис. 5

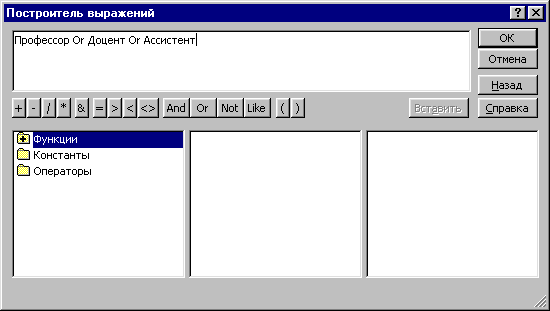


Рис. 6

Далее в строке **Сообщение об ошибке** введем текст «Такой должности нет, введите данные правильно».

В качестве ключевого поля задается «Код преподавателя».

Для сохранения таблицы нужно выбрать **Файл/Сохранить**. В окне сохранения указывается имя таблицы (например, *Преподаватели*).

**Примечание.** Режим конструктора позволяет в любой момент изменить ее структуру вне зависимости от способа ее создания.

Далее в режиме таблицы заполним эту таблицу (см.Рис. 7)

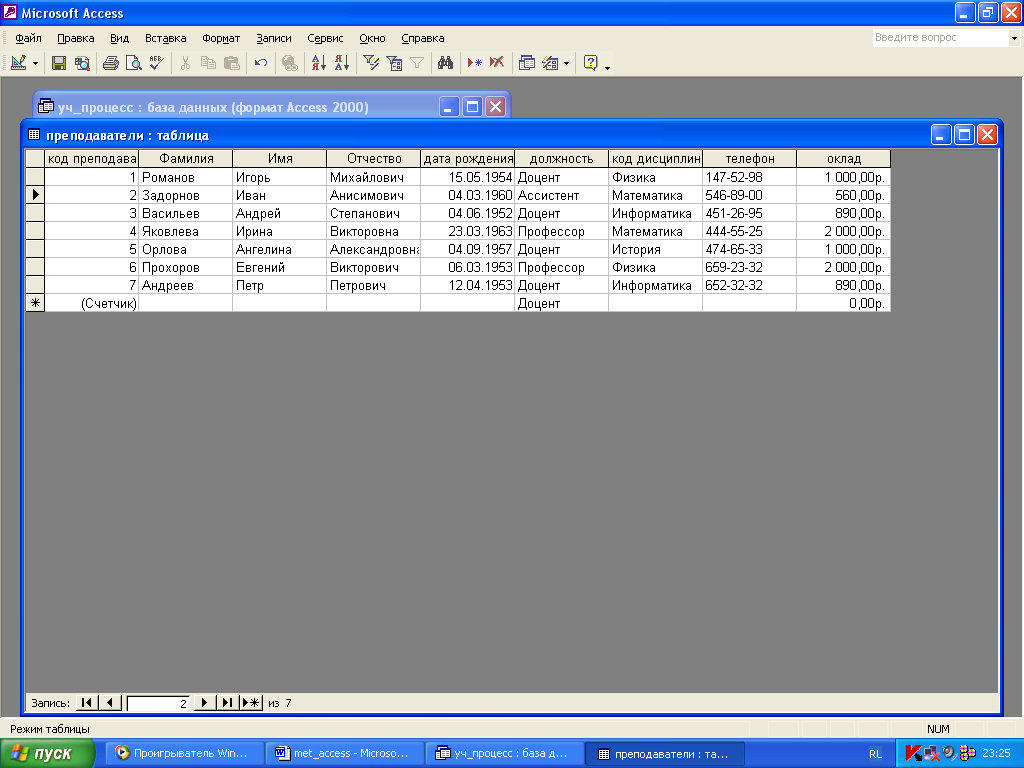


Рис. 7

## Создание таблицы в режиме таблицы

При выборе *режима таблицы* пользователь может создать таблицу, не определяя предварительно ее структуру. Создание новой таблицы в режиме таблицы осуществляется выбором строки **Создание таблицы путем ввода данных** в рабочем поле окна базы данных или строки **Режим таблицы** в окне **Новая таблица**. После выбора режима открывается пустая таблица, в которую можно вводить данные (Рис. 8).

При сохранении этой таблицы Access проанализирует данные и автоматически присвоит соответствующий тип данных каждому полю, т.е. автоматически создает структуру таблицы. Таблица, предлагаемая Access в таком режиме создания, содержит 10 столбцов и 21 строку. Полям таблицы по умолчанию присваиваются имена **Поле1, Поле2** и т.д. Любое поле этой таблицы можно переименовать в соответствии с требованиями пользователя, непосредственно редактируя имена в заголовке столбцов. Для этого дважды щелкните кнопкой мыши на области выделения столбца, содержащей его имя, или выполните команду **Формат/Переименовать столбец**. Если требуется создать таблицу, содержащую более 10 полей, то можно вставить новые столбцы. Для этого установите курсор в области столбца, слева от которого требуется вставить новый, и выполните команду **Вставка/Столбец**. Кроме переименования и вставки столбцов, допускается удаление столбцов или изменение их положения в таблице. Для удаления столбца его необходимо пометить, щелкнув кнопкой мыши на его заголовке, и выполнить команду **Правка/Удалить столбец**. Для перемещения столбца его следует пометить, оставив курсор на заголовке, нажать кнопку мыши и, удерживая ее, перетащить столбец в нужное место.

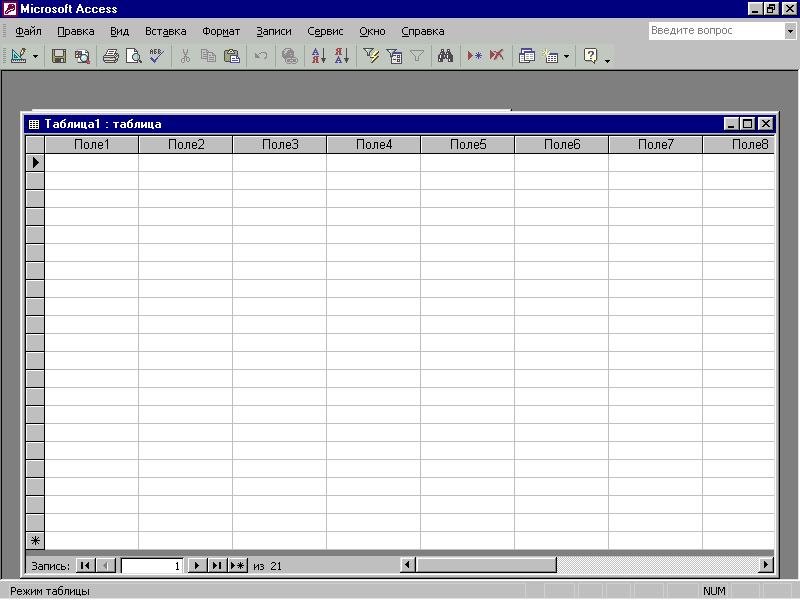


Рис. 8

По окончании ввода данных во все нужные столбцы необходимо сохранить таблицу, выполнив соответствующую команду. При сохранении таблицы выводится приглашение для создания ключевого поля. При выборе кнопки **Да** в таблицу будет добавлено ключевое поле типа **Счетчик**. Если введены данные, которые могут однозначно идентифицировать каждую запись, и их надо сделать ключевыми, то нужно нажать кнопку **Нет**. В этом случае пользователь должен самостоятельно создать ключ в режиме конструктора.

Создадим в этом режиме таблицу *Код дисциплин*, включив в нее следующие дисциплины: информатика, математика, физика, история. Присвоим каждой дисциплине код (Рис. 9).

После ввода данных переименуем поля, для этого дважды щелкнем мышкой в области названия **Поле1** и введем *Код дисциплины,* аналогично поменяем имя поля Поле2 на *Название дисциплины.* После этого нужно закрыть таблицу и в окне на сохранение задать имя этой таблице (например, *Дисциплины*). При сохранении в случае отсутствия ключевых полей Access выводит сообщение (Рис. 10). В случае выбора кнопки **Да** Access автоматически добавляет еще одно поле типа *Счетчик* и делает его ключевым. Так как в нашем случае мы ввели ключевое поле (*Код дисциплины*), но не указали это, а собирались сделать это впоследствии, то мы должны выбрать кнопку **Нет**. Кнопка **Отмена** оставляет окно ввода данных в таблицу открытым.

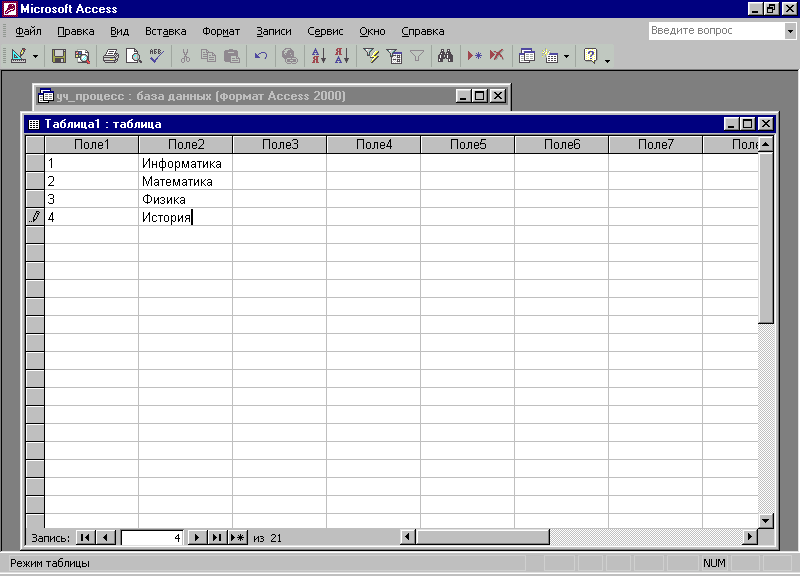


Рис. 9

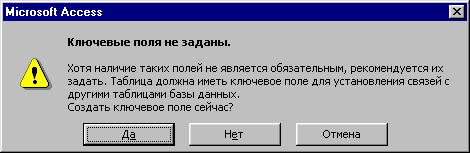


Рис. 10

## Создание таблицы с помощью мастера таблиц

Мастер создания таблиц автоматически создает таблицу по одному из шаблонов. Начать работу мастера можно выбрав строку **Создание таблицы с помощью мастера** в рабочем поле объекта **Таблицы** окна базы данных или строку **Мастер таблиц** в окне **Новая таблица**. Мастер автоматически создает таблицу по одному из шаблонов. Пользователю предлагается для выбора более 50 образцов таблиц, предназначенных для использования в различных целях (Рис. 11). Каждая таблица шаблона содержит соответствующий набор полей, из которых пользователь может выбрать нужные для включения в создаваемую таблицу.

Включаемые в таблицу поля могут быть переименованы. Мастер определит ключ таблицы. Может быть создана связь новой таблицы с уже существующими в БД. При этом ключ новой таблицы будет включен в таблицу, с которой устанавливается связь. По запросу пользователя мастер создает форму для ввода данных в таблицу. После создания таблицы мастером можно в любое время доработать структуру таблицы в режиме конструктора.

Определив структуру, можно приступить к вводу данных в нее. Непосредственный ввод данных в таблицу осуществляется в *режиме таблицы*. Переход в этот режим из окна БД выполняется нажатием кнопки **Открыть** при выделенном объекте **Таблица** в окне БД. Переход в режим таблицы из режима конструктора осуществляется нажатием кнопки **Вид** на панели инструментов или выполнением команды **Вид/Режим таблицы**. Для удобства работы с таблицей можно изменить ее представление на экране. При этом можно менять ширину столбцов, высоту строки, шрифт данных таблицы, цвет текста, линии сетки и фона, оформление, которое может быть обычным, приподнятым или утопленным. Можно выводить на экран только те столбцы, которые нужны для текущей работы, или зафиксировать столбец при просмотре широких таблиц. Настройка макета выполняется в режиме таблицы.

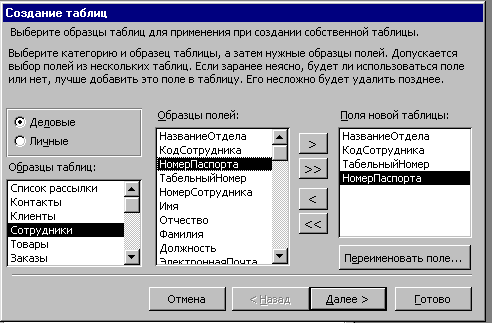


Рис. 11

# Поиск, сортировка и фильтрация данных

Обработка данных в режиме таблицы и формы включает просмотр, добавление и удаление записей, обновление полей и множество других операций. Для выполнения основных работ в режиме таблицы и режиме формы предназначены соответствующие панели инструментов, которые могут быть дополнены и настроены для пользователя.

### 

### 6.1 Поиск и замена значений в полях

Для поиска значений в текущем или во всех полях таблицы необходимо в режиме таблицы или формы воспользоваться командой меню **Правка/Найти**. В появившемся диалоговом окне **Поиск и замена** (Рис. 12) на вкладке **Поиск** в поле **Образец** вводится значение, которое требуется найти в полях таблицы. В раскрывающемся списке **Поиск в** выбирается область поиска. Если в таблице не выделен ни один из столбцов, этот список позволяет выбрать в качестве области поиска поле, в котором установлен курсор, или таблицу целиком. Если выделено несколько столбцов, то этот список не активен и поиск автоматически производится только в выделенных столбцах.

В раскрывающемся списке **Совпадение** можно выбрать степень совпадения значений: **С любой часть поля, Поля целиком, С начала поля**. Раскрывающийся список **Просмотр** позволяет выбрать направление поиска: **Все, Вверх, Вниз**. Если при сравнении требуется учитывать регистр и формат поля, необходимо, соответственно, установить флажки **С учетом регистра** и **С учетом формата полей**. После нажатия кнопки **Найти далее** отыскивается первое значение, удовлетворяющее условию поиска. Для получения следующего значения, удовлетворяющего условию поиска, необходимо продолжить поиск, нажимая кнопку **Найти далее.** Образец поиска может быть задан полным значением поля или его частью. При этом могут использоваться операторы шаблона.

Например, для поиска преподавателя с фамилией «Орлова» введем в окно **Образец** фамилию *Орлова,* в строке параметра **Просмотр** оставим *Все* (т.е. поиск по всем записям таблицы), в строке параметра **Совпадение** выберем *С любой частью поля*. После нажатия кнопки **Найти далее** курсор установится на запись, которая содержит в поле **Фамилия** значение *Орлова*. Повторное нажатие кнопки **Найти далее** приведет к появлению сообщения, что такой фамилии больше нет.

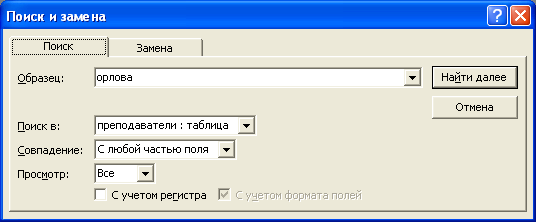


Рис. 12

Если в диалоговом окне выбрать вкладку **Замена**, оно дополнится раскрывающимся списком **Заменить на**, позволяющим ввести в найденные поля новое значение (Рис. 13). Кроме того, появятся две дополнительные кнопки, позволяющие управлять процессом замены.

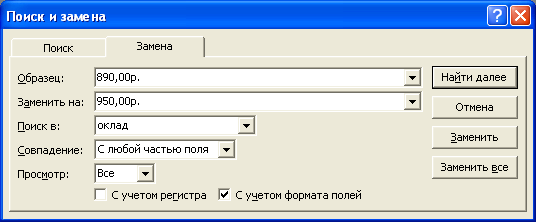


Рис. 13

Например, для замены оклада 890р. на 950р. на вкладке введем эти цифры в соответствующие поля. После нажатия кнопки **Найти далее** курсор переместится на первую запись с окладом 890р. Если нужно поменять 890 на 950 во всех записях таблицы, то нажимаем кнопку **Заменить все**, в противном случае выбираем **Заменить** или **Найти далее.**

### 6.2 Сортировка записей

Сортировка записей по возрастанию или убыванию значений одного поля может быть выполнена как в режиме таблицы, так и в режиме формы. Для этого курсор устанавливается в поле сортировки и нажимается соответствующая кнопка на панели инструментов:

|  |  |
| --- | --- |
|  | - **Сортировка по возрастанию** |
|  | - **Сортировка по убыванию** |

Для сортировки записей можно также использовать команду меню **Записи/Сортировка**.

Для сортировки записей по нескольким полям необходимо выделить соответствующие столбцы и выполнить команду сортировки. Сортируемые поля должны быть смежными, при этом старшим полем сортировки является поле, расположенное слева. Разместить поля таблицы в нужном порядке можно путем их выделения и перетаскивания при помощи мыши.

### 6.3 Использование фильтра

Для просмотра и корректировки записей БД, удовлетворяющих указанным пользователем условиям отбора, предусмотрена фильтрация таблицы. *Фильтр* – это набор условий, применяемый для отбора подмножества записей из таблицы, формы или запроса.

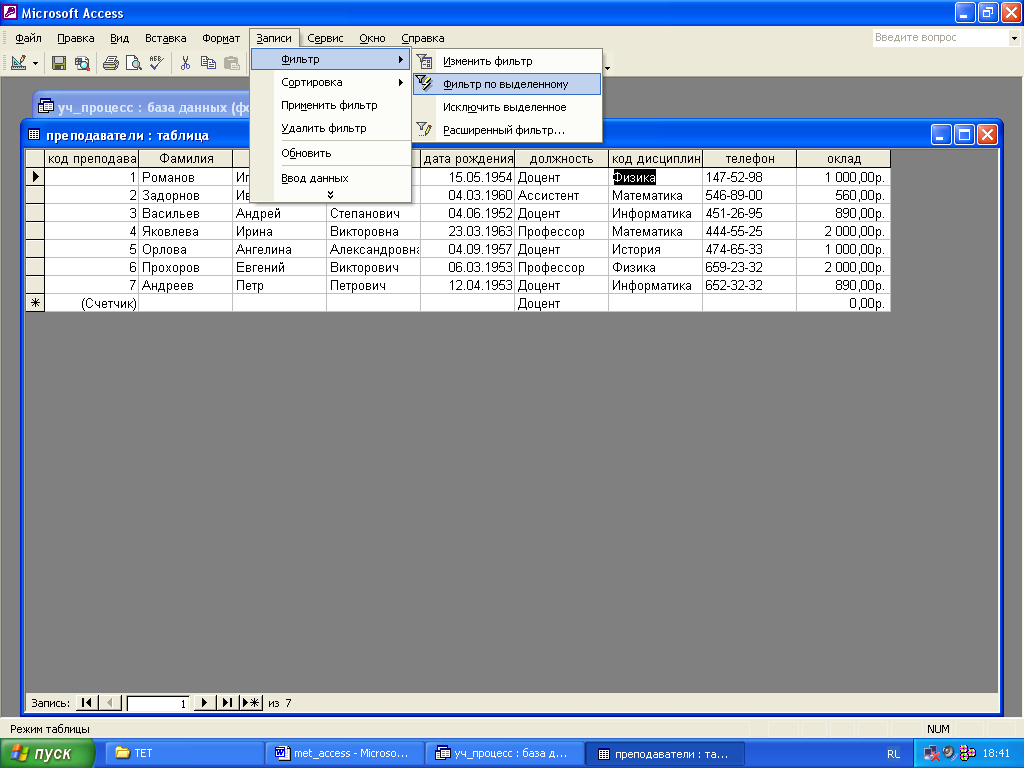


Рис. 14

Простейшим способом задания условия отбора записей является выделение в таблице или форме некоторого значения поля или его части (*фильтр по выделенному*). Для этого необходимо открыть таблицу в режиме таблицы и в поле, по которому надо сделать фильтр, выделить значение поля или его часть и выбрать команду **Записи/Фильтр/Фильтр по выделенному** или после выделения нажать правую кнопку мыши и в контекстном меню выбрать команду **Фильтр по выделенному** (Рис. 14).Для отбора записей, поля которых не содержат выделенного значения, необходимо после выделения значения выбрать команду **Записи/Фильтр/Исключить выделенное**.

В нашем примере в таблице *Преподаватели* в поле *Код дисциплины* выделим слово **Физика**, далее применим (по описанному выше правил) **Фильтр по выделенному**. В результате в окне таблицы останутся только те записи, кодом дисциплины в которых является **Физика** (см.Рис. 15)

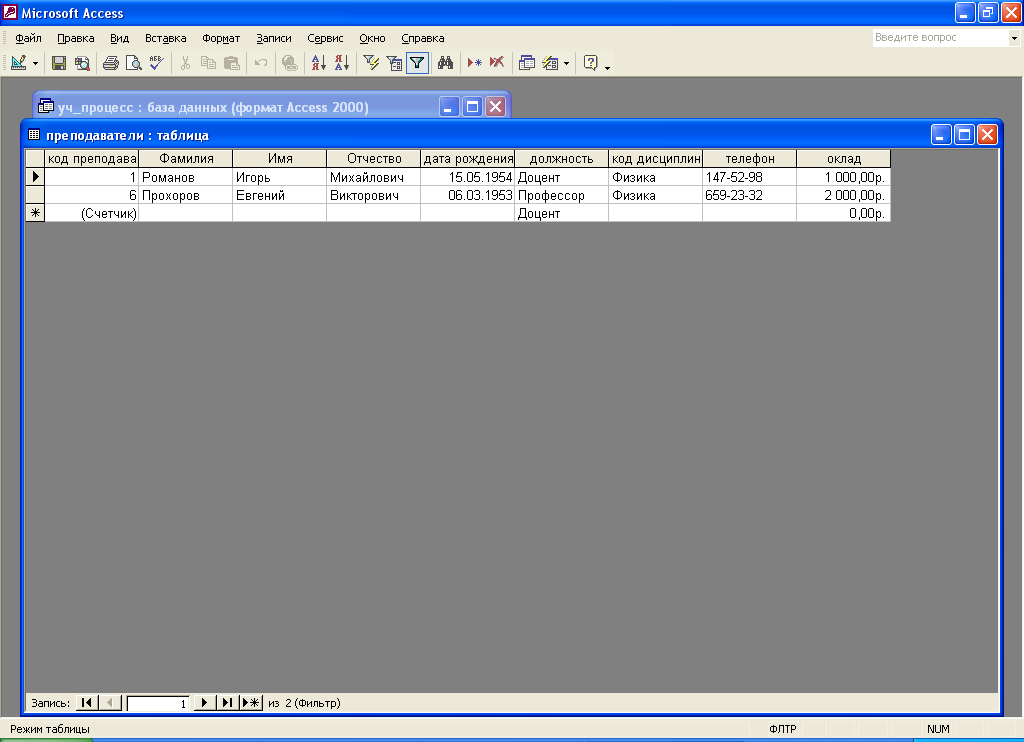


Рис. 15

Если необходимо работать со всеми записями таблицы, то действие фильтра можно отменить, воспользовавшись командой **Записи/Удалить фильтр**. При этом фильтр снимается с таблицы, но сохраняется, и в любой момент его можно применить повторно, используя команду **Записи/Применить фильтр**. Последний созданный командой **Фильтр по выделенному** или **Исключить выделенное** фильтр сохраняется автоматически при сохранении таблицы. При последующих открытиях таблицы этот фильтр может быть применен снова.

Кроме фильтра по выделенному можно использовать и другие: обычный и расширенный.

*Обычный фильтр* вызывается командой **Записи/Фильтр/Изменить фильтр**, *расширенный фильтр* – командой **Записи/Фильтр/Расширенный фильтр**.

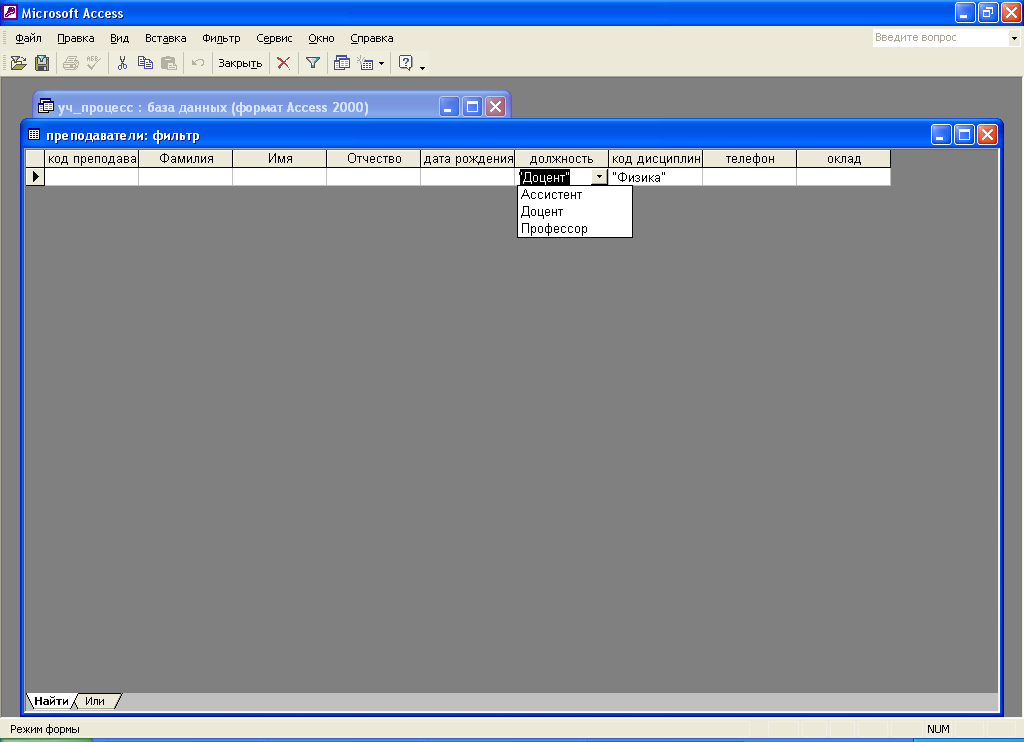


Рис. 16

После выполнения команды **Записи/Фильтр/Изменить фильтр** в окне обычного фильтра на экран выводится пустая таблица или форма для активного объекта БД. На вкладке **Найти** в поля фильтра вводятся значения, по которым будут отбираться записи. На Рис. 16 фильтрация осуществляется по значению **Физика** в поле *Код дисциплины*, а также по значению **Доцент** в поле *Должность*. Значения могут вводиться с клавиатуры или выбираться из списка, как это показано для поля *Должность*. Для того, чтобы указать альтернативный вариант отбора записей, включаемых результирующий набор, выбирается вкладка **Или**. При этом открывается новое поле фильтра для задания альтернативного варианта. Набор условий, введенных в разные поля, связывается логической операцией *ИЛИ*. Фильтр будет отбирать записи, содержащие в полях все значения, указанные на вкладке **Найти**, и записи, содержащие в полях все значения, указанные на вкладке **ИЛИ**. Для выполнения фильтрации нажмите кнопку **Применение фильтра**.

*Расширенный фильтр* вызывается командой **Записи / Фильтр / Расширенный фильтр**. При этом в окне расширенного фильтра (Рис. 17) выводится список полей активной таблицы. В нижней части окна выводится *бланк запроса*. В строку бланка запроса **Поле** из списка перетаскиваются мышью роля, по которым необходимо задать условия отбора записей. Кроме того, бланк запроса позволяет выбрать тип сортировки для одного или нескольких выбранных полей.

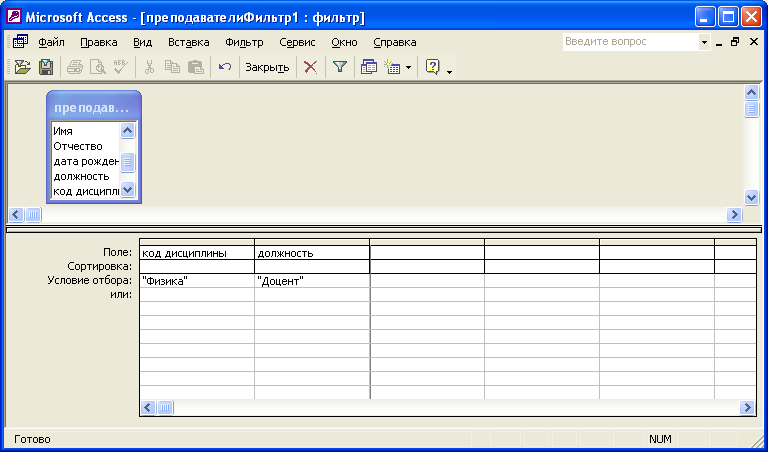


Рис. 17

Обычный и расширенный фильтры, так же как и фильтр по выделенному, сохраняется автоматически при сохранении таблицы, запроса или формы. При повторных открытиях объекта этот сохраненный фильтр является текущим и может использоваться по команде **Записи/Применение фильтра**. Если создается новый фильтр, он заменяет ранее сохраненный. Для уничтожения фильтра на панели инструментов **Фильтр** нажмите кнопку **Очистить бланк** и затем кнопку **Применение фильтра**. Только в этом случае фильтр будет уничтожен.

Для сохранения подготовленного фильтра нужно выполнить **Записи/Фильтр/Изменить фильтр** и нажать кнопку **Сохранить как запрос** на панели инструментов. В открывшемся окне **Сохранение в виде запроса** необходимо ввести имя сохраняемого запроса. В дальнейшем, вместо того, чтобы создавать фильтр заново, можно использовать существующий фильтр. Для этого в режиме **Изменить фильтр** воспользоваться командой **Загрузить из запроса.**

# Построение реляционной базы данных и разработка инфологической модели

Одно из важнейших достоинств реляционных баз данных, как говорилось ранее, состоит в том, что можно хранить логически сгруппированные данные в разных таблицах и задавать связи между ними, объединяя их в единую базу. Для задания связи таблицы должны иметь поля с одинаковыми именами или хотя бы с одинаковыми форматами данных. Связь между таблицами устанавливает отношения между совпадающими значениями в этих полях. Такая организация данных позволяет уменьшить избыточность хранимых данных, упрощает их ввод и организацию запросов и отчетов.

В Access можно задать три вида связи между таблицами: *Один-ко-многим, Многие-ко-многим, Один-к-одному*. Связь *Один-ко-многим* – наиболее часто используемый тип связи между таблицами. В такой связи каждой записи таблицы А может соответствовать несколько записей таблицы В (поля с этими записями называют внешними ключами), а запись таблицы В не может иметь более одной соответствующей ей записи в таблице А. При связи *Многие-ко-многим* одной записи таблицы А может соответствовать несколько записей в таблице В, а одной записи таблицы В – несколько записей таблицы А. Такая схема реализуется только с помощью третьей (связующей) таблицы, ключ которой состоит по крайней мере из двух полей, одно из которых является общим с таблицей А, а другое – общим с таблицей В. При связи *Один-к-одному* запись в таблице А может иметь не более одной связанной записи в таблице В и наоборот. Это тип связи используют не очень часто, поскольку такие данные могут быть помещены в одну таблицу. Связь с отношением *Один-к-одному* применяют для разделения очень широких таблиц, для отделения части таблицы в целях ее защиты, а также для сохранения сведений, относящихся к подмножеству записей в главной таблице.

Рассмотрим создание модели на примере следующих таблиц: *Преподаватели, Студенты, Дисциплины* и *Оценки*. Структура первой таблицы описана в Таблица 1., структура таблицы *Дисциплины* представлена на Рис. 9, структура двух оставшихся таблиц представлена в Таблица 2 и Таблица 3.

Таблица 2

|  |  |  |
| --- | --- | --- |
| Имя поля | Тип данных | Размер поля |
| Код студента | Счетчик |  |
| Фамилия | Текстовый | 15 |
| Имя | Текстовый | 15 |
| Отчество | Текстовый | 15 |
| Номер группы | Числовой | Целое |
| Телефон | Текстовый | 9 |
| Стипендия | Логический | Да/Нет |

Таблица 3

|  |  |  |
| --- | --- | --- |
| Имя поля | Тип данных | Размер поля |
| Код студента | Числовой | Длинное целое |
| Код дисциплины | Числовой | Длинное целое |
| Оценки | Числовой | Байт |

Рассмотрим связь между объектами *Студенты* и *Дисциплины.* Студент изучает несколько дисциплин, что соответствует многозначной связи. Понятно, что каждая дисциплина изучается множеством студентов. Это тоже многозначная связь. Таким образом, связь между *Студенты* и *Дисциплины* – *Многие-ко-многим* (M : N). Множественные связи усложняют управление базой данных (в Access при такой связи нельзя использовать механизм каскадного обновления), поэтому использовать такие связи нежелательно и нужно строить реляционную модель, не содержащую связей типа *Многие-ко-многим*. Для свзи будем использовать объект *Оценки*. Каждый студент имеет оценки по нескольким дисциплинам, поэтому связь между *Студенты* и *Оценки* будет *Один-ко-многим* (1 : M). Каждую дисциплину сдает множество студентов, поэтому связь между *Дисциплины* и *Оценки* также будет *Один-ко-многим* (1 : M). В результате получаем информационно-логическую модель базы данных, приведенную на Рис. 18

Студенты Дисциплины Преподаватели

Оценки

Рис. 18

Для создания логической модели нужно после создания, по крайней мере, структур таблиц в окне базы данных нужно выполнить команды **Сервис/Схема данных** или щелкнуть на панели инструментов иконку . После этого откроется пустое окно **Схема данных**. Для добавления в схему таблиц нужно выполнить команды **Связи/Добавить таблицу** или щелкнуть на панели инструментов соответствующую иконку . В окне **Добавление таблицы** выбираем нужную таблицу (делаем текущей) и нажимаем кнопку **Добавить**. После добавления всех нужных таблиц закрываем окно нажатием кнопки **Закрыть**. После добавления таблиц в схему установим связи между ними. Для этого разместим таблицы в нужном порядке (установим курсор мыши в заголовок таблицы и удерживая кнопку мыши нажатой перетаскиваем ее). Далее выделяем в таблице поле для связи и опять же, удерживая кнопку мыши нажатой, тянем ее к соответствующему полю другой таблицы. На экране появляется окно **Изменение связей**, где определяется тип отношения (см.Рис. 19), указывается, нужно ли обеспечивать целостность данных, а также, нужно ли каскадное обновление связанных полей и каскадное удаление связанных полей. После нажатия кнопки **Создать** это окно закрывается, а в схеме данных появляется «связь». После создания всех связей схема данных выглядит, как указано на Рис. 20

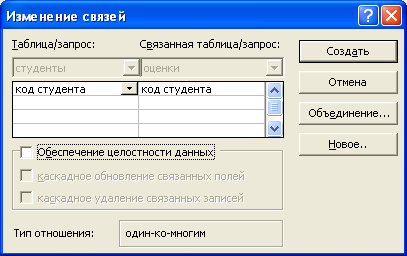


Рис. 19

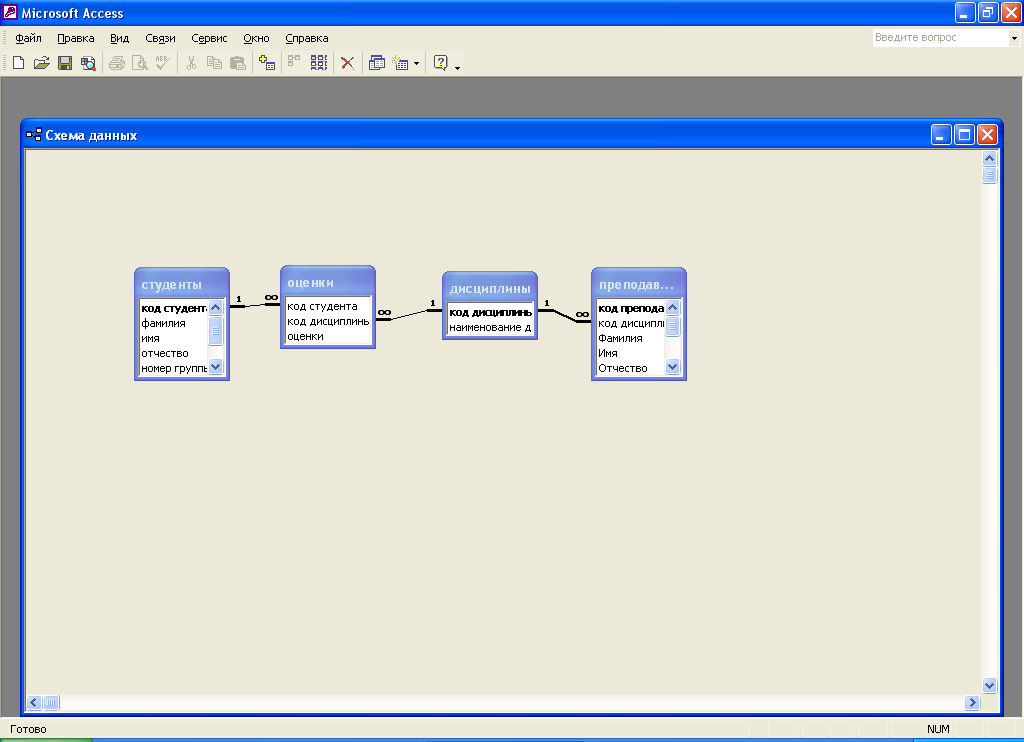


Рис. 20

Созданная схема данных будет в дальнейшем использоваться для создания форм, запросов и отчетов.

# Разработка форм

Основой диалогового интерфейса пользователя для работы с БД являются формы. Через формы может осуществляться первоначальная загрузка данных во взаимосвязанные таблицы БД, просмотр данных, а также их корректировка. Работая с формой, можно добавлять и удалять записи из таблицы, изменять значения полей, получать расчетные данные. В процессе работы может осуществляться контроль вводимых данных. Access позволяет создавать однотабличные формы (на основе одной таблицы) и многотабличные (для работы с несколькими взаимосвязанными таблицами). Создавать формы можно в режиме *Конструктора* и с помощью *Мастера форм.*. Рассмотрим создание формы для ввода/вывода данных с помощью *Мастера форм.*. С его помощью можно легко получить типовой макет формы, а далее доработать ее средствами *Конструктора форм.*

## 

## 8.1 Разработка формы с помощью мастера форм

Предположим, нам надо создать форму для ввода данных в таблицу Преподаватели. Чтобы начать создание формы, в окне БД (см.Рис. 3) нужно выбрать строку **Формы**. Для получения доступа к списку всех вариантов создания новой формы необходимо на панели инструментов окна БД нажать кнопку **Создать**. Для вызова *Мастера форм* в окне **Новая форма** выбираем соответствующую строку (Рис. 21).

Далее в окне для выбора источника нужно выбрать таблицу, для которой создается форма. После выбора таблицы запускаем **Мастер форм**. Появляется окно **Создание форм** (Рис. 22), в котором Access предлагает выбрать те поля, которые должны присутствовать в форме. Для этого нужно сделать текущим соответствующее поле и кнопкой со стрелкой перенести его в **Выбранные поля**. Если необходимо поместить в форме все поля таблицы (как в нашем случае), то надо нажать кнопку .

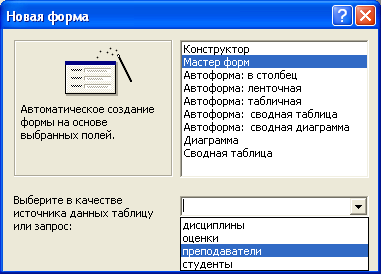


Рис. 21

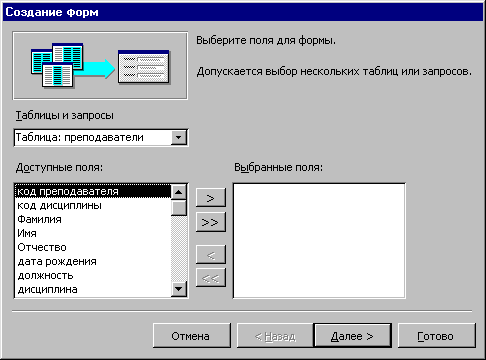


Рис. 22

В результате описанных выше действий окно **Создание форм** принимает вид, показанный на Рис. 23. для продолжения построения формы нажимаем кнопку **Далее**.

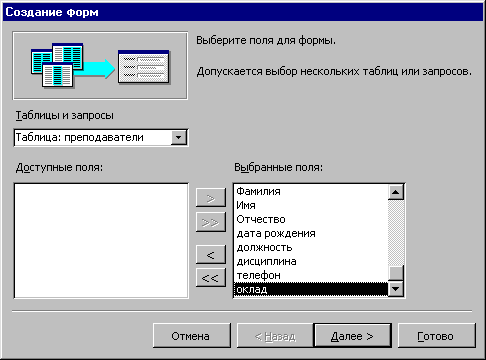


Рис. 23

В следующем окне Access предлагает выбрать внешний вид формы: в один столбец, ленточный, табличный, выровненный, сводная таблица или сводная диаграмма. В левой части окна предлагается образец выбранного внешнего вида. Мы выберем первый вариант – в один столбец (см.Рис. 24)

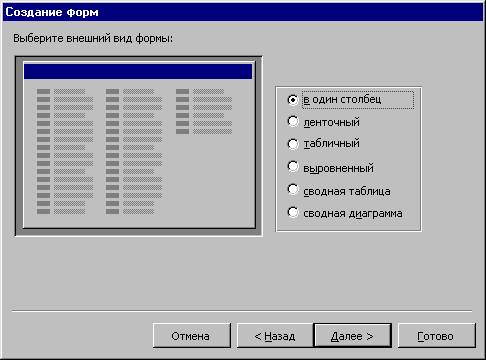


Рис. 24

При необходимости можно вернуться на шаг назад, нажав кнопку **Назад** или прервать создание формы, нажав кнопку **Отмена**. Мы выбираем кнопку **Далее** и выбираем стиль формы. Перечень стилей представлен на Рис. 25, в левой части опять присутствует образец выбранного стиля. Мы выбираем стиль *Стандартный*. И теперь остается последний шаг: задать имя формы (Рис. 26). На этом построение формы закончено.

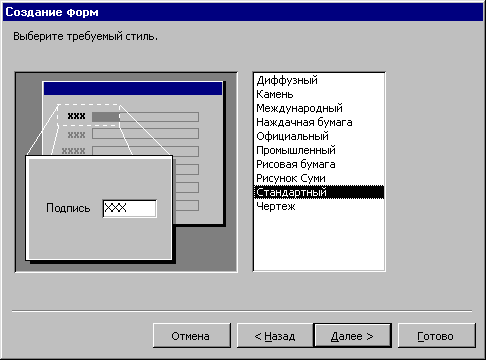


Рис. 25

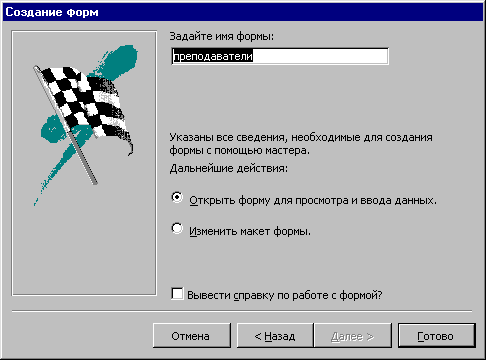


Рис. 26

Созданная форма представлена на Рис. 27. При необходимости ее можно изменить в режиме конструктора, передвигая и изменяя размеры элементов - заголовков полей, ячеек для ввода данных.

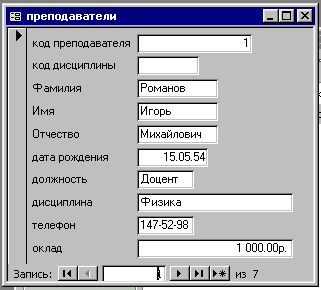


Рис. 27

В режиме формы, построенной по описанным выше правилам, можно выполнять поиск, сортировку, фильтрацию и выполнять корректировку данных. С помощью кнопок в нижней части формы можно переходить от одной записи к другой.

## 8.2 Разработка многотабличных форм

Теперь рассмотрим создание более сложной формы. Создадим форму, в которой с названиями дисциплин была бы связана подчиненная форма *Студенты* и подчиненная форма *Оценки.* Для этого выполним команду **Создать** в окне БД и текущей строке **Формы**. Выберем **Мастер форм** и нажмем кнопку **Ок**, не выбирая конкретную таблицу (Рис. 21). Выберем из таблицы Дисциплины поле *Название* дисциплины, из таблицы *Студенты –* поля *Код студента, Фамилия, Имя,* Отчество и *Номер* группы, из таблицы Оценки – поле *Оценки.* В следующем окне выберем вид представления данных (в левом окне текущим сделаем дисциплины). Выбрав таким же образом табличный вид подчиненной формы, стандартный стиль и задав имя формы, получим форму, представленную на Рис. 28. В представленной форме можно вводить оценки студентов по указанным дисциплинам. После создания такой формы в перечне форм в окне БД появится два новых имени: *Дисциплины* и *Оценки подчиненная форма*.

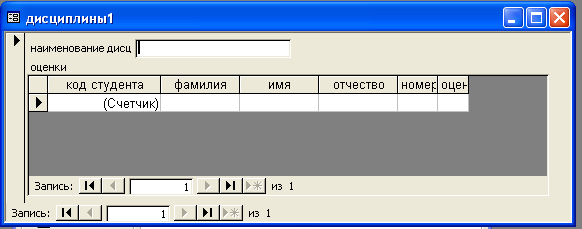


Рис. 28

# Запросы

Запросы являются одним из основных инструментов выборки записей, обновления таблиц и обработки данных в таблицах БД. Результат выполнения запроса – это новая, чаще всего временная таблица, которая существует до закрытия запроса. Структура этой таблицы определяется выбранными полями. Условия отбора, сформулированные в запросе, позволяют фильтровать записи, составляющие результат объединения таблиц. Простейшие запросы могут быть созданы с помощью мастера. Любой запрос можно создать в режиме конструктора. В Access может быть создано несколько видов запроса:

* *Запрос на выборку* – выбирает данные из взаимосвязанных таблиц и других запросов; результатом является таблица, которая существует до закрытия запроса; на основе этого вида могут строиться запросы других видов.
* *Запрос на создание таблицы* – выбирает данные из взаимосвязанных таблиц и других запросов и сохраняет результат в новой постоянной таблице.
* *Запрос на обновление, добавление, удаление* – запросы действия, в результате выполнения которых изменяются данные в таблицах.

Лучшим способом создания запроса является использование графического конструктора. Основные принципы конструирования различных запросов заложены в технике конструирования *запроса на выборку,* являющегося основой всех видов запросов.

## 9.1 Разработка запроса в режиме конструктора

Для создания запроса необходимо в окне БД выбрать строку **Запросы** и выбрать на панели иконку **Создать.** В открывшемся окне **Новый запрос** выбрать строку **Конструктор (**Рис. 29**).**

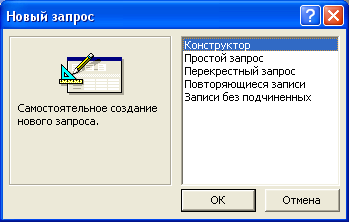


Рис. 29

Далее в открывшемся окне **Добавление таблицы** нужно выбрать используемые в запросе таблицы и нажать кнопку **Добавить** (Рис. 30). Для закрытия окна нажать кнопку **Закрыть**.

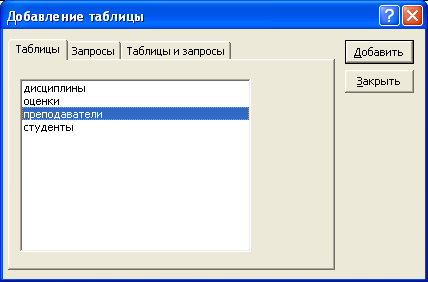


Рис. 30

В результате появится окно конструктора запросов – **Запрос № : запрос на выборку** (Рис. 31). Окно конструктора запросов разделено на две панели: *схему данных запроса,* которая включает выбранные таблицы, и *бланка запроса* по образцу, который нужно заполнить.

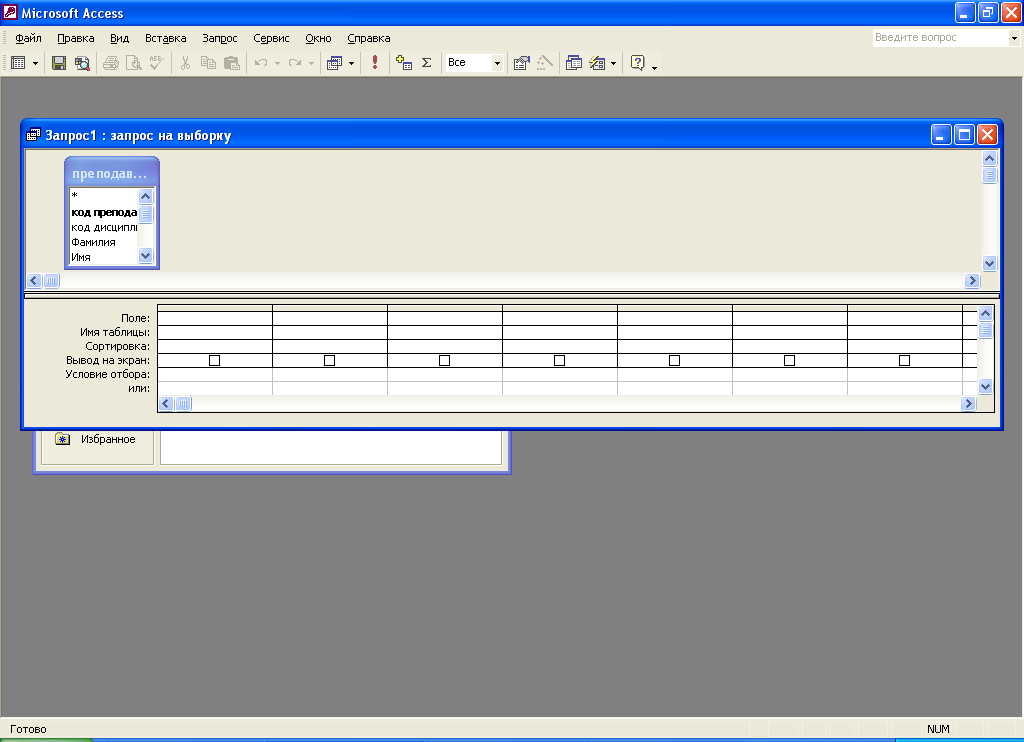


Рис. 31

Как конструировать запрос рассмотрим на примере: из таблицы *Преподаватели* отберем поля фамилии, имени, отчества и их должности. Для включения требуемых полей таблицы в соответствующие столбцы запроса можно воспользоваться следующими приемами:

* В первой строке бланка запроса **Поле** щелчком мыши вызвать появление кнопки списка и выбрать из списка нужное поле; список содержит все поля таблицы, выбранной в запрос.
* Переместить с помощью мыши требуемое поле из списка полей таблицы в схеме данных запроса в первую строку бланка запроса.
* Дважды щелкнуть на имени поля таблицы в схеме данных запроса.
* Для включения в запрос всех полей таблицы можно дважды щелкнуть на символе **\*** (звездочка) в этом списке.

Выбрав нужные поля одним из описанных выше способов, получим сформированный запрос (Рис. 32).

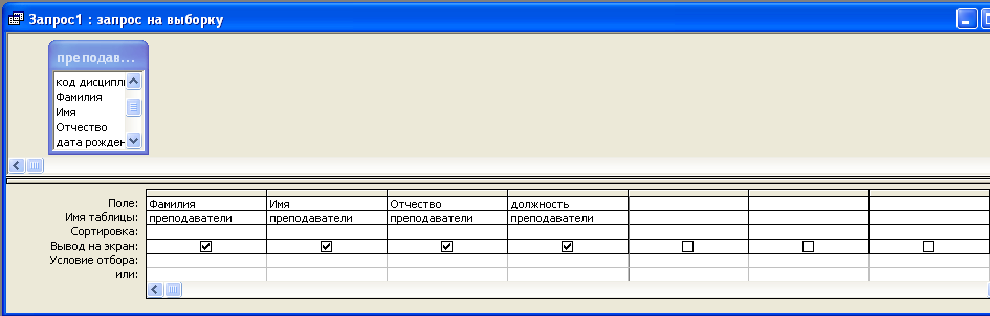


Рис. 32

Выполнить сформированный запрос можно выполнив команды меню **Запрос/Запуск**. В результате на экране появится таблица, соответствующая нашим условиям (Рис. 33).

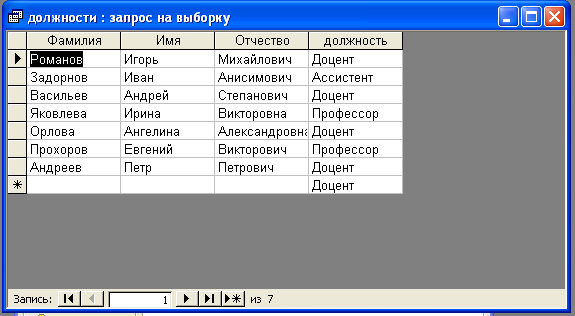


Рис. 33

Для сохранения запроса можно выполнить команды **Файл/Сохранить** или просто закрыть окно запроса. Появится вопрос, нужно ли сохранять созданный запрос. В случае подтверждения сохранения предлагается окно для ввода имени запроса.

Можно впоследствии модифицировать уже созданные запросы:

* Добавление таблицы в схему данных запроса осуществляется с помощью команды **Запрос/Добавить таблицу** или нажатием соответствующей кнопки панели **Конструктор запросов**
* Добавление поля в бланк запроса осуществляется посредством одного из рассмотренных выше действий, например, перемещением с помощью мыши имени поля из таблицы в схеме данных в нужное место бланка.
* Удаление поля в бланке запроса требует предварительного выделения соответствующего столбца. Для этого следует переместить курсор в область маркировки столбца, где он примет вид направленной вниз черной стрелки, и щелкнуть кнопкой мыши. Далее нажать клавишу «Delete» или выбрать пункт меню **Правка/удалить столбцы**.
* Для перемещения поля в бланке запроса нужно выделить с помощью мыши соответствующий столбец и переместить его на новую позицию. Столбец, на место которого перемещен новый, и все столбцы справа от него будут сдвинуты вправо.

Для отбора определенных записей из таблицы в запросе есть строка **Условие отбора**. Условием отбора является выражение, которое состоит из операторов сравнения и операндов. В качестве операндов выражения могут использоваться: литералы (конкретные значения, воспринимаемые Access так, как они записаны; в качестве литералов могут быть числа, текстовые строки, даты; текстовые строки заключаются в кавычки, даты в знаки #), константы (неизменяющиеся значения, которые определены в Access, например, True, False, Null, Да, Нет), идентификаторы (имена полей, таблиц, форм, отчетов и т.д.). Для формирования логических выражений в качестве условий отбора существует **Построитель выражений**. Для отбора из таблицы *Преподаватели* всех преподавателей, родившихся до 1960 года, запрос на выборку представлен на Рис. 34

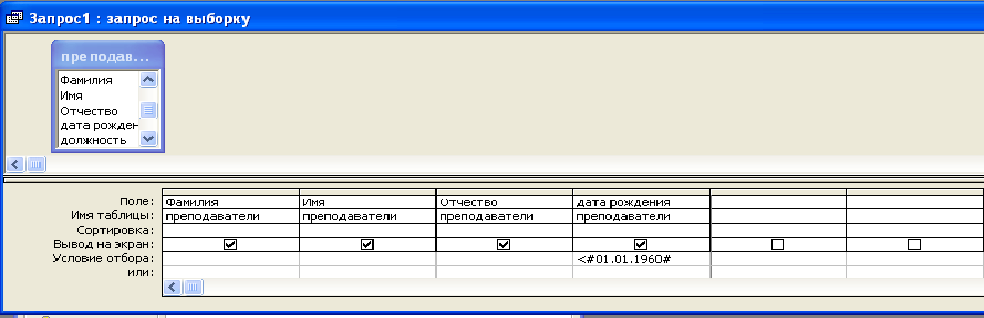


Рис. 34

По описанному выше правилу в запрос помещаем необходимые поля и в строке **Условие отбора** помещаем условие – <01/01/1960. В результате выполнения этого запроса получим новый список преподавателей (Рис. 35).

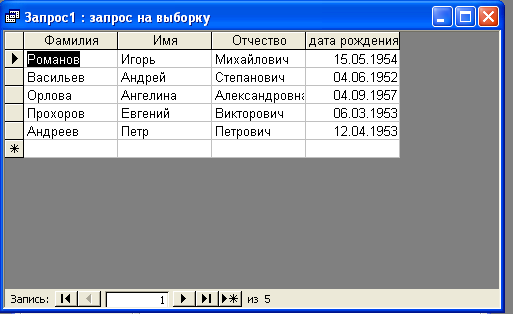


Рис. 35

## 

## 9.2 Разработка запросов на редактирование данных

### 

### 9.2.1 Запросы на обновление данных

Для обновления данных в полях таблиц может быть использован *запрос на обновление*. Изменения вносятся в группу записей, отбираемых с помощью указанных пользователем условий отбора. Значения для изменений в полях определяются в бланке запроса в поле **Обновление.** Чтобы задать запрос на обновление, необходимо сначала создать *запрос на выборку*, который затем преобразуется в окне конструктора запросов в запрос на обновление при выборе команды **Обновление** контекстного меню, появляющегося при нажатии кнопки **Тип запроса** на панели инструментов конструктора запросов, или команды меню **Запрос/Обновление**. После выполнения этой команды в бланке запроса появляется строка **Обновление**. Для отбора обновляемых записей следует включить в бланк запроса поля, требующие обновления, а также поля, по которым задаются условия отбора. Процедура задания условия отбора аналогична созданию запроса на выборку. Достаточно ввести в строку **Обновление** значение или выражение, определяющее новое значение поля. Для создания выражения можно воспользоваться построителем выражений. После выполнения команды **Запрос/Запуск** открывается диалоговое окно с сообщением о числе обновляемых записей и вопросом о продолжении операции обновления. Если необходимо просмотреть только содержимое обновленных полей, после выполнения запроса следует переключиться в режим таблицы. Для этого выберите команду **Вид/Режим таблицы**. Для просмотра старого содержимого обновляемых полей в режим таблицы необходимо переключиться до выполнения запроса. Рассмотрим этот вид запроса на примере: изменим на 10% заработную плату преподавателей, у которых она меньше 900 рублей. Для этого создадим простой запрос (Рис. 36). По описанным выше правилам из таблицы *Преподаватели* выберем поле *Оклад*.

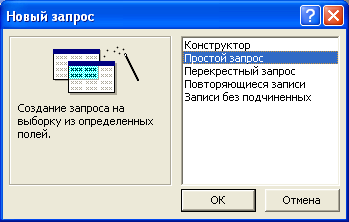


Рис. 36

Остальные параметры создания выберем по умолчанию, запрос назовем *Изменение оклада*. Далее созданный запрос откроем в режиме конструктора и введем условие отбора: <900, выполним команду меню **Запрос/Обновление**. В строке запроса **Обновление** введем *[Оклад]*\*1.1 (Рис. 37). После выполнения этого запроса появится сообщение (Рис. 38) на подтверждение обновления записей.

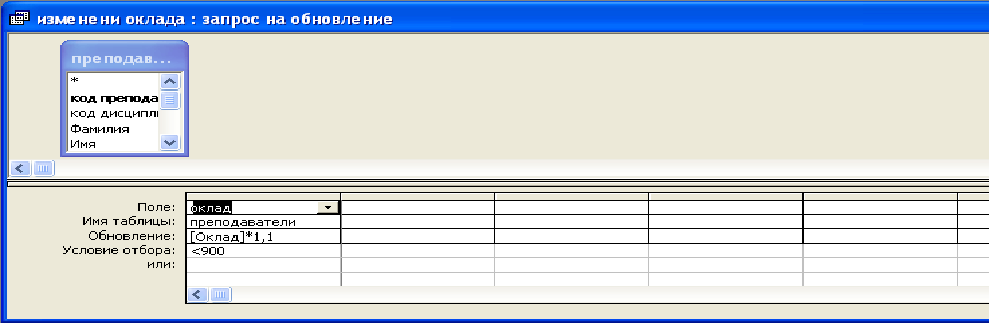


Рис. 37

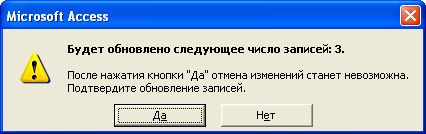


Рис. 38

После выполнения запроса и сохранения его просмотрите таблицу и убедитесь в обновлении данных.

### 9.2.2 Запросы на удаление данных

*Запрос на удаление* позволяет удалить записи из одной таблицы или нескольких взаимосвязанных таблиц, для которых установлен флажок **каскадное удаление связанных записей**. В запросе указываются таблицы, из которых должны удаляться записи, и задаются условия отбора. Первоначально запрос на удаление создается как *запрос на выборку*, затем запрос в режиме конструктора преобразуется в запрос на удаление (**Запрос/Удаление**). После преобразования в запросе появляется строка **Удаление**. Затем формируется бланк запроса. Для предварительного просмотра удаляемых записей можно нажать кнопку **Вид**. Для удаления записей запрос нужно запустить на выполнение. Следует иметь в виду, что удаленные записи нельзя восстановить, поэтому перед удалением целесообразно выполнить предварительный просмотр удаляемых записей. Предположим, из таблицы *Студенты* необходимо отчислить студента Гаврилова И.П. из группы 101. Создадим обычный запрос, включив таблицу *Студенты* и выбрав поля *Фамилия, Имя, Отчество, номер группы. С*охраним запрос с именем *Отчисленные студенты*. Далее откроем запрос в режиме конструктора и введем условия отбора, пометим запрос как запрос на удаление (**Запрос/Удаление**), в строке **Условие отбора** в соответствующих столбцах введем фамилию, имя, отчество и номер группы для отчисляемого студента (Рис. 39)

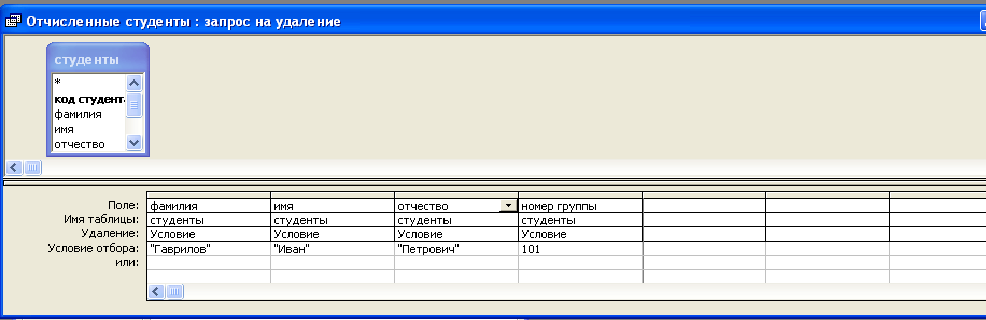


Рис. 39

После запуска запроса на выполнение Access выводит сообщение, которое представлено на Рис. 40. Пользователь должен сам выбрать свое действие.

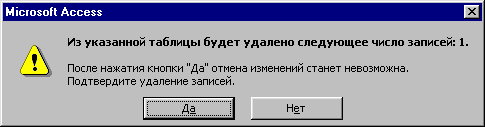


Рис. 40

### 

### 9.2.2 Запросы на добавление данных

С помощью *запроса на добавление* производится добавление записей из таблицы результата запроса в таблицу БД. Поэтому поля добавляемых записей в запросе должны соответствовать структуре существующих записей таблицы. Структура записи таблицы запроса может не полностью совпадать со структурой записи таблицы, в которую добавляются записи. В записи запроса может быть меньше полей, если на поля существующей таблицы не наложено требование обязательности их заполнения. Допускается несоответствие типов полей, если возможно преобразование типа данных одного поля в тип данных другого поля. Кроме того, в бланк запроса могут быть включены поля, по которым задаются условия отбора. Первоначально *запрос на добавление* создается как *запрос на выборку* для одной или нескольких взаимосвязанных таблиц. Затем в окне конструктора запросов нужно выполнить команды **Запрос/Добавление**, после чего на экране появляется окно для ввода имени таблицы для добавления (Рис. 41). В этом же окне можно также указать, что добавление производится в другую базу данных.

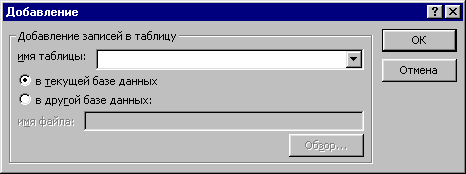


Рис. 41

После преобразования запроса в его бланке появляется строка **Добавление**. При формировании добавляемых записей следует иметь в виду, что в бланк запроса должны быть включены поля, соответствующие полям таблицы, в которую будет производиться добавление. Если в таблице, куда добавляются записи, есть ключ, ключевые поля должны быть обязательно включены в бланк запроса. Кроме того, в бланк запроса могут быть включены поля, по которым задаются условия отбора, которые вносятся в поля строки **Условия отбора**. Предположим, что нам необходимо добавить записи в таблицу *Оценки* для восстановившегося студента Петрова Петра Петровича из группы 101. Создаем простой запрос, включаем в него поля *Фамилия, имя, отчество и номер группы* из таблицы *Студенты* и поле *код дисциплины* из таблицы *Дисциплины*. В последнем окне создания запроса пометим пункт *Изменить макет запроса*, убрав галочку в строке *Открыть запрос для просмотра данных.* Выполним команду **Запрос/Добавление**, в строке *Условие отбора* введем фамилию, имя, отчество и номер группы (Петров Петр Петрович 101) и дисциплину Информатика (Рис. 42)и затем выполним запрос. Появляется окно для подтверждения добавления записей (Рис. 43).

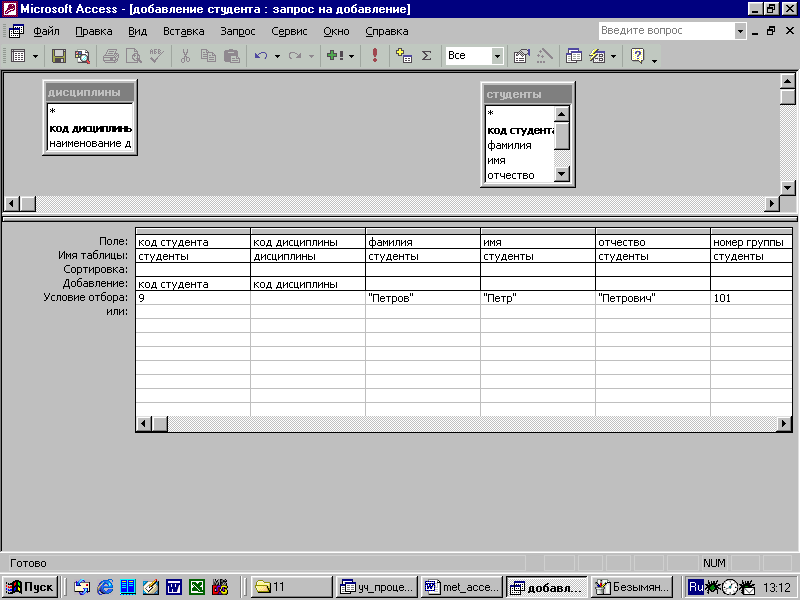


Рис. 42

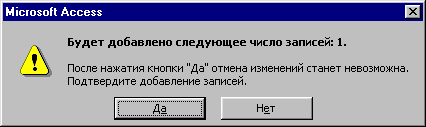


Рис. 43

В результате выполнения этого запроса в таблице *Оценки* добавится запись для нового студента по предмету *Информатика.* Таким же образом можно добавить для этого студента записи по всем предметам. Для этого в строке запроса *Условие отбора* в столбце *Наименование дисциплины* с помощью *Построителя выражений (*или с клавиатуры) ввести выражение: **Информатика or Физика or Математика or Экономика**. В этом случае будет добавлено 4 записи.

# Разработка отчетов

Средства Access по разработке отчетов предназначены для конструирования макета отчета, по которому может быть осуществлен вывод данных в виде выходного печатного документа. Перед началом конструирования отчета нужно спроектировать его макет, т.е. определить состав и содержание разделов отчета, размещение в нем значений, выводимых из полей таблиц (запросов) БД, и вычисляемых реквизитов, поля, по которым необходимо группировать данные. Для каждого уровня группировки определяются заголовки и примечания, вычисляемые итоговые значения. Кроме того, оформляются заголовки и подписи реквизитов отчета и определяется порядок вывода данных. Отчет можно создать с помощью мастера или в режиме конструктора отчетов. Иногда удобно использовать мастер отчетов. Созданный мастером отчет всегда можно доработать в режиме конструктора.

## 10.1 Создание однотабличного отчета в режиме конструктора

Создание и изменение макета отчета осуществляется в окне конструктора отчетов. При создании отчета в режиме конструктора первоначально отображаются пустые разделы отчета: заголовок отчета, верхний колонтитул, область данных. Нижний колонтитул, примечание.. Наличие этих разделов, а также их удаление или включение, определяются командами меню **Вид/Колонтитулы** и **Вид/Заголовок/Примечание отчета**. При создании отчетов необходимо заполнить эти разделы. В заголовок помещается текст из шапки макета отчета. В верхний и нижний колонтитул обычно помещают заголовки, номера страниц и даты. В области данных размещаются поля таблиц БД или запросов. При необходимости группировки записей по полю в окно конструктора отчетов могут быть добавлены разделы **Заголовок группы** и **Примечание группы**. В заголовке группы, как правило, размещаются поля, по которым производится группировка. Допускаются до 10 уровней группировки.

Рассмотрим создание простого отчета для одной таблицы на примере таблицы *Преподаватели*. Выберем в окне БД объект **Отчеты**. Далее выберем **Создание отчета в режиме конструктора/Создать**. В окне **Новый отчет** (Рис. 44) выберем **Мастер отчетов**, в строке источника выберем таблицу *Преподаватели*. Нажмем **Ок.** В окне **Создание отчетов** выберем поля для включения в отчет (фамилия, имя, отчество, дата рождения и должность). В следующем окне **Создание отчетов** нужно определить уровни группировки (определим группировку по должностям, см.Рис. 45). Далее необходимо указать сортировку для выводимых данных (определим сортировку по возрастанию для поля Имя).

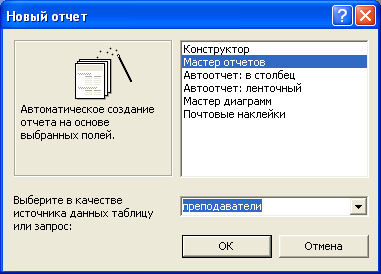


Рис. 44

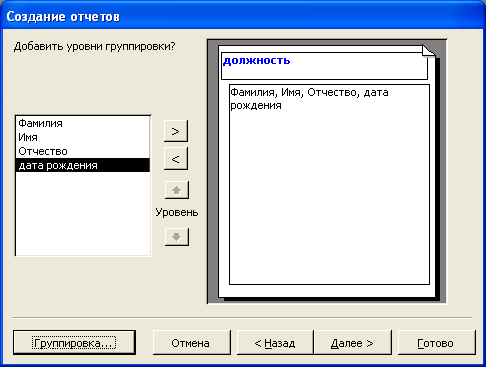


Рис. 45

Далее определим вид макета (ступенчатый), ориентацию (альбомная) и поставим галочку в окне **Настроить ширину полей для размещения на одной странице (**Рис. 46). В следующем окне определяем стиль заголовков (выберем полужирный) и далее определим заголовок отчета (зададим **Преподаватели**). В результате просмотра созданного отчета мы получим печатный документ, отображенный на Рис. 47

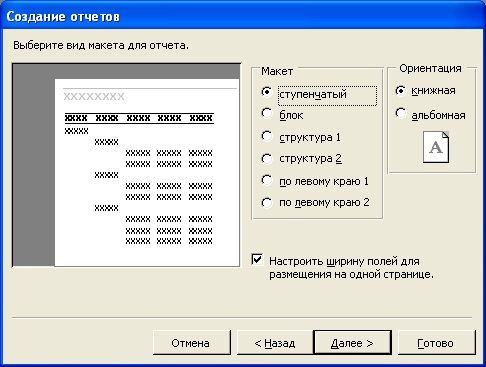


Рис. 46

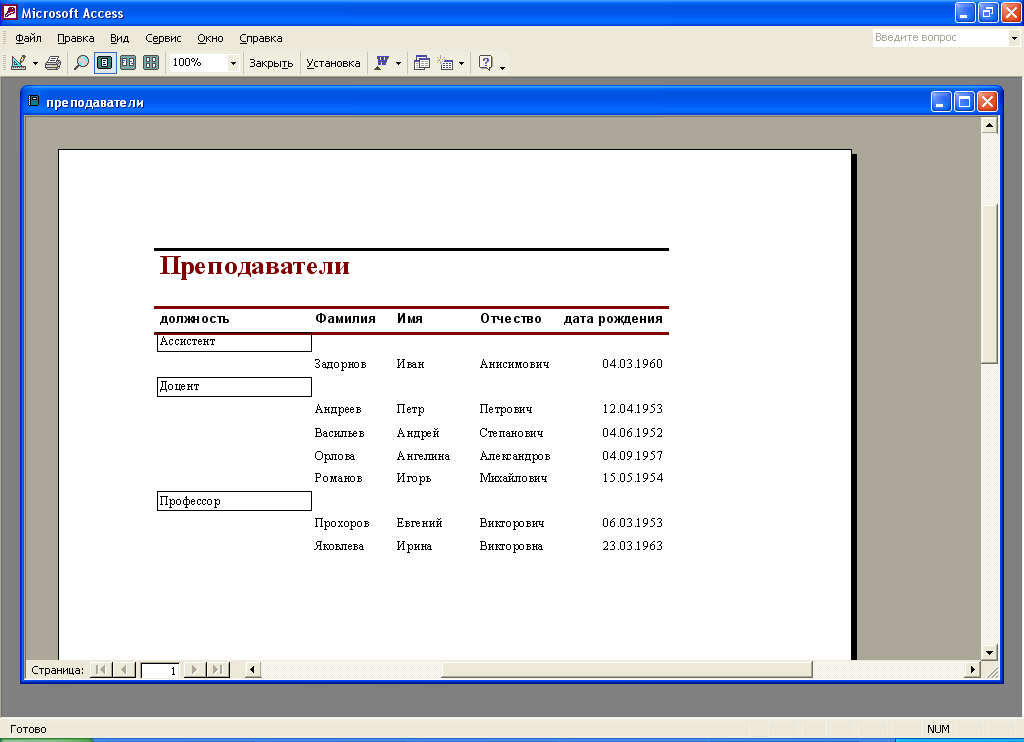


Рис. 47

## 10.2 Создание отчета на основе запроса

Для вывода в отчет определенных полей необходимо определить таблицы-источники данных и их взаимосвязи. Таким образом, сначала необходимо создать запрос на выборку. Предположим, нам необходимо вывести на печать оценки студентов обеих групп по предмету информатика. Для этого создаем запрос, включаем в него таблицы *Студенты* (добавим поля *Фамилия, Имя, Отчество, Номер группы*)*, Дисциплины* (*Информатика*) и *Оценки* (*Оценки*). Сохраним его с именем *Ведомость с оценками* и откроем его в режиме конструктора (Рис. 48).

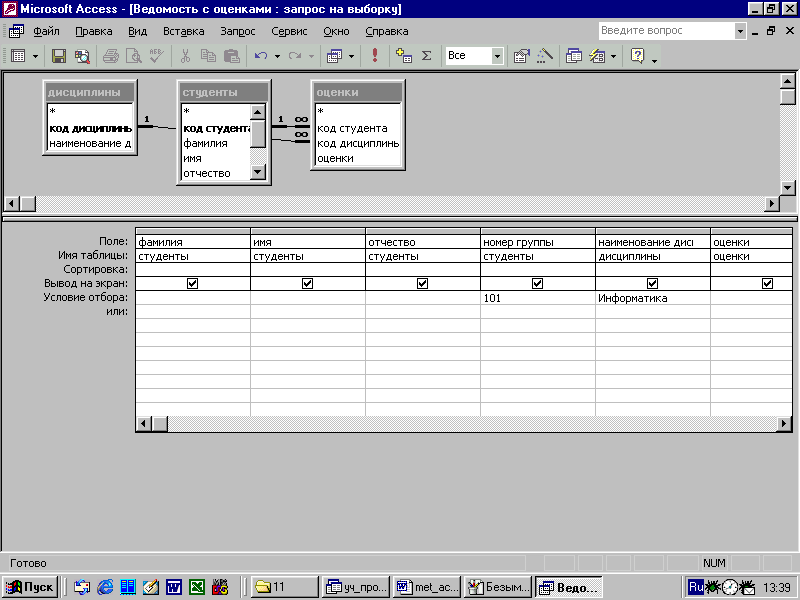


Рис. 48

В строке *Условие отбора* в столбце *Наименование дисциплины* – **Информатика**. Сохраним введенные изменения, а также можем просмотреть результат выполнения запроса. Далее в окне БД сделаем текущей строку **Отчеты**, далее **Создание отчета в режиме конструктора/Создать**.

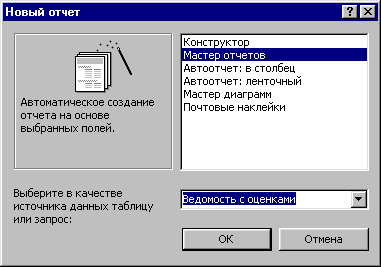


Рис. 49

В окне **Новый отчет** выбираем **Мастер отчетов**, в качестве источника выбираем запрос **Ведомость с оценками**. Далее выбираем все поля для вывода, затем вид представления данных (чтобы в заголовок отчета попала дисциплина, делаем в левом окне текущей таблицу *Дисциплины*, см.Рис. 50).

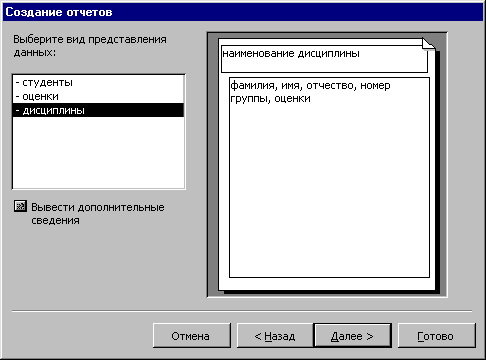


Рис. 50

В следующем окне определим уровень группировки по группе (Рис. 51).

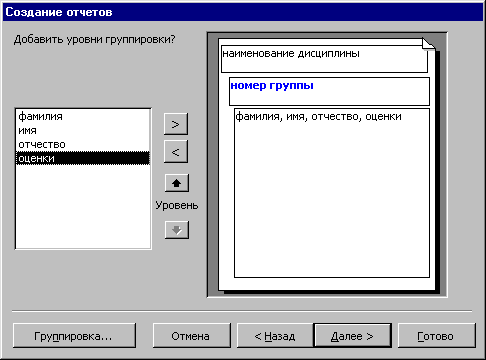


Рис. 51

В следующем окне укажем сортировку по фамилии, далее вид макета - ступенчатый, заголовок – полужирный, имя дадим **Ведомость 1.** После запуска на просмотр получим следующий вид (см.Рис. 52).

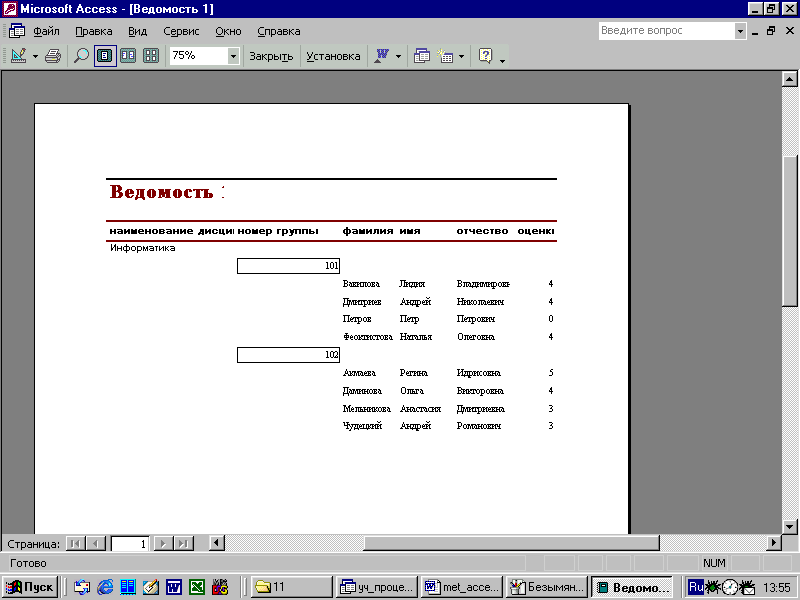


Рис. 52

# Задание к лабораторной работе

Далее представлены задания на лабораторную работу. Номер задания соответствует варианту.

1. **Библиотечный каталог**

Основные таблицы:

* авторы (фамилия, имя, отчество);
* книги (наименование, ключевые слова, количество страниц);

Правила:

* у книги может быть несколько авторов;
* у одного автора может быть несколько книг;

Создать базу данных (по крайней мере, 3 таблицы), создать формы для ввода данных, заполнить таблицы, создать запрос на добавление, на выборку, на основе запроса на выборку создать отчет, создать отчет по одной из таблиц.

1. **Сессия**

Основные таблицы:

* студенты (фамилия, имя, отчество, попадание в приказ об отчислении);
* предметы (наименование, статус (экзаменов не было, экзамены начались, экзамены закончились));

Правила:

* каждый студент может иметь оценки по нескольким предметам, каждый предмет сдают несколько студентов;
* экзаменов по предмету не было, если ни один студент не имеет по нему оценок;
* экзамены по предмету закончились, если все студенты имеют по нему оценки;

Создать базу данных (по крайней мере, 3 таблицы), создать формы для ввода данных, заполнить таблицы, создать запрос на обновление, на выборку, на основе запроса на выборку создать отчет, создать отчет по одной из таблиц.

1. **Знание иностранных языков**

основные таблицы:

* люди (фамилия, имя, отчество, статус полиглота);
* языки (наименование);

правила:

* каждый человек может знать несколько языков на каком-то уровне (от 1 до 5).
* статус полиглота обеспечивается средним баллом знания всех языков =4.8

Создать базу данных (по крайней мере, 3 таблицы), создать формы для ввода данных, заполнить таблицы, создать запрос на удаление, на выборку, на основе запроса на выборку создать отчет.

1. **Администрирование пользователей**

Основные таблицы:

* пользователи (фамилия, имя, отчество, логин, пароль);
* группа пользователей (наименование);
* приложение (наименование);

Правила:

* каждый пользователь может входить в несколько групп пользователей;
* каждая группа пользователей может содержать несколько пользователей;
* каждая группа пользователей имеет права на некоторые приложения;
* на каждое приложение могут иметь права несколько групп пользователей.

Создать базу данных (по крайней мере, 4 таблицы), создать формы для ввода данных, заполнить таблицы, создать запрос на обновление, на выборку, на основе запроса на выборку создать отчет, создать отчет по одной из таблиц.

1. **Отдел кадров**

Основные таблицы:

* работники (ФИО, дата рождения);
* штатное расписание (наименование отдела, должность, оклад);

Правила:

* один работник может работать в нескольких отделах, в каждом отделе на разных должностях;
* в каждом отделе несколько работников;

Создать базу данных (по крайней мере, 3 таблицы), создать формы для ввода данных, заполнить таблицы, создать запрос на удаление, на выборку, на основе запроса на выборку создать отчет, создать отчет по одной из таблиц.

1. **Авансовые отчеты**

основные таблицы:

* работники (ФИО);
* авансовые отчеты (дата, номер, сумма, остаток предыдущего аванса);
* статьи расходов (сумма, номер счета по бухгалтерии);

правила:

* у каждого работника может быть несколько отчетов;
* в каждом отчете может быть несколько статей расходов;

Создать базу данных (по крайней мере, 4 таблицы), создать формы для ввода данных, заполнить таблицы, создать запрос на добавление, на выборку, на основе запроса на выборку создать отчет, создать отчет по одной из таблиц.

1. **Склад**

Основные таблицы:

* товар (наименование, цена);
* приход (номер, товар, количество);
* расход (номер, товар, кол-во);

Правила:

* расход может быть из нескольких приходов;
* несколько расходов может быть из одного прихода;

Создать базу данных (по крайней мере, 4 таблицы), создать формы для ввода данных, заполнить таблицы, создать запрос на обновление, на выборку, на основе запроса на выборку создать отчет, создать отчет по одной из таблиц.

# Используемая литература

1. Ю.Бекаревич, Н.Пушкина «Самоучитель Microsoft Access 2002». – Спб. «БХВ-Петербург», 2002.- 720 стр.
2. Специальное издание «Использование Microsoft Office 97» Профессиональный выпуск. – Киев, Москва, Санкт-Петербург: Издат.дом «Вильямс», 1998. – 1120 стр.