Содержание

Введение

1. Описание предметной области

1.1 Описание атрибутов и их свойств

1. Проектирование отношений
   1. Описание связей между таблицами-отношениями
   2. Установление функциональной зависимости между атрибутами, выделение ключевых атрибутов
2. Проектирование базы данных
3. Разработка и создание форм
4. Организация запросов
5. Разработка и создание отчётов

Заключение

Список используемой литературы

# Введение

### Дипломная работа посвящена анализу проектирования баз данных, а также освещению методов построения форм и отчетов на примере построения программы ведения электронной документации учебного заведения. В качестве инструмента построения базы данных использован Microsoft Access. С самого начала эту СУБД отличала простота использования в сочетании с широкими возможностями по разработке законченных приложений.

Актуальность темы.

В настоящее время, несмотря на повышение компьютеризации общества, в сфере образования до сих пор нет средств, позволяющих в достаточной мере автоматизировать процесс ведения документации и отчетности.

Одной из составных задач можно рассматривать проблему составления расписания учебного процесса, а так же оперативную корректировку расписания при возникновении необходимости в этом.

О своевременности и актуальности рассматриваемой проблемы говорит тот факт, что большую часть своего времени администраторы заведений и преподаватели тратят на оформление различной документации и отчетов. Огромное количество учебных заведений и отсутствие предложений в данной сфере гарантируют высокую потребность в данном продукте.

Базы данных (БД) составляют в настоящее время основу компьютерного обеспечения информационных процессов, входящих практически во все сферы человеческой деятельности.

Действительно, процессы обработки информации имеют общую природу и опираются на описание фрагментов реальности, выраженное в виде совокупности взаимосвязанных данных. Базы данных являются эффективным средством представления структур данных и манипулирования ими. Концепция баз данных предполагает использование интегрированных средств хранения информации, позволяющих обеспечить централизованное управление данными и обслуживание ими многих пользователей. При этом БД должна поддерживаться в среде ЭВМ единым программным обеспечением, называемым системой управления базами данных (СУБД). СУБД вместе с прикладными программами называют банком данных.

Одно из основных назначений СУБД – поддержка программными средствами представления, соответствующего реальности.

1. Описание предметной области

Предметной областью называется фрагмент реальности, который описывается или моделируется с помощью БД и ее приложений. В предметной области выделяются информационные объекты – идентифицируемые объекты реального мира, процессы, системы, понятия и т.д., сведения о которых хранятся в БД. В данной курсовой работе разработана база данных "Расписание" для автоматизации и уменьшения времени составления расписания занятий в ВУЗах.

Расписание включает в себя следующие таблицы :

1 Преподаватели

2 Аудитории

3 Группы

4 Занятия

5 Время пары

6 Расписание

Таблица 1.Преподаватели

|  |  |
| --- | --- |
| Преподаватели | |
| Код преподавателя | Ф И О |
| 1 | Акимов А.В.. |
| 2 | Александрова О.А. |
| 3 | Кирилов Н. Я |
| 4 | Павлековская И.М. |
| 5 | Рунова Л.Д. |
| 6 | Новиков И С |

# Таблица 2.Аудитории

|  |  |  |  |
| --- | --- | --- | --- |
| Аудитория | | | |
| Код аудитории | Здание | Этаж | Номер |
| 1 | Л | 3 | 743г |
| 2 | Л | 2 | 211е |
| 3 | Л | 5 | 214е |

# Таблица 3.Группы

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Группы | | | | |
| Код группы | Факультет | Специальность | Курс | Номер |
| 1 | Информатики | Информационные системы | 2 | пи |
| 2 | Юриспреденции и права | Правовед | 3 | ВЛИ-605Г |
| 3 | Информатики | Прикладная информатика (в экономике) | 3 | СЛИ-508У |
| 4 | Экономики | история эк.наук | 4 | ГДР-111Х |
| 5 | Экономики | Маркетинг | 2 | БУГ-123АГА |
| 6 |  |  |  | 1 |

# Таблица 4. Занятия

|  |  |
| --- | --- |
| Занятия | |
| Код занятия | Тип занятия |
| 1 | Базы данных |
| 2 | Программирование |
| 3 | Операционные Системы |
| 4 | Менеджмент |
| 5 | Философия |

# Таблица 5. Время пары

|  |  |  |
| --- | --- | --- |
| Время пары | | |
| №пары | Начало | Конец |
| 1 | 10-00 | 11-30 |
| 2 | 11-40 | 12-20 |
| 3 | 12-30 | 14-00 |
| 4 | 14-10 | 16-40 |
| 5 | 16-50 | 17-30 |
| 6 | 17-40 | 18-30 |

Таблица 6.Расписание

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| Расписание | | | | | | |
| Код занятия | Дата | №пары | Группа | Тип занятия | Преподаватель | Аудитория |
| 4 | 12.02.2007 | 2 | СЛИ-508У | Программирование | Александрова О.А. | 211е |
| 5 | 13.01.2007 | 5 | пи | Менеджмент | Кирилов Н. Я | 743г |
| 6 | 12.02.2007 | 6 | ВЛИ-605Г | Менеджмент | Павлековская И.М. | 214е |
| 7 | 13.01.2007 | 3 | ГДР-111Х | Программирование | Акимов А.В.. | 743г |
| 8 | 13.01.2007 | 1 | пи | Базы данных | Кирилов Н. Я | 211е |
| 9 | 14.01.2007 | 4 | СЛИ-508У | Операционные Системы | Александрова О.А. | 211е |
| 10 | 13.05.2007 | 2 | ВЛИ-605Г | Программирование | Кирилов Н. Я | 421г |

### Описание атрибутов и их свойств

Атрибуты или поля имеют тип данных. При выборе типа данных необходимо учитывать следующее:

* Какие значения должны отображаться в поле.
* Сколько места необходимо для хранения значений в поле.
* Какие операции должны производится со значениями в поле.
* Нужна ли сортировка или индексирование поля.
* Будет ли использоваться группировка в запросах или отчетах.
* Каким образом должны быть отсортированы значения в поле.

Всего имеется 10 типов полей.

Таблица

|  |  |  |  |
| --- | --- | --- | --- |
| Nr | Тип данных | Применение | Размер |
|  | Текстовый | Текст или комбинация текста и чисел | До 255 символов |
|  | МЕМО | Длинный текст. Не сортируется и не индексируется. | До 64000 символов |
|  | Числовой | Числовые данные, использующиеся для вычислений | 1, 2, 4, 8 байт |
|  | Дата/время | Даты и время, обеспечивает правильную сортировку. | 8 байт |
|  | Денежный | Значения валют. Для предотвращения округлений. | 8 байт |
|  | Счетчик | Автоматическая вставка последовательных или случайных чисел при добавлении записи. | 4 байта |
|  | Логический | Два значения типа Да/Нет | 1 бит |
|  | Объект OLE | Объекты, созданные в других программах. | До 1 Гигабайта |
|  | Гиперссылка | Гиперссылки типа UNC или URL. | До 64000 символов |
|  | Мастер подстановок | Позволяет выбрать значения из другой таблицы или из списка значений. | По размеру ключевого поля |

Кроме типа поля имеют форматы, которые позволяют указать форматы вывода текста, чисел, дат и значений времени на экран и на печать. Формат поля для элементов управления задается в окне свойств, а для поля в таблице или запросе в режиме конструктора таблицы (в разделе свойств поля) или в окне запроса (в окне свойств поля). Форматы можно выбирать из списка встроенных форматов для полей, имеющих числовой, денежный, логический типы данных, а также типы данных счетчика и даты/времени. Также для любых типов полей, отличных от объектов OLE есть возможность создания собственных специальных форматов. Кроме того, значение данного свойства можно задать в макросе или в программе Visual Basic.

1. Проектирование отношений

Таблица 7. Связи информационных объектов

|  |  |  |  |
| --- | --- | --- | --- |
| Номер связи | Главный объект | Подчиненный объект | Тип связи |
| 1 | Преподаватели | Расписание | 1:М |
| 2 | Аудитория | Расписание | 1:М |
| 3 | Группы | Расписание | 1:М |
| 4 | Занятия | Расписание | 1:М |
| 5 | Время пары | Расписание | 1:М |

## 

## Описание связей между таблицами-отношениями

После разработки всех таблиц необходимо установить связи между ними.

Связь между таблицами устанавливает отношения между совпадающими значениями в ключевых полях, обычно между полями разных таблиц, имеющих одинаковые имена. В большинстве случаев с ключевым полем одной таблицы, являющимся уникальным идентификатором каждой записи, связывается внешний ключ другой таблицы.

Наиболее часто используемый тип связи "один-к-многим". В такой связи каждой записи в таблице А могут соответствовать несколько записей в таблице В, а запись в таблице В не может иметь более одной соответствующей ей записи в таблице А.

Еще могут быть отношения "многие-к-многим". При таком отношении одной записи в таблице А могут соответствовать несколько записей в таблице В, а одной записи в таблице В – несколько записей в таблице А. Такая схема реализуется только с помощью третьей (связующей) таблицы, ключ которой состоит по крайней мере из двух полей, являющихся полями внешнего ключа в таблицах А и В.

Самый редкий тип связи "один-к-одному". При таком отношении запись в таблице А может иметь не более одной связанной записи в таблице В, и наоборот. Используют такую связь для разделения очень широких таблиц, для отделения части таблицы по соображениям защиты, а также для сохранения сведений, относящихся к подмножеству записей в главной таблице. Например, для хранения конфиденциальных данных.

Для создания схемы следует воспользоваться средством создания схемы данных.

Схема данных представлена на рисунке 1. Таблица Группа, Аудитория, Преподаватели, Занятия, Время пары связанны с таблицей расписание посредством кодового поля.

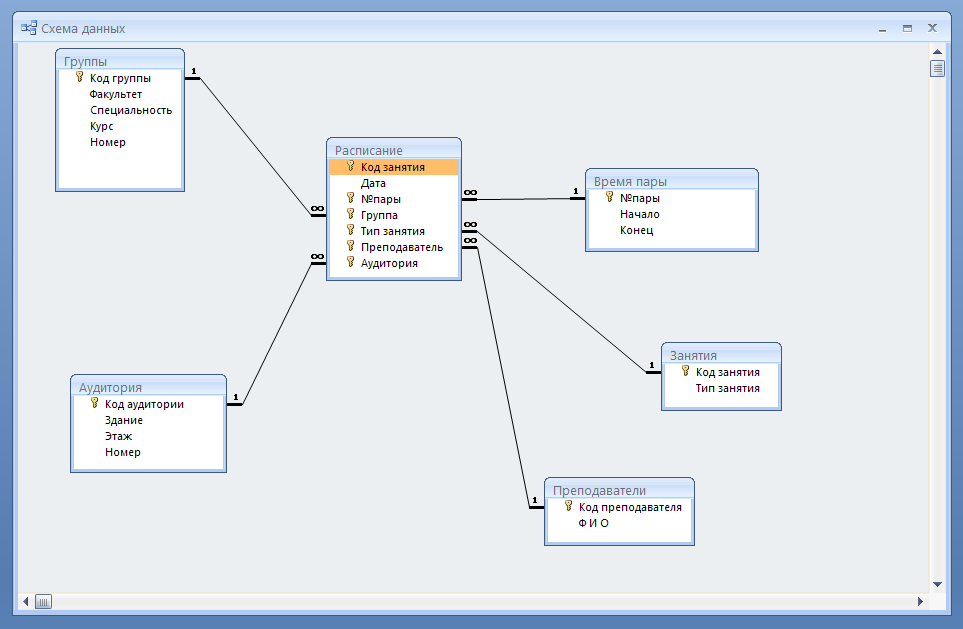


Рис 1.Схема данных "Расписание"

## Установление функциональной зависимости между атрибутами, выделение ключевых атрибутов

Мощь реляционных БД заключается в том, что с их помощью можно быстро найти и связать данные из разных таблиц при помощи запросов, форм и отчетов. Для этого каждая таблица должна содержать одно или несколько полей, однозначно идентифицирующих каждую запись в таблице. Эти поля называются ключевыми.

В Microsoft Access можно выделить три типа ключевых полей: счетчик, простой ключ и составной ключ.

Указание поля счетчика в качестве ключевого – наиболее простой способ создания ключевых полей.

Если поле содержит уникальные значения, такие как коды или инвентарные номера, то это поле можно определить как ключевое.

В случаях, когда невозможно гарантировать уникальность значений каждого отдельного поля, можно создать ключ, состоящий из нескольких полей. Уникальным будет сочетание этих полей.

Индексы – это объекты БД, которые обеспечивают быстрый доступ к отдельным строкам в таблице. Индекс создается с целью повышения производительности операций запросов и сортировки данных таблицы. Индексы также используются для поддержания в таблицах некоторых типов ключевых ограничений.

Индексы обеспечивают наибольшие выгоды для относительно мало изменяющихся таблиц, по которым часто выполняются запросы.

# 3. Проектирование базы данных

В Microsoft Access, прежде чем создавать таблицы, формы и другие объекты необходимо задать структуру базы данных. Хорошая структура базы данных является основой для создания адекватной требованиям, эффективной базы данных.

## Этапы проектирования базы данных

Ниже приведены основные этапы проектирования базы данных:

1. Определение цели создания базы данных.
2. Определение таблиц, которые должна содержать база данных.
3. Определение необходимых в таблице полей.
4. Задание индивидуального значения каждому полю.
5. Определение связей между таблицами.
6. Обновление структуры базы данных.
7. Добавление данных и создание других объектов базы данных.
8. Использование средств анализа в Microsoft Access.
9. Определение цели создания базы данных

На первом этапе проектирования базы данных необходимо определить цель создания базы данных, основные ее функции и информацию, которую она должна содержать. То есть нужно определить основные темы таблиц базы данных и информацию, которую будут содержать поля таблиц.

База данных должна отвечать требованиям тех, кто будет непосредственно с ней работать. Для этого нужно определить темы, которые должна покрывать база данных, отчеты, которые она должна выдавать, проанализировать формы, которые в настоящий момент используются для записи данных, сравнить создаваемую базу данных с хорошо спроектированной, подобной ей базой.

1. Определение таблиц, которые должна содержать база данных

Одним из наиболее сложных этапов в процессе проектирования базы данных является разработка таблиц, так как результаты, которые должна выдавать база данных (отчеты, выходные формы и др.) не всегда дают полное представление о структуре таблицы.

При проектировании таблиц вовсе не обязательно использовать Microsoft Access. Сначала лучше разработать структуру на бумаге. При проектировке таблиц, рекомендуется руководствоваться следующими основными принципами:

* Информация в таблице не должна дублироваться. Не должно быть повторений и между таблицами.

Когда определенная информация храниться только в одной таблице, то и изменять ее придется только в одном месте. Это делает работу более эффективной, а также исключает возможность несовпадения информации в разных таблицах. Например, в одной таблице должны содержаться адреса и телефоны учеников.

* Каждая таблица должна содержать информацию только на одну тему.

Сведения на каждую тему обрабатываются намного легче, если содержаться они в независимых друг от друга таблицах. Например, адреса и заказы учеников хранятся в разных таблицах, с тем, чтобы при удалении заказа информация о ученике осталась в базе данных.

1. Определение необходимых в таблице полей

Каждая таблица содержит информацию на отдельную тему, а каждое поле в таблице содержит отдельные сведения по теме таблицы. Например, в таблице с данными о преподавателях могут содержаться поля с названием специальность, факультет, курс, номер группы. При разработке полей для каждой таблицы необходимо помнить:

* Каждое поле должно быть связано с темой таблицы.
* Не рекомендуется включать в таблицу данные, которые являются результатом выражения.
* В таблице должна присутствовать вся необходимая информация.
* Информацию следует разбивать на наименьшие логические единицы (Например, поля "Имя" и "Фамилия", а не общее поле "Имя").

1. Задание индивидуального значения каждому полю

С тем чтобы Microsoft Access мог связать данные из разных таблиц, например, данные о клиенте и его заказы, каждая таблица должна содержать поле или набор полей, которые определяют уникальность каждой записи в таблице. Такое поле или набор полей называют первичным ключом.

1. Определение связей между таблицами

После распределения данных по таблицам и определения ключевых полей необходимо выбрать схему для связи данных в разных таблицах. Для этого нужно определить связи между таблицами.

Желательно изучить связи между таблицами в уже существующей базе данных.

1. Обновление структуры базы данных

После проектирования таблиц, полей и связей необходимо еще раз просмотреть структуру базы данных и выявить возможные недочеты. Желательно это сделать на данном этапе, пока таблицы не заполнены данными.

Для проверки необходимо создать несколько таблиц, определить связи между ними и ввести несколько записей в каждую таблицу, затем посмотреть, отвечает ли база данных поставленным требованиям. Рекомендуется также создать черновые выходные формы и отчеты и проверить, выдают ли они требуемую информацию. Кроме того, необходимо исключить из таблиц все возможные повторения данных.

1. Добавление данных и создание других объектов базы данных

Если структуры таблиц отвечают поставленным требованиям, то можно вводить все данные. Затем можно создавать любые запросы, формы, отчеты, макросы и модули.

1. Использование средств анализа в Microsoft Access

В Microsoft Access существует два инструмента для усовершенствования структуры баз данных. Мастер анализа таблиц исследует таблицу, в случае необходимости предлагает новую ее структуру и связи, а также переделывает ее.

Анализатор быстродействия исследует всю базу данных, дает рекомендации по ее улучшению, а также осуществляет их.

## Разработка и создание форм

Формы – одно из основных средств для работы с базами данных в Access - используются для ввода новых записей (строк таблиц), просмотра и редактирования уже имеющихся данных, задания параметров запросов и вывода ответов на них и др. Формы представляют собой прямоугольные окна с размещенными в них элементами управления.

Форму можно создать тремя различными способами.

При помощи авто-формы на основе таблицы или запроса. С помощью авто-форм можно создавать формы, в которых выводятся все поля и записи базовой таблицы или запроса. Если выбранный источник записей имеет связанные таблицы или запросы, то в форме также будут присутствовать все поля и записи этих источников записей.

При помощи мастера на основе одной или нескольких таблиц или запросов. Мастер задает подробные вопросы об источниках записей, полях, макете, требуемых форматах и создает форму на основании полученных ответов.

Вручную в режиме конструктора. Сначала создается базовая форма, которая затем изменяется в соответствии с требованиями в режиме конструктора.

Создание форм в режиме Мастера форм.

* В окне базы данных выберите Формы в списке Объекты.
* Нажмите кнопку Создать на панели инструментов окна базы данных.
* В диалоговом окне Новая форма выберите нужного мастера. Описание мастера появляется в левой части диалогового окна.
* Выберите имя таблицы или другого источника записей, содержащего данные, на которых должна быть основана форма. Примечание. Если в списке выбран "Мастер форм" или "Сводная таблица", то этот шаг не обязателен— источник записей для формы можно указать позднее.
* Нажмите кнопку OK.
* Следуйте инструкциям мастера.

Созданную форму можно изменить в режиме конструктора, сводной таблицы или сводной диаграммы.

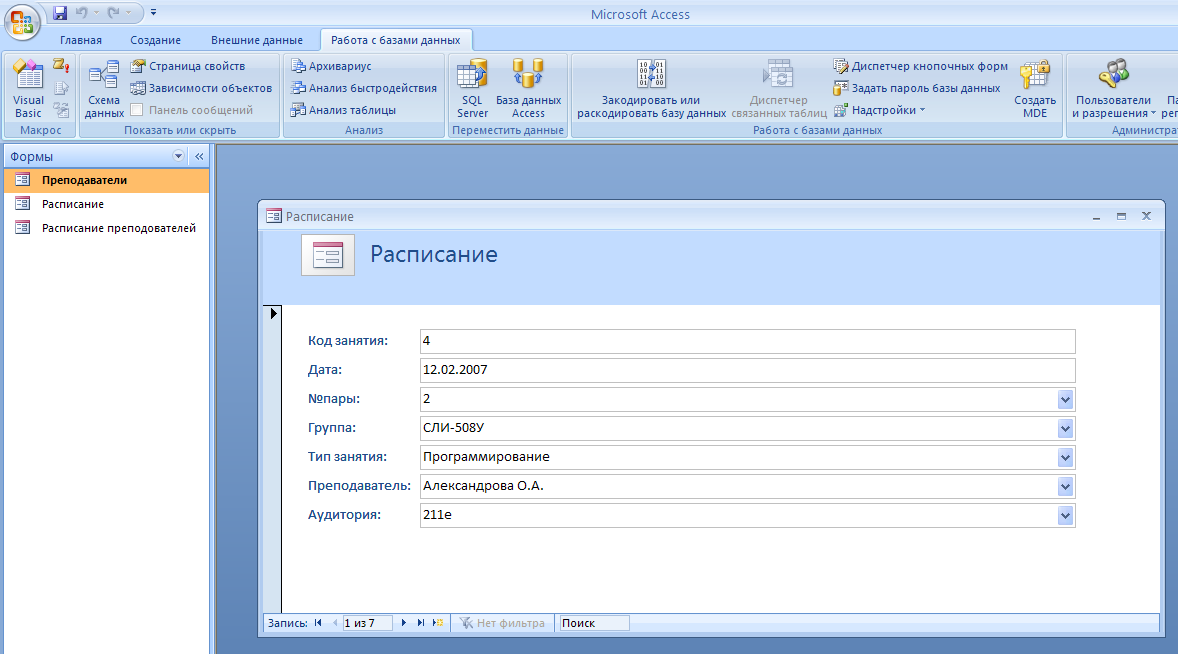


Рис 2 Форма расписание

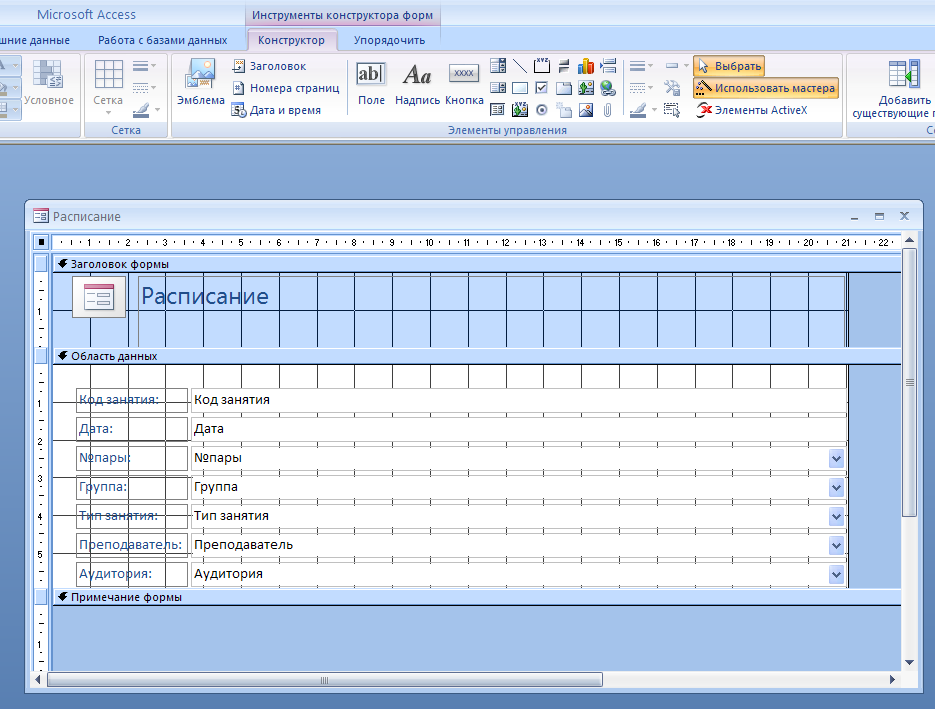


Рис 3 Конструктор форм

## Организация запросов

Запросы в Microsoft Access можно создать либо автоматически, с помощью мастеров, либо самостоятельно, в окне конструктора запросов. Доработать и оптимизировать инструкцию запроса можно в режиме SQL.

Мастера запросов автоматически выполняют основные действия в зависимости от ответов пользователя на поставленные вопросы. Мастер простого запроса на выборку используется для создания запросов для получения данных из полей, определенных в одной или нескольких таблицах или запросах. С помощью мастера можно также вычислять суммы, число записей и средние значения для всех записей или определенных групп записей, а также находить максимальное и минимальное значение в поле. Однако нельзя ограничить количество записей, возвращаемых этим запросом, с помощью условий отбора.

* В окне базы данных в списке Объекты выберите Запросы и на панели инструментов окна базы данных нажмите кнопку Создать.
* В диалоговом окне Новый запрос выберите в списке строку Простой запрос и нажмите кнопку OK.
* Следуйте инструкциям в диалоговых окнах мастера. Последнее диалоговое окно позволяет либо запустить запрос, либо открыть его в режиме конструктора.
* Если получился не тот запрос, который был нужен, можно снова создать запрос с помощью мастера или изменить этот запрос в режиме конструктора.

Работа с запросами в режиме конструктора.

* В окне базы данных в списке Объекты выберите Запросы и на панели инструментов окна базы данных нажмите кнопку Создать.
* В диалоговом окне Новый запрос щелкните строку Конструктор, а затем нажмите кнопку OK.
* В диалоговом окне Добавление таблицы выберите вкладку, содержащую объекты, данные из которых будут использованы в запросе.
* Дважды щелкните объекты, которые нужно добавить в запрос, а затем нажмите кнопку Закрыть.
* Добавьте поля в строку Поле в бланке запроса и, если необходимо, укажите условия и порядок сортировки.
* Для просмотра результатов запроса на панели инструментов нажмите кнопку Вид.

Запросы могут нам в основном понадобиться для создания отчетов. Или для распечатки выборочных данных. Для этой базы данных в запросах есть необходимость. Создадим запрос по дате.

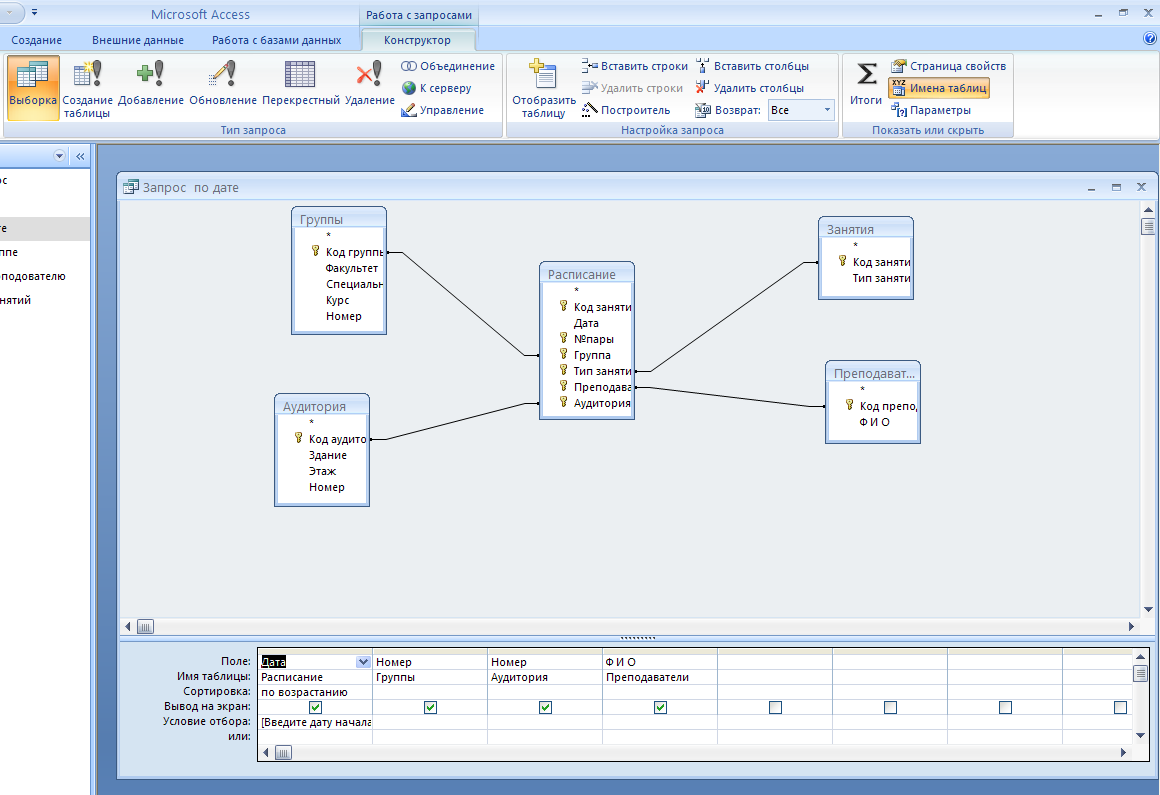


Рис4. Запрос по дате

## 6. Разработка и создание отчетов

Отчет является эффективным средством представления данных в печатном формате. Имея возможность управлять размером и внешним видом всех элементов отчета, пользователь может отобразить сведения желаемым образом.

Отчеты во многом схожи с формами. При проектировании отчетов используются многие технологии, применяемые для форм.

Большинство отчетов являются присоединенными к одной или нескольким таблицам и запросам из базы данных. Источником записей отчета являются поля в базовых таблицах и запросах. Отчет не должен включать все поля из каждой таблицы или запроса, на основе которых он создается.

Присоединенный отчет получает данные из базового источника записей. Другие данные такие как, заголовок, дата и номера страниц, сохраняются в макете отчета.

Связь между отчетом и его источником данных создается при помощи графических объектов, называемых элементами управления. Элементами управления являются поля, в которых отображаются имена и числа, надписи, в которых отображаются заголовки, а также декоративные линии, графически структурирующие данные и улучшающие внешний вид отчета.

Мастера помогают быстро создать отчеты различных типов. Мастер наклеек используется для создания почтовых наклеек, мастер диаграмм помогает создать диаграмму, а мастер отчетов создавать стандартные отчеты. Мастер выводит на экран вопросы и создает отчет на основании ответов пользователя. После этого пользователь имеет возможность доработать и изменить отчет в режиме конструктора.

Можно настроить отчет следующими способами.

Источник записей. Измените таблицу или запрос, на котором основан отчет.

Сортировка и группировка данных. Можно сортировать данные по возрастанию и по убыванию. Также можно группировать записи по одному или нескольким полям и отображать промежуточные и общие итоги в отчете.

Окно отчета. Можно добавить или удалить кнопки Развернуть и Свернуть, изменить текст заголовка и другие элементы окна отчета.

Разделы. Можно добавить, удалить, скрыть или изменить размер подписи, примечания и области данных отчета. Можно также задавать свойства разделов, определяющие вид отчета на экране и при печати.

Элементы управления. Пользователь может перемещать элементы управления, изменять их размеры и задавать свойства шрифта для элементов. В отчет можно также добавлять элементы управления для отображения вычисляемых значений, итогов, текущей даты и времени и других полезных сведений.

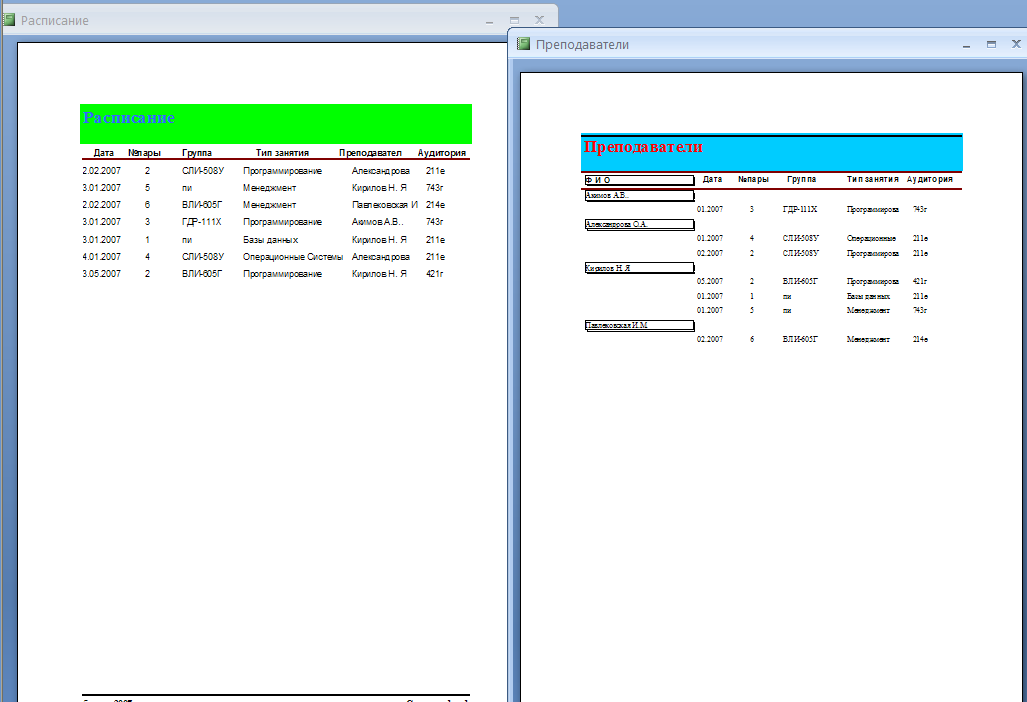


Рис 5. Отчет по Расписанию.

Заключение

Рассмотрев свою курсовую работу на практическом примере, я понимаю, что Microsoft Access обладает всеми чертами классической СУБД и предоставляет дополнительные возможности. Access - это не только мощная, гибкая и простая в использовании СУБД, но и система для разработки работающих с базами данных приложений. С помощью Access можно создать приложение, работающее в среде Windows и полностью соответствующее нашим потребностям по управлению данными. Используя запросы, можно выбирать и обрабатывать хранящуюся в таблицах информацию. Можно создавать формы для ввода, просмотра и обновления данных, а также использовать Access для создания как простых, так и сложных отчетов. Формы и отчеты "наследуют" свойства базовой таблицы или запроса, так что в большинстве случаев вы указываете форматы, условия на значения и некоторые другие характеристики данных только один раз. К числу наиболее мощных средств Access относятся средства разработки объектов - Мастера, которые можно использовать для создания таблиц, запросов различных типов форм и отчетов, просто выбрав с помощью мыши нужные опции. Чтобы полностью автоматизировать работу вашего приложения, с помощью макросов Access можно легко связать данные с формами и отчетами. Вы можете создать большинство приложений, не написав ни единой строки программы, но если необходимо создать нечто уж совсем изощренное, то на этот случай Microsoft Access предоставляет мощный язык программирования - Microsoft Access Basic.

Я рад, что нашёл время разобраться и выполнить работу самостоятельно, так как считаю, что в моей профессии такие навыки, пусть даже базовые, мне со временем очень пригодятся.

Список используемой литературы

1. ACCESS 2000: Разработка приложений. И. Харитонова, В. Михеева. BHV, 2000.
2. Access97. В. Корнелюк, З. Веккер, Н. Зиновьев. "СОЛОН", 1998.
3. Крёнке Д.Теория и практика построения баз данных, 8-е изд. СПб: "Питер", 2003.
4. Д., Цыганков В. М., Мальцев М. Г. Базы данных: Учебник для высших учебных
5. заведений /Под ред. проф. А. Д. Хомоненко. Спб.: КОРОНА принт, 2000.
6. Справочная система Microsoft Access.