|  |
| --- |
| ГОУ ВПО «Московский государственный открытый университет»Чебоксарский политехнический институт (филиал) |
| Кафедра Управления и информатики в технических системах |
| **КУРСОВАЯ РАБОТА** |
| **по дисциплине: «Технологии программирования»** |
| **на тему: «Создание базы данных туристической фирмы»** |
|  | **Выполнил**:студент 3 курса очной формы обученияшифр специальности 220201учебный шифр 607015**Проверил:** |
| Чебоксары - 2009 |

**СОДЕРЖАНИЕ**

1 Введение……………………………………………………………………...3

2 Пояснительная записка……………………………………………………...5

2.1 Краткая характеристика предметной области…………………………5

2.2 Создание таблиц………………………………………………………....5

2.2.1 Разработка структуры БД……………………………………………..5

2.2.2 Инфологическое проектирование…………………………………....7

2.2.3 Структура и создание таблиц………………………………………...8

2.2.4 Реляционная схема базы данных…………………………………...11

2.2.5 Заполнение базы данных……………………………………………13

2.3 Оперирование данными………………………………………………..15

2.3.1 Создание запросов…………………………………………………...15

2.4 Выходные данные……………………………………………………...16

2.4.1 Отчет о поставщиках и поставляемых ими товарах……………16

2.4.2 Создание кнопочной формы……………………………………...21

3 Заключение…………………………………………………………………23

4 Список литературы…………….………………………………….….……25

1. **ВВЕДЕНИЕ**

В настоящее время среди разработчиков базы данных (БД) большой популярностью пользуется реляционная СУБД ACCESS, входящая в состав пакета Microsoft Office 2003. Дружественный интерфейс и простота настройки, эффективные средства создания таблиц, форм, запросов, интеграция с другими приложениями пакета, средства организации работы с базами данных и защита информации - вот далеко не полный перечень достоинств этого приложения.

Основные функции СУБД – это описание структуры базы данных, обработка данных и управление данными.

База данных – это совокупность сведений о реальных объектах, процессах, событиях или явлениях, относящихся к определённой теме или задаче, организованная таким образом, чтобы обеспечить удобное представление этой совокупности, как в целом, так и любой её части. Реляционная база данных представляет собой множество взаимосвязанных таблиц, каждая из которых содержит информацию об объектах определённого типа. Каждая строка таблицы содержит данные об одном объекте (например, клиенте, автомобиле, документе), а столбцы таблицы содержат различные характеристики этих объектов – атрибуты (например, наименования и адреса клиентов, марки и цены автомобилей). Строки таблицы называются записями, все записи имеют одинаковую структуру – они состоят из полей, в которых хранятся атрибуты объекта. Каждое поле в записи содержит одну характеристику объекта и имеет строго определённый тип данных (например, текстовая строка, число, дата). Все записи имеют одни и те же поля, только в них содержатся разные значения атрибутов.

Любая СУБД позволяет выполнять четыре простейшие операции с данными:

- добавить в таблицу одну или несколько записей;

* удалить из таблицы одну или несколько записей;
* обновить значения некоторых полей в одной или нескольких записях;
* найти одну или несколько записей, удовлетворяющих заданному условию.

 Для выполнения этих операций используется механизм запросов. Результатом выполнения запросов является либо отобранное по определённым критериям множество записей, либо изменение в таблицах.

1. **ПОЯСНИТЕЛЬНАЯ ЗАПИСКА**
	1. **Краткая характеристика предметной области**

В данном курсовом проекте в качестве предметной области рассматривается туристическая фирма. Наша база данных решает следующие задачи: учёт заказов, представление данных о доступных турах, сотрудниках, клиентах и о поставщиках услуг.

В режиме форм создано кнопочное меню. С его помощью можно быстро получить доступ к данным о сотрудниках, клиентах, заказах, турах, поставщиках услуг, стоимости путёвок и предоставляемых услугах. Осуществлены запросы по поиску сведений о доступных турах, клиентах, сотрудниках, поставщиках услуг, заказах, услугах. Также производится процедура наценки стоимости на путёвки на 20 % и увеличения заработной платы сотрудникам на 5 %.

Применяемая СУБД: ACCESS 2003*.*

Изложим данные о туристической фирме. У фирмы есть поставщики (сотрудничающие фирмы), осуществляющие предоставление определённых услуг (перевозка пассажиров, обеспечение их временным жильём, помощь с оформлением виз и т.д.)

 Также в фирме есть свой штат сотрудников: менеджеры и консультанты, осуществляющие продажу путёвок, бухгалтер, охранник и уборщик.

**2.2 Создание таблиц.**

**2.2.1 Разработка структуры БД**

Удачная разработка базы данных обеспечивает простоту ее поддержания. Данные следует сохранять в таблицах, причем каждая таблица должна содержать информацию одного типа, например, сведения о поставщиках. Тогда достаточно будет обновить конкретные данные, такие как адрес, только в одном месте, чтобы обновленная информация отображалась во всей базе 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 лучше разработать структуру на бумаге и только затем начинать работу с СУБД Access. При проектировке таблиц, рекомендуется руководствоваться следующими основными принципами:

- 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 фамилии клиентов.

- 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 клиентов хранятся в разных таблицах, с тем, чтобы при удалении заказа информация о клиенте осталась в базе данных.

Каждая таблица содержит информацию на отдельную тему, а каждое поле в таблице содержит отдельные сведения по теме таблицы. Например, в таблице с данными о поставщиках могут содержаться поля с названием компании, адресом и номером телефона. 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»).

**2.2.2 Инфологическое проектирование**

Первым этапом и самым главным этапом в процессе проектирования и создания базы данных, является разработка инфологической модели.

Цель инфологического моделирования – обеспечение наиболее естественных для человека способов сбора и представления той информации, которую предполагается хранить в создаваемой базе данных. Основными конструктивными элементами инфологических моделей являются сущности, связи между ними и их свойства (атрибуты).

Ключ – минимальный набор атрибутов, по значениям которых можно однозначно найти требуемый экземпляр сущности. Минимальность означает, что исключение из набора любого атрибута не позволяет идентифицировать сущность по оставшимся.

Связь – ассоциирование двух или более сущностей. Если бы назначением базы данных было только хранение отдельных, не связанных между собой данных, то ее структура могла бы быть очень простой. Однако одно из основных требований к организации базы данных – это обеспечение возможности отыскания одних сущностей по значениям других, для чего необходимо установить между ними определенные связи.

Между двумя сущностям, например, А и В возможны четыре вида связей.

**Первый тип** – связь ОДИН-К-ОДНОМУ (1:1): в каждый момент времени каждому представителю (экземпляру) сущности А соответствует 1 или 0 представителей сущности В:

Студент может не "заработать" стипендию, получить обычную или одну из повышенных стипендий.

**Второй тип** – связь ОДИН-КО-МНОГИМ (1:М): одному представителю сущности А соответствуют 0, 1 или несколько представителей сущности В.

Квартира может пустовать, в ней может жить один или несколько жильцов.

Так как между двумя сущностями возможны связи в обоих направлениях, то существует ещё два типа связи МНОГИЕ-К-ОДНОМУ (М: 1) и МНОГИЕ-КО-МНОГИМ (М: М). Но мы их использовать не будем.

* + 1. **2.2.3 Структура и создание таблиц**

В таблицах данные распределяются по столбцам (которые называют полями) и строкам (которые называют записями). Все данные, содержащиеся в поле таблицы, должны иметь один и тот же тип. Каждое поле таблицы характеризуется наименованием, типом и шириной поля. При задании типа дан­ных поля можно также указать размер, формат и другие параметры, влияющие на отображение значения поля и точность числовых данных. Основные типы данных:

* *Текстовый.* Текст или числа нетребующие проведения расчётов.
* *МЕМО.* Поле этого типа предназначено для хранения небольших текстовых данных (до 64000 символов). Поле этого типа не может быть ключевым или проиндексированным.
* *Числовой.* Этот тип данных содержит множество подтипов. От выбора подтипа (размера) зависит точность вычислений.
* *Счётчик.* Уникальные, последовательно возрастающие числа, автоматически вводящиеся при добавлении новой записи в таблицу.
* *Логический.* Логические значения, а так же поля, которые могут содержать одно из двух возможных значений.
* *Денежный.* Денежные значения и числовые данные, используемые в математических вычислениях.
* *Дата/Время.* Дата и время хранятся в специальном фиксированном формате.
* *Поле объекта OLE.*  Включает звукозапись, рисунок и прочие типы данных. Поле этого типа не может быть ключевым или проиндексированным.
* *Гиперсвязь.* Содержит адреса Web-страниц.

|  |
| --- |
| **Таблица: Клиенты** |
| Поле  | Тип данных |
| Код клиента | Числовой |
| Фамилия | Текстовый |
| Имя | Текстовый |
| Отчество | Текстовый |
| Адрес | Текстовый |
| Телефон | Текстовый |

 Определим, какие типы данных будут присутствовать в наших таблицах:

|  |
| --- |
| **Таблица: Сотрудники** |
| Поле | Тип данных |
| Код сотрудника | Числовой |
| Фамилия | Текстовый |
| Имя | Текстовый |
| Отчество | Текстовый |
| Дата рождения | Дата/время |
| Зарплата | Текстовый |
| Должность | Текстовый |
| Адрес | Текстовый |
| Телефон | Текстовый |

|  |
| --- |
|  **Таблица: Туры** |
| Поле | Тип данных |
| Код тура | Числовой |
| Код услуги | Числовой |
| Страна | Текстовый |
| Курорт | Текстовый |
| Транспорт | Текстовый |
| Визовое обслуживание | Логический |
| Проживание | Логический |
| Тип проживания | Текстовый |
| Питание | Логический |
| Тип питания | Текстовый |
| Экскурсии | Логический |
| Дата отправления | Дата/время |
| Дата прибытия | Дата/время |
| Стоимость тура | Денежный |

|  |
| --- |
| Таблица: Поставщик |
| Поле | Тип данных |
| Код поставщика | Числовой |
| Название поставщика | Текстовый |
| Представитель поставщика | Текстовый |
| Обращаться | Текстовый |
| Телефон | Текстовый |
| Адрес | Текстовый |

|  |
| --- |
| **Таблица: Услуги** |
| Поле | Тип данных |
| Код услуги | Числовой |
| Код поставщика | Числовой |
| Дата исполнения | Дата/время |

|  |
| --- |
| **Таблица: Заказы** |
| Поле | Тип данных |
| Код заказа | Числовой |
| Код тура | Числовой |
| Код сотрудника | Числовой |
| Код клиента | Числовой |
| Дата оформления | Дата/время |

Для создания базы данных нужно сначала запустить программу Microsoft Office Access 2003, для этого нужно проделать следующие действия: Пуск → Программы → Microsoft Office → Microsoft Office Access 2003.

После этого откроется окно программы. Для создания новой базы данных нужно выполняем:

- команду Файл → Создать

- в открывшемся окне диалога «Создание» выбираем «Новая база данных». На экране появится окно с запросом директории для новой базы данных, вводим имя базы Курсовая, затем «ОК». После этого появится окно базы данных (Рисунок 1).

Рисунок 1. Окно базы данных.

Создание таблиц, как и других объектов баз данных, начинается с активации окна базы данных. В этом окне выбирается элемент управления Таблицы, после чего на правой панели окна будет представлен список таблиц, уже входящих в состав БД. Если в базе нет таблиц, то на правую панель окна выводится только три элемента управления для создания новой таблицы: Создание таблицы в режиме конструктора, Создание таблицы с помощью мастера и Создание таблицы путём ввода данных.

Выбираем элемент управления Создание таблицы в режиме конструктора. Теперь необходимо заполнить Имена полей и выбрать Типы данных. По окончании заполнения через меню Файл→Сохранить как сохраняем полученную таблицу с требуемым именем. На рисунке 2 можно увидеть таблицы для БД туристической фирмы в режиме конструктора.

 Рисунок 2. Таблицы базы данных туристической фирмы в режиме конструктора.

* + 1. **Реляционная схема базы данных**

Реляционная база данных – это совокупность отношений, содержащих всю информацию, которая должна храниться в БД. Однако пользователи могут воспринимать такую базу данных как совокупность таблиц.

1. Каждая таблица состоит из однотипных строк и имеет уникальное имя.

2. Строки имеют фиксированное число полей (столбцов) и значений (множественные поля и повторяющиеся группы недопустимы). Иначе говоря, в каждой позиции таблицы на пересечении строки и столбца всегда имеется в точности одно значение или ничего.

3. Строки таблицы обязательно отличаются друг от друга хотя бы единственным значением, что позволяет однозначно идентифицировать любую строку такой таблицы.

4. Столбцам таблицы однозначно присваиваются имена, и в каждом из них размещаются однородные значения данных (даты, фамилии, целые числа или денежные суммы).

5. Полное информационное содержание базы данных представляется в виде явных значений данных, и такой метод представления является единственным. В частности, не существует каких-либо специальных "связей" или указателей, соединяющих одну таблицу с другой.

6. При выполнении операций с таблицей ее строки и столбцы можно обрабатывать в любом порядке безотносительно к их информационному содержанию. Этому способствует наличие имен таблиц и их столбцов, а также возможность выделения любой их строки или любого набора строк с указанными признаками.

Разработаем схему данных, (создание связей между таблицами). Для этого:

* нажимаем по кнопку на панели инструментов (или команда **Сервис**, **Схема данных).** На экране появится окно <<Схема данных>>;

* щёлкаем по кнопке на панели инструментов (или команда **Связи, Добавить таблицу);**

* в появившемся окне будет выделено название одной таблицы. Щелкаем по кнопке <Добавить>, переводим выделение на имя следующей таблицы и щелкните по кнопке <Добавить>. Аналогично добавляем оставшиеся таблицы;
* закройте окно, щелкнув по кнопке <3акрыть>;
* чтобы не выполнять все вышеописанные действия, можно просто перетащить мышкой таблицы из окна «Базы данных Таблицы» в окно «Схема данных»**;**
* создадим связь между таблицами *Сотрудники* и *Заказы.* Для этого курсором мыши перетаскиваем <<Код сотрудника>> в таблице *Сотрудники* на поле <<Код сотрудника>> в таблицу *Заказы.* На экране откроется окно <<Связи>>;
* устанавливаем флажок («галочку») в свойствах *Обеспечение целостности данных,* *Каскадное обновление связанных полей* и *Каскадное удаление связанных записей;*
* щелкаем по кнопке <Создать>. Связь будет создана;
* аналогично создаем связи между полем <<Код клиента>> в таблице *Клиенты* и полем <<Код клиента>> в таблице *Заказы,* и т.д. Рисунок 3 отображает полученную схему базы данных компьютерного магазина.
* закрываем окно схемы данных, ответив *ДА* на вопрос о сохранении макета.

 Рисунок 3. Реляционная структура СУБД Access для информационной системы туристической фирмы.

* + 1. **Заполнение базы данных.**

В созданной базе данных существует возможность правки, добавления, просмотра всех данных. Причем за счет связей при процессе правки одной из таблиц, будут автоматически изменяться данные в других таблицах.

Формы служат для ввода и просмотра данных в удобном для пользователя виде, который соответствует привычному для него документу. При выводе данных с помощью форм можно применять специальные средства оформления.

Для создания формы нужно открыть вкладку «Формы» окна базы данных и нажать кнопку «Создать»

После этого в появившемся окне следует выбрать способ создания формы «Мастер форм» и выбрать таблицу, по которой необходимо сделать форму, в нашем случае таблица «Сотрудники».

Далее при помощи кнопок (перенести все записи) или (перенести выбранную запись) нужно выбрать поля, которые будут отражены в форме. Нажимаем кнопку «Далее». Затем нужно выбрать внешний вид формы, стиль (рисунок 7), задать имя формы и форма готова. Если автоматическое расположение полей формы нас не устраивает, то необходимо его изменить. Перейдем в режим конструктора (кнопка на панели инструментов ). Здесь можно менять расположение и размер полей формы по своему усмотрению: достаточно мышкой выделить интересующий объект.

 Рисунок 4 Готовая форма таблицы Сотрудники.

Предположим, что вам необходимо найти в базе данных определённого сотрудника (например, по фамилии). Для этого нужно:

- установить курсор в поле «Фамилия».

- выполнить команду меню Правка → Найти, или воспользоваться кнопкой панели инструментов.

Или необходимо найти всех сотрудников – менеджеров. Для этого:

- выбираем необходимое поле (в данном случае ставим курсор в поле «Должность» → «менеджер») и нажимаем кнопку (или выполняем команду **Записи, Фильтр, Фильтр по выделенному)**. В форме (или таблице) останутся только записи о сотрудниках, имеющих должность менеджер;

- для отмены фильтрации нажимаем на кнопку на панели инструментов или выполняем команду **Записи, Удалить фильтр**. В форме (таблице) появятся все данные. Как правило, фильтрацию удобнее использовать в табличных формах или непосредственно в самих таблицах СУБД Access.

* 1. **Оперирование данными**

**2.3.1 Создание запросов**

Запросы позволяют выбирать данные из одной или нескольких связанных таблиц. Результатом выполнения запроса является результирующая таблица, которая наряду с другими таблицами может быть использована при обработке данных. С помощью запросов можно также обновлять, удалять или добавлять данные в таблицы. Основным назначением запросов является отбор данных по критериям поиска.

Создадим запросы на наличие путёвки в определённую страну.

Для создания запроса необходимо открыть вкладку «Запросы» окна базы данных, нажать кнопку создать и в появившемся окне выбрать «Простой запрос». В этом случае будет предложено указать список таблиц и их полей. Выбираем таблицу «Туры», добавляем необходимые поля (Рисунок 5), нажимаем кнопку «Далее». На основании этих данных будет создан запрос. В следующем окне нажимаем «Далее». Теперь задаем имя запроса «Доступные туры» и отмечаем галочкой пункт «Изменить макет запроса», нажимаем «Готово». Запрос откроется в режиме конструктора. Теперь в столбце «Страна» в строке «Условие отбора» задаем [Введите страну]. При выполнении запроса (значок на панели управления) появится окно с предложением ввести название страны. (Рисунок 6)

Рисунок 5. Создание запроса «Доступные туры».

Рисунок 6. Окно с предложением ввести страну.

Для удобства использования запросов, для них создаются формы.

* 1. **Выходные данные**
		1. **Отчеты**

Отчёты предназначены для формирования выходных документов и вывода их на печать. По своим свойствам и структуре отчёты во многом подобны формам. Основное их отличие заключается в том, что в отчёте отображаются все данные и в них предусмотрена возможность группировать данные по различным критериям. Отчёты в отличие от форм могут содержать специальные элементы оформления, характерные для печати документов: колонтитулы, номера страниц и т.д.

Отчёты, так же как и формы, можно создавать с помощью конструктора или мастера отчётов. Используется также автоматическое создание отчётов.

Последовательность действий при формировании отчёта с помощью Мастера такая же, как и при создании формы. Дополнительными являются два шага: добавление уровня группировок и задание требуемого порядка сортировки. Добавление уровня группировки позволяет выводить записи, объединённые по выбранным полям. Например, при задании уровня группировки по полю Должность записи будут сгруппированы по конкретному значению этого поля. Access автоматически сортирует данные по группирующим полям, а внутри группы можно также задать сортировку по любому из полей, входящих в группу.

По каждой из групп при необходимости можно подводить итоги. Для получения итоговых значений по числовым полям необходимо после добавления уровня группировки в диалоговом окне Создание отчётов щелкнуть на кнопке Итоги, после чего для соответствующего числового поля выбрать требуемые функции. При выборе функции Sum (суммы) для какого-либо поля Access не только автоматически подсчитает сумму значений для каждой группы, но и подведёт итоги по всем записям выбранного поля. Программа предусматривает вывод как данных (записей) и итогов, так и только итогов. Если не задан уровень группировки, то кнопка «Итоги» становится недоступной.

Из режима просмотра пользователь может скопировать отчёт в виде отдельного файла текстового редактора Word или электронной таблицы Excel. Для этого необходимо щёлкнуть на кнопке «Связи с Office», расположенной на панели инструментов «Предварительный просмотр».

Для создания отчета следует нужно открыть вкладку «Отчеты» в окне базы данных и нажать кнопку «Создать» в верхней части окна базы данных.

В появившемся окне «Новый отчет» выбрать пункт «Мастер отчетов» и указать источник данных – таблицу или запрос (в нашем случае это запрос на данные о поставщиках услуг), нажать кнопку «ОК». Выбрать поля для отчета из имеющихся источников как мы это делали при создании форм и запросов. Выбираем вид представления данных, нажимаем «Далее». Добавляем уровни группировки, нажимаем «Далее». Выберите макет отчета. Рекомендуем ступенчатый, так как он занимает меньше места и в нем наглядно представлены данные (хотя это дело вкуса). Щелкните по кнопке «Далее». Выберите стиль отчета и щелкните по кнопке «Далее». Введите название отчета Итоги сессии и щелкните по кнопке «Готово». На экране появится отчет. Его можно просмотреть, изменяя масштаб (щелкнув по листу) и перелистывая страницы (в нижней части экрана). Его можно также распечатать, выполнив команду **Файл,** Печать. После завершения необходимых вам операций закройте окно просмотра отчета.

**2.4.1 Создание кнопочной формы**

Для создания кнопочного меню выполните следующие действия:

* выполните команду Сервис, Служебные программы, Диспетчер кнопочных форм;
* подтвердите создание кнопочной формы, щелкнув по кнопке «Да»;
* Access предложит вам работать с главной кнопочной формой или создать дополнительно новую. Создайте свою форму, щелкнув по кнопке «Создать»;
* введите имя *Меню* и щелкните по кнопке «ОК»;
* в окне выберите *Меню* и щелкните по кнопке «Изменить»;
* создайте элементы данной кнопочной формы, щелкнув по кнопке «Создать»;
* в строке *Текст* введите поясняющую надпись к первой создаваемой кнопке — Сотрудники;
* в строке Команда выберите из списка Открытие формы в режиме редактирования; **Примечание.** Диспетчер напрямую может связать кнопку с открытием формы или отчета. Чтобы открыть таблицу или запрос, надо создать соответствующий макрос и указать это в диспетчере.
* в строке Форма выберите из списка форму Сотрудники и щелкните по кнопке «ОК»;
* введите в меню все созданные формы и отчеты по данному образцу;
* закройте окно кнопочной формы, щелкнув по кнопке «3акрыть»;
* щелкните по кнопке «По умолчанию»;
* закройте диспетчер кнопочных форм, щелкнув по кнопке «3акрыть»;

Моё меню.

**3. ЗАКЛЮЧЕНИЕ**

База данных – это совокупность структурированных и взаимосвязанных данных и методов, обеспечивающих добавление выборку и отображение данных. Microsoft Access позволяет управлять всеми сведениями из одного файла базы данных. В рамках этого файла используются следующие объекты:

* таблицы для сохранения данных;
* запросы для поиска и извлечения только требуемых данных;
* формы для просмотра, добавления и изменения данных в таблицах;
* отчеты для анализа и печати данных в определенном формате;

 Удачная разработка базы данных обеспечивает простоту ее поддержания.

Разработанная в данном курсовом проекте база данных для решения задач документооборота в туристической фирме, позволяет автоматизировать весь документооборот, что позволяет разгрузить отчетный документооборот, и сократить ошибки в документации. При появлении новых производственных задач разработчик может в кратчайшие сроки реализовать их в базе данных, путем добавления строк, столбцов и целых таблиц.

Описанная в этом курсовом проекте база данных должна быть полезна при работе в данной области.

В ходе проделанной работы, нами были созданы:

* 6 таблиц.
* 7 запросов.
* 8 форм.
* 5 отчетов.

Microsoft Access, обладая всеми чертами классической СУБД, предоставляет и 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 потребностям по управлению данными. Используя запросы, есть возможность 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 необходимо указать 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.

 С помощью Access также легко импортировать данные из текстовых файлов, документов текстовых процессоров и электронных таблиц (и экспортировать данные в них).

Microsoft Access имеет множество областей применения: на малом предприятии; корпорации и даже в сфере домашнего применения.

**4. СПИСОК ЛИТЕРАТУРЫ**

|  |  |
| --- | --- |
|  | Бакаревич Ю.Б., Пушкина Н.В. Самоучитель Microsoft Access 2002. – СПб.: БХВ-Петербург, 2002 |
|  | Бондарева Г.А., Сахарова Е.В., Королькова Л.Н., Информатика. Методические указания. Ставрополь, СТИС, 2006 |
|  | Бойко В.В., Савинков В.М. Проектирование баз данных информационных систем. – М.: Финансы и статистика, 1989. |
|  | Гончаров А. Ю. Access 2003. Самоучитель с примерами., Москва, 2004г. |
|  | Григорьев В.А., Ревунков В.И. Банки данных. Учебник для вузов. – М., МВТУ им.Баумана, 2002. |
|  | Домашняя страница Access: справка и обучение http://office.microsoft.com/ru-ru/access/FX100487571049.aspx |
|  | Мейер М. Теория реляционных баз данных. – М.: Мир, 1987. |
|  | Основы проектирования реляционных баз данных. Электронное учебное пособие. |
|  | Симонович С. В., Евсеев Г. А., Алексеев А. Г. Специальная информатика: Учебное пособие. – М.: АСТ-ПРЕСС: Инфорком-Пресс, 2000 |
|  | Хаббард Дж. Автоматизированное проектирование баз данных. – М.: Мир, 1984. |