Министерство образования и науки Челябинской области

ГОУ СПО Челябинский государственный промышленно-гуманитарный техникум им. А.В. Яковлева

КОНТРОЛЬНАЯ РАБОТА

по дисциплине

Информатика

Выполнил: студент

заочного отделения, гр. 309з

Чернышев А.Ю.

Проверил: преподаватель

Вардугин А.Г.

Челябинск, 2010г

**Содержание**

1. Формы представления информации

2. Табличный процессор Excel: типы данных

3. СУБД Access: запрос с параметром (принцип работы, этапы создания)

Список используемой литературы

**1. Формы представления информации**

*Информация* по своей сути включает в себя весь окружающий мир и все то, что его наполняет: знания, навыки, предметы, чертежи, культура, искусство, речь, взгляд и т.д. Информация может быть *непрерывной* и *дискретной*.

*Непрерывными величинами* называют величины, характеризующиеся физическими величинами (длина, ширина, вес, объем, расстояние, плотность, масса, температура и т.д.).

Дискретными величинами называют величины, характеризующиеся числом, количеством предметов или информации (10 Мб, 3книги, 15 шагов и т.д.).

Информация в персональном компьютере (ПК) представляется в двоичном формате (системе счисления) в виде единиц и нулей. Минимальной единицей информации является бит. Группа из восьми бит представляет собой байт. Каждый байт в памяти компьютера имеет уникальный адрес, начиная с нуля. Таким образом, байт является минимальной адресуемой единицей памяти.

При определении информационной емкости современных ПК байт как единица информации используется редко. Чаше используются его производные - Кбайт (1 Кбайт = 1024 байт). Мбайт (1Мбайт = 1024 Кбайт) и т.д.

Для хранения информации и работы с ней в компьютерах используются различные системы кодировки (ANSI, ASCII, Кои-8 и пр.). Например, американский национальный стандартный код для обмена информацией ASCII 8-ю битами обеспечивает представление 256 символов, включая символы для национальных алфавитов.

Сочетанием двоичных цифр (битов) можно представить любое значение. Но соглашению биты в байте пронумерованы от 0 до 7 справа налево. Двоичная система счислении, как и привычная, для нас десятичная, является позиционной, и значение двоичною числа определяется относительной позицией каждого бита и наличием единичных битов. Для того, чтобы «прочитать» число, записанное в какой-либо позиционной системе счисления (перевести его в десятичное), можно воспользоваться следующей формулой:



где *Р* - основание системы счисления (количество цифр алфавита), ап­позиция цифры в числе справа налево, начиная с 0.

Например, набор битов 01000001 представляет число 65



***Двоичная арифметика***

Так как ПК выполняет арифметические операции только в двоичном формате, полезно иметь представление о двоичной арифметике. Если быть более точным, то из всех простейших арифметических операций компьютер может выполнять только одну - сложение. Остальные операции выполняются через сложение: вычитание производится через сложение с отрицательным числом, умножение - через многократное суммирование, деление - через многократное вычитание.

Рассмотрим, как в ПК производится суммирование и вычитание двоичных чисел.

Двоичный алфавит состоит из цифр 0 и 1, а правила суммирования следующие:

0+0=0

1+0=1

1+1=(1)0

1+1+1=(1)1.

Здесь необходимо обратить внимание на перенос единичного бита в последних двух случаях.

Например, при сложении чисел 01000001 и 00101010 (65 и 42) получим следующий результат.

Двоичная Десятичная

01000001 65

00101010 42

\_\_\_\_\_\_\_\_ \_\_\_

01101(111 107.

Представленные выше числа являются положительными, что показывает наличие нуля в самом левом (старшем) разряде. Отрицательные двоичные числа содержат единичный бит в старшем разряде и выражаются двоичным дополнением. Для получения двоичного дополнения необходимо инвертировать все биты исходного числа и к результату прибавить 1.

Например, для получения двоичного дополнения числа 65 (01000001) необходимо сделать инверсию всех битов (10111110) и прибавить единицу. Полученный набор битов (10111111) представляет число-65.

Для того, чтобы «прочитать» отрицательное двоичное число, необходимо определить его абсолютное значение и приписать знак «минус». При получении модуля отрицательного числа необходимо повторить операции - инвертировать все биты и прибавить 1.

Доказать правильность приведенного выше можно простым сложением: при суммировании противоположных чисел мы должны получить ноль. В нашем примере

01000001 (-65)

10111111 (-65)

\_\_\_\_\_\_\_\_ \_\_\_\_

0000000

Все восемь бит имеют нулевое значение. Перенос единичного бита потерян. Такой перенос является корректным, так как был перенос единицы в знаковый (старшин) бит и из него. Если же при двоичном суммировании был перенос только в знаковый разряд, либо только га знакового, то фиксируется ошибка сложения.

***Шестнадцатеричное представление***

Специалисты разработали «стенографический» метод представления двоичных данных, по которому байт делится пополам и каждые полбайта выражаются соответствующей шестнадцатиричной цифрой: десятичными цифрами (0 - 9) и А (10), В (11), С (12), D (13), Е (14), F (15). Так как цифр всего 16, система называется шестнадцатеричной.

Для индикации шестнадцатеричного числа непосредственно после него ставится символ Н, например, 25Н (десятичное представление 37). 16-теричное число всегда начинается с десятичной цифры 0-9, таким образом, число B8Н записывается как 0B8Н.

Шестнадцатеричная система счисления также является позиционной, поэтому прочитать «десятичный» эквивалент шестнадцатиричного числа можно с применением правила, описанного выше.

**2. Табличный процессор Excel: типы данных**

Окно табличного процессора Excel предназначено для ввода электронной таблицы и содержит следующие элементы:

* стандартные элементы окна Windows;
* поле имени содержит имя или адрес активной ячейки или диапазона ячеек;
* строка формул предназначена для отображения и редактирования содержимого активной ячейки;
* строка состояния выводит информацию о режиме работы, состоянии индикаторов режимов и клавиатуры.

В рабочей области окна расположена рабочая книга. Рабочая книга - это файл, предназначенный для хранения электронной таблицы, имеет расширение.xls. Рабочая книга состоит из рабочих листов. По умолчанию во вновь создаваемой книге содержится 3 рабочих листа. Пользователь может управлять этим количеством с помощью установки значения параметра Листов в новой книге на вкладке Общие диалогового окна команды Параметры меню Сервис.

Каждый рабочий лист имеет имя (ярлык рабочего листа). По умолчанию листы именуются Лист1, Лист2, Лист3, Диаграмма1, они могут быть следующих типов:

* рабочий лист - электронная таблица;
* лист диаграммы - графическое представление данных электронной таблицы.

Рабочий лист представляет собой сетку из строк и столбцов. Максимальный размер рабочего листа - 256 столбцов, 65536 строк. Столбцы именуются латинскими буквами от А до Z и от АА до IV. Строки именуются числами от 1 до 65536

На пересечении строки и столбцов рабочего листа расположены ячейки (клетки). Каждая ячейка имеет адрес, который образуется: <имя столбца><имя строки>, например А10. Ввод и редактирование данных производится в активной ячейке. Активная ячейка выделяется жирной рамкой Ее имя содержится в поле имени. Существует также понятие диапазона ячеек. Диапазон (блок, интервал) ячеек - это прямоугольная область в таблице, содержащая несколько выделенных ячеек. Адрес диапазона образуется как: <адрес 1-й ячейки>: <адрес последней ячейки>, например А1:А10, A10:D20.

В ячейки рабочего листа можно вводить данные двух типов: константы и формулы. Константы - это значения, которые не изменяются до тех пор, пока их не изменяют преднамеренно. Константы могут быть следующих типов: числовые, текстовые (надписи), даты и времени суток, а также двух специальных типов - логические значения и ошибочные значения.

Число в Excel может состоять только из следующих символов: цифры от 0 до 9, +, -, (,), /, $, %, (.), Е, е. Запятая в числе интерпретируется как разделитель десятичных разрядов. Символ разделителя может быть изменен в приложении Язык и стандарты панели управления Windows.

Существуют следующие правила ввода чисел:

1. Если ввод числа начинается со знака «+» или «-», пиксел опускает «+» и сохраняет «-», интерпретируя введенное значение как отрицательное число.
2. Числовые значения, заключенные в круглые скобки, интерпретируются как отрицательные. Например, (5) интерпретируется, как -5.
3. Символ Е или е используется при вводе чисел в поненциальном представлении. Например, 1Е6 интерпретируется как 1 000 000 (единица, умноженная на десять в шестой степени).
4. При вводе больших чисел позволяется вставлять робел для отделения сотен от тысяч, тысяч от миллионов и т. д. При таком вводе числа в ячейках появляются пробелами, а в строке формул без пробелов.
5. Если ввод числа начать со знака денежной единицы, к ячейке будет применен денежный формат.
6. Если ввод числа закончить знаком %, к ячейке будет применен процентный формат.
7. Перед вводом рациональной дроби, чтобы Excel не интерпретировал ее как дату, следует ввести 0 и пробел, например 3/4 ввести 0 3/4. Числа можно вводить в различных форматах. В Excel имеется набор стандартных числовых форматов, которые ри желании могут быть изменены. Также можно создать собственные пользовательские форматы.

Обычно ввод чисел осуществляется в общем числовом формате. В соответствии с ним числа в ячейке отражаются в привычном виде. Если длина числа не превышает ширину ячейки, то оно отображается в том виде, в котором вводится, если превышает, то число будет выведено в экспоненциальной форме. Если значение числа превышает допустимое по формату значение, то в ячейке выводится признак переполнения - ####. Изменить, стандартный формат можно на вкладке Вид команды Ячейки меню Формат.

Ввод текста аналогичен вводу числовых значений. Текст - это произвольная последовательность символов, не воспринимаемая как число, дата, время суток или формула. При вводе длинного текста, который не может быть полностью отображен в одной ячейке, Excel способен вывести его, перекрывая соседние ячейки. Но при этом текст все равно будет храниться только в одной ячейке. При вводе текста в ячейку, которая перекрыта содержимым другой ячейки, перекрывающий текст обрезается. В строке формул при активизации ячейки с длинным текстом отображается весь хранящийся в ней текст. Длинный текст в ячейке можно увидеть, расширив столбец двойным щелчком на границе столбца в его заголовке. Ширина столбца настроится по максимальной ширине значений в этом столбце. Облегчить чтение длинных текстовых значений может также перенос текста. Этот режим позволяет вводить длинные текстовые значения с переносом на следующие строки без наложения текста на другие ячейки. При этом Excel увеличивает высоту строки, которая содержит ячейку с дополнительными строками. Чтобы установить этот режим, следует включить флажок Переносить по словам на вкладке Выравнивание команды Ячейки меню Формат.

Иногда требуется ввести число со знаком «+» перед ним. При простом наборе «плюс число» Excel воспримет ввводимое значение как числовое, и знак «+» опустит. Чтобы заставить Excel обращаться со специальными символами, как с обычными, нужно ввести числовой текст. Числовой текст может состоять из текста и чисел или только из чисел. Если значение, вводимое в ячейку, будет состоять из текста и чисел, оно будет интерпретироваться как текстовое. Для того чтобы создать текстовое значение, состоящее целиком из числовых символов, следует начать ввод с апострофа или ввести сначала знак равенства, а затем значение, заключенное в кавычки. Знак равенства с кавычками или апостроф появляются в строчке формул, но не выводятся в ячейке. В то время как числовые значения по умолчанию выравниваются по правому краю, числовой текст, как и обычный, выравнивается по левому.

В Excel дата и время суток интерпретируются как числа. Основной единицей измерения времени в Excel являются сутки. Они представляются последовательными десятичными значениями от 1 до 65380. Базовая дата, представляемая десятичным числом 1, - это воскресенье, 1 января 1900 г. Максимальное десятичное значение даты; 65380 представляет 31 декабря 2078 года. При вводе даты г Excel сохраняет ее в виде десятичного значения, которое равно количеству дней между заданной и базовой датой. Время суток - это десятичная дробь, которая представляет часть суток между их началом (12:00 ночи) и заданным временем. Например, 12:00 дня представляется значением 0,5.

Внешнее представление в ячейках рабочего листа зависит от формата, назначенного ячейке. В форматах даты и времени используются следующие разделители: «.», «/ », « - » - для даты; «> - для времени.

При вводе даты между 1920 и 2010 гг. можно указывать только две последние цифры года. При вводе даты вне этого диапазона год нужно записывать полностью.

Чтобы ввести текущее время в ячейку или в формулу, следует одновременно нажать клавиши Ctrl, Shift и «:». Для ввода текущей даты в ячейку или формулу следует одновременно нажать клавиши Ctrl и «;».

При вводе даты и времени нет различий между строчными и прописными буквами. При использовании 12-часового формата после ввода времени через пробел следует ввести АР (А) - для ввода времени до полудня и РМ (Р) - для ввода времени после полудня. Например, 3:00 РМ означает 15:00. Дату и время можно ввести в одну ячейку. Тогда их следует разделить пробелом.

Данные типа даты и времени суток могут участвовать в вычислениях (сложение, вычитание), а также входить в состав формул в виде текста (тогда их необходимо заключить в двойные кавычки).

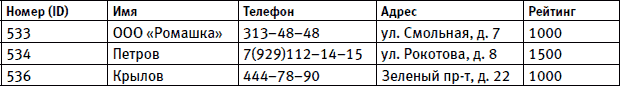
**3. СУБД Access: запрос с параметром (принцип работы, этапы создания)**

Программа Access представляет собой *реляционную* СУБД. Это означает, что база данных Access состоит из *взаимосвязанных* таблиц.

### Таблицы

Таблица базы данных – это обычная таблица из строк и столбцов. Ниже приведен пример таблицы (табл. 1), содержащей сведения о клиентах компании.

##### Таблица 1 - Клиенты



### 

### Связи между таблицами

*Связь* между двумя таблицами организуется посредством общих полей этих таблиц. Приведем пример таблицы (табл. 2), связанной с таблицей **Клиенты.**

##### Таблица 2 - Заказы компании



В этом примере общими полями являются поле **Номер (ID)** таблицы **Клиенты** и поле **Клиент** таблицы **Заказы**. А именно, поле **Клиент** содержит номер клиента из таблицы **Клиенты**. Таким образом, каждая запись таблицы **Заказы** ссылается на какую-либо запись таблицы **Клиенты**.

### Ключевые поля

В реляционной базе данных существует два вида ключевых полей: первичный ключ и вторичный ключ. Для идентификации записей в таблице используется *первичный ключ* – поле или набор полей таблицы, чьи значения однозначно определяют запись. Соответственно значения первичного ключа должны быть уникальны, то есть в таблице не должно быть двух строк с одинаковыми значениями первичного ключа. Практически всегда в качестве первичного ключа используется специальное поле, содержащее номер записи (например, поле **Номер (ID)** в таблице **Клиенты**). Программа Access присваивает записям номера автоматически и гарантирует уникальность номеров. *Вторичным ключом* называется поле дочерней таблицы, содержащее значения первичного ключа родительской таблицы. Например, поле **Клиент** в таблице **Заказы** является вторичным ключом. Таким образом, связи между таблицами организуются с помощью ключевых полей.

## Проектирование структуры данных

Построение базы данных начинается с проектирования. Чтобы понять, какая структура базы будет для вас наиболее удобной и полезной, следуйте нижеприведенным этапам проектирования.

1. Для начала необходимо выяснить, каково назначение базы данных и какую информацию необходимо получать из базы. Составьте список типичных запросов к базе данных:

• список ваших клиентов с контактными данными;

• список заказов конкретного клиента;

• список всех заказов за определенный период;

• какие товары в каком количестве были заказаны за определенный период и т.д.

2. Выделите из этого набора классы объектов, о которых нужно будет получать информацию из базы данных. Объекты каждого класса будут храниться в отдельной таблице, и вы тем самым получите список таблиц будущей базы данных. В примере из предыдущего пункта целесообразно выделить следующие классы: клиенты, заказы и товары. Таким образом, в базе данных будет три таблицы.

3. Выпишите интересующие вас атрибуты объектов каждого класса. Например, для каждого клиента необходимо хранить в базе его имя (или название организации) и контактную информацию, для товара – наименование и цену. В результате вы получите список полей каждой таблицы. Учтите, что в таблице должен быть первичный ключ, и добавьте в каждую таблицу дополнительное поле – уникальный номер записи. Таким образом, были перечислены атрибуты и установлены связи между таблицами. Например, если в таблице заказов есть поля **Товар** и **Клиент**, содержащие код товара и номер клиента, то таблица **Заказы** ссылается и на таблицу **Товары**, и на таблицу **Клиенты**.

## Этапы работы с базой данных

Итак, если вы создаете базу данных «с нуля», вам предстоит выполнить следующие операции.

1. Создание базы данных или открытие уже созданной;

2. Создание таблиц, настройка полей таблиц и связей между таблицами;

3. Заполнение таблиц информацией;

4. И, наконец, получение информации из базы данных – то, ради чего она и создавалась. Для поиска, отбора, сортировки и агрегации данных вы можете создать *запросы* (см. гл. 5), для наглядного представления данных – *отчеты* (см. гл. 7). Теперь перейдем к практическим действиям и рассмотрим первый этап работы с базой данных – ее открытие или создание.

### Открытие существующей базы данных

Если база данных уже была создана, то ее можно открыть, щелкнув кнопкой мыши на ее названии в области **Открыть последнюю базу данных**. Если нужной базы данных нет в списке, щелкните на ссылке **Другие**. На экране появится стандартное окно Windows для открытия файла. В этом окне выберите файл базы данных и нажмите кнопку **Открыть**. Кроме того, вы можете воспользоваться кнопкой (рис. 1)



Рис. 1

в левом верхнем углу окна, в появившемся меню выбрать пункт **Открыть**, а затем выбрать файл в стандартном окне Windows.

### Создание пустой базы данных

Если вы хотите самостоятельно разработать структуру базы данных, создать таблицы и ввести данные, вначале необходимо создать пустую базу данных. Для этого выполните следующие действия.

1. Щелкните кнопкой мыши на значке **Новая база данных** в разделе **Новая пустая база данных** в центральной части начального окна (см. рис.  2) либо нажмите кнопку



Рис. 2

в левом верхнем углу окна, а затем в появившемся меню выберите пункт **Создать**.

2. В правой части окна появится область **Новая база данных**. В поле **Имя файла** введите имя новой базы данных. Нажмите кнопку (рис .3)



Рис. 3

для выбора папки, в которой будет храниться файл базы данных.

3. Нажмите кнопку **Создать**. Откроется новая база данных с единственной пустой таблицей, и программа перейдет в режим редактирования таблицы.

### Создание копии базы данных

Следующий способ создания базы данных – скопировать существующую базу, чтобы продолжить работу с копией. Для этого выполните следующие действия.

1. Откройте для чтения базу данных, которую необходимо скопировать. Нажмите кнопку (рис. 4)



Рис. 4

в левом верхнем углу окна, затем в появившемся меню выберите пункт **Открыть**. На экране появится стандартное окно Windows для открытия файла.

2. Щелкните кнопкой мыши на нужном файле базы данных и нажмите кнопку, находящуюся справа от кнопки **Открыть** (рис. 4).

В меню открытия файла выберите пункт **Открыть для чтения** или пункт **Монопольно для чтения**.

• После открытия базы данных в любом из этих двух режимов вы не сможете вносить в нее изменения. Под лентой (кнопочным меню) возникнет панель сообщений с предупреждением, что база данных открыта только для чтения.

• Если выбранная база данных недоступна другим пользователям, пункты меню **Открыть для чтения** или **Монопольно для чтения** равнозначны.

• Вы сможете открыть базу данных в режиме **Открыть для чтения**, если она не открыта в данный момент другим пользователем в режиме **Монопольно**. Другие пользователи смогут продолжать работу с базой и открывать ее в любом режиме доступа, кроме режима **Монопольно**. При этом вы будете просматривать последнюю сохраненную версию, несохраненные изменения будут проигнорированы.

• Открытие базы данных в режиме **Монопольно для чтения** возможно, только если она в данный момент не используется другим пользователем в режиме редактирования (**Открыть** или **Монопольно**). При этом другие пользователи не смогут открывать эту базу данных в режиме редактирования.

### Создание базы данных на основе шаблона

В Access 2007 включены шаблоны баз данных. Их можно использовать для решения типичных задач или для учебных целей, поскольку в их состав входят таблицы, уже заполненные данными, готовые отчеты и формы для ввода информации. Чтобы создать базу данных с использованием шаблона, выполните следующие действия. 1. В левой области начального окна Access (см. рис. 1.2) выберите категорию шаблонов. В центре окна отобразится список шаблонов этой категории. Щелкните кнопкой мыши на значке подходящего шаблона (рис.  5). 2. В правой части окна появится область с названием шаблона и его категории. В поле **Имя файла** введите имя новой базы данных. С помощью кнопки



Рис. 5

выберите папку, в которой будет храниться файл базы данных. 3. Если требуется присоединить новую базу данных к веб-узлу SharePoint, установите флажок **Создание и присоединение базы данных к узлу Windows SharePoint Services**.

4. Нажмите кнопку **Создать** (для шаблонов Microsoft Office Online она называется **Загрузка**). Access сгенерирует базу данных по указанному шаблону и откроет ее в режиме редактирования таблицы. 5. Если в созданной базе данных Access обнаружит потенциально опасное содержимое, то под лентой появится панель сообщений с предупреждением **Часть содержимого базы данных отключена**. Чтобы открыть содержимое базы данных полностью, в том числе заблокированную часть, нажмите кнопку **Параметры** панели сообщений. В появившемся окне **Параметры безопасности Microsoft Office** установите переключатель в положение **Включить это содержимое** и нажмите кнопку **ОК**.

## Список используемой литературы

## 1. Глушаков С.В. Информатика. Курс лекций для студентов первого курса МГУ 2009 г.

## 2. Первишин А.Г. «Основы информатики» 2008 г.

## 3. Пелевин В.В., Реутова Р.А. «Учение об информатике» 2009 г.

## 4. Панов А.А. «Основы Microsoft Office» 2008 г.

## 5. Ростовский А.П., Керимов С.Р. «Базы данных глава 4: Основы СУБД Access» 2009 г.