**Дисциплина: Офисное программирование**

**Группы: 1ПКС-17, 2ПКС-17**

**Дата: 01.10.2020 г.**

**ЗАДАНИЕ:**

1. **Освоить лекционный материал.**
2. **Подчеркнутые определения законспектировать в тетрадь.**

**Тема: Встроенные функции VBA**

В VBA имеется большой набор встроенных функций и процедур, упрощающих программирование, которые можно разделить на следующие категории:

* математические;
* функции проверки типов;
* функции обработки строк;
* функция форматирования;
* функции преобразования форматов;
* функции даты и времени.

**Функции проверки типов**

Функции проверки типов позволяют определить, какой тип данных имеет переменная.

IsArray(х) – функция проверяет является ли переменная массивом;

IsDate(х) определяет является ли переменная датой;

IsError(х) определяет является ли переменная кодом ошибки;

IsNull(х) определяет является ли переменная пустым значением;

IsNumeric(х) определяет является ли переменная числовым значением;

IsObject(х) определяет является ли переменная объектом.

**Функция форматирования**

Функция форматирования возвращает значение типа Variant (String), содержащее выражение, оформленное согласно синтаксиса функции:

Format(Выражение[,Формат [,Первый день недели[,Первая Неделя Года]]]), где:

Выражение – обязательный аргумент (любое допустимое выражение – комбинация ключевых слов, операторов, переменных и констант, результатом которой является строка, число или объект);

Формат – необязательный параметр (любое допустимое именованное или определенное пользователем выражение формата).

Если к имени функции добавляется знак $, то функция возвращает значение типа String. При создании собственного числового формата можно использовать следующие символы: 0 – резервирует позицию цифрового разряда. Отображает цифру или нуль. Если у форматируемого числа есть какая-нибудь цифра в этой позиции разряда, где в строке формата находится 0, функция отображает эту цифру, если нет, то в этой позиции отображается нуль; "#" - действие данного символа аналогично действию 0 с той лишь разницей, что незначащие нули не отображаются;

. – резервирует позицию десятичного разделителя, определяет, сколько разрядов необходимо отображать слева и справа от десятичной точки;

% - резервирует процентное отображение числа; . – разделяет сотни от тысяч.

**Функции преобразования форматов**

Val(строка) – возвращает числа, содержащиеся в строке, как числовое значение соответствующего типа.

Str(число) – возвращает значение типа Variant (String), являющееся строковым представлением числа.

Кроме функций Val и Str имеется ряд функций по преобразованию типов выражений.

**Функции обработки строк**

Среди множества функций можно выделить следующие:

Chr(код) – преобразует ASCII – код в строку. Например, Chr(10) осуществляет переход на новую строку, Chr(13) – возврат каретки;

Mid(string, pos[,length]) – возвращает подстроку строки, содержащую указанное число символов, где String – строковое выражение, из которого извлекается подстрока;

Pos – позиция символа в строке String, с которого начинается нужная подстрока;

Length – число возвращаемых символов подстроки;

Len(строка) – возвращает число символов строки.

В таблице 1 вы можете найти информацию об основных строковых функциях VBA.

|  |  |
| --- | --- |
| Таблица 1. Строковые функции | |
| **Функция** | **Описание** |
| Len(string) | Возвращает длину строки. Например, длина строки "Добрый день" составляет 11 символов - учитывая пробел. Выходное значение имеет тип Long |
| LCase(string) | Возвращает строку, все символы которой записаны в нижнем регистре. Например, строка "Привет" превратится в "привет" |
| UCase(string) | Возвращает строку, все символы которой записаны в верхнем регистре. Например, для "Привет" мы получим "ПРИВЕТ" |
| String(number, character) | Возвращает строку, состоящую из number символов character |
| Left(string, length) | Возвращает length символов, начиная с первого левого символа строки string |
| Right(string, length) | Возвращает length символов, начиная с самого правого символа строки string |
| LTrim(string) | Возвращает строку, в которой вырезаны все пробелы слева |
| RTrim(string) | Вырезает из строки все пробелы справа |
| Trim(string) | Вырезает из строки все пробелы слева и справа |
| Mid(string, start[, length]) | Вырезает из строки string с позиции start length символов |
| Asc(string) | Возвращает ASCII-код первого символа строки |
| Chr(charcode) | Возвращает символ, соответствующий коду символа |

ASCII (American Standard Code For Information Intercharge) - американский стандартный код для обмена информацией. Каждый символ в ASCII закодирован восьмибитным кодом. В результате получается таблица, в которой каждому управляющему символу, цифре, букве латинского или национального алфавитов сопоставлен свой код. Коды записывают в различных представлениях - в основном - в шестнадцатеричном и десятичном.

Коды в диапазоне 0-31 имеют управляющие символы. Символ возврата каретки (тот самый, который вставляется в документ при нажатии клавиши **Enter** ) имеет код 13.

Коды в диапазоне 32-127 имеют латинские символы, цифры, знаки препинания - эта часть таблицы остается постоянной для различных кодовых таблиц. Например, пробел имеет код 32, точка - 46. Диапазон 48-57 занимают цифры от 0 до 9, диапазон 65-90 занимают заглавные латинские буквы от A до Z, диапазон 97-122 - строчными буквы a-z.

В диапазоне 128-225 расположены символы национальной кодировки. В русифицированных версиях MS Windows обычно применяется кодовая страница MS Windows 1251. В ней коды 192-223 имеют заглавные буквы от А до Я, 224-255 - строчные буквы от а до я.

Используя коды символов можно вводить в документы (или записывать в файлы, создаваемые программно) символы, которые нельзя ввести с клавиатуры.

**Функции даты и времени**

Функции даты и времени позволяют проводить различные действия с переменными типа Date.

* **Date**() — возвращает текущую системную дату. Установить ее можно при помощи одноименного оператора, например, так:

|  |
| --- |
| **Date** = #5/12/2006# |

* **Time**() — возвращает текущее системное время
* **Now**() — возвращает дату и время вместе.
* **DateAdd**() — возможность добавить к дате указанное количество лет, кварталов, месяцев и так далее — вплоть до секунд. Интервалы(год, месяц и т.д.) указываются в текстовом формате. Список допустимых значений:  
  "yyyy" Год  
  "q" Квартал  
  "m" Месяц  
  "y" День года  
  "d" День  
  "w" День недели  
  "ww" Неделя  
  "h" Час  
  "n" Минута  
  "s" Секунда

Сначала указывается интервал, затем сколько единиц добавить и самый последний аргумент -

к какой дате (включая время). Например, чтобы добавить 3 года к текущей дате-времени, надо записать функцию так:  
*MsgBox DateAdd("yyyy", 3, Now)*

Аналогичным образом выполняется вычитание:

*MsgBox DateAdd("yyyy", -3, Now)*

* **DateDiff**() — возможность получить разницу между датами (в единицах от лет до секунд).

|  |
| --- |
| **Dim** lDaysCnt **As** **Long**  lDaysCnt = DateDiff("d", "20.11.2012", Now)  MsgBox "С 20.11.2012 прошло дней: " & lDaysCnt |

* Первый аргумент определяет период времени, в котором необходимо вернуть разницу между датами. Допустимые значения:  
  "yyyy" Год  
  "q" Квартал  
  "m" Месяц  
  "y" День года  
  "d" День  
  "w" День недели  
  "ww" Неделя  
  "h" Час  
  "n" Минута  
  "s" Секунда

Наиболее полезна DateDiff при вычислении полных лет. Например, чтобы вычислить сколько лет на текущий момент человеку, в зависимости от даты рождения, можно использовать функцию так:

|  |
| --- |
| MsgBox DateDiff("yyyy", "20.12.1978", Now) |

* **DatePart**() — функция возвращает указанную часть даты (например, только год, только месяц или только день недели), на основании заданной даты. Часто применяется для получения номера недели для даты.  
  Первый аргумент - период времени. Принимаемые значения те же, что и для функции DateDiff(годы, месяцы, недели и т.д.)  
  Второй аргумент - непосредственно дата, часть которой необходимо получить:

|  |
| --- |
| MsgBox "Номер недели года: " & DatePart("ww", Now) |

* **DateSerial**() — возможность создать значение даты, задавая месяц, год и день числовыми значениями:

|  |
| --- |
| MsgBox DateSerial(2012, 6, 7) |

* **DateValue**()— делает то же, что и DateSerial(). Отличия — в формате принимаемых значений. Эта функция в качестве аргумента принимает дату в текстовом формате и преобразует её в формат даты:

|  |
| --- |
| MsgBox DateValue("07.06.12") |

Аналогичным образом (для времени) работают <emf**TimeSerial**() и **TimeValue**()</emf

* **Day**(), **Year**(), **Month**(), **Weekday**(), **Hour**(), **Minute**(), **Second**() — специализированные заменители функции DatePart(), которые возвращают нужную часть даты/времени (которую именно — видно из названия).
* **MonthName**() — возвращает имя месяца словами по его номеру. Возвращаемое значение зависит от региональных настроек. Если они русские, то вернется русское название месяца.
* **Timer**() — возвращает количество секунд, прошедших с полуночи.