

*Авторская разработка преподавателя информатики и ИКТ
Курбатова Вячеслава Валерьевича, г. Гусев*

Разбор домашнего задания

	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15
1	Горизонтальная раскладка букв введенной строки по ячейкам														
2															
3	Вводимое значение	Иванов Василий					Раскладка по горизонтали								
4															
5	Смещение по строке	8					Раскладка по вертикали								
6	Смещение по столбцу	2													
7															
8		И	в	а	н	о	в		В	а	с	и	л	и	й
9		в													
10		а													
11		н													
12		о													
13		в													
14															
15		В													
16		а													
17		с													
18		и													
19		л													
20		и													
21		й													
22															

**Worksheets("НазваниеЛиста").Cells(Строка[+ смещение – 1], _
 Столбец[+ смещение – 1]) = Символьная переменная**

Программа:

```
Sub RaskladkaGor_Click()    ' Раскладка текста по горизонтали

For x = 1 To Len(Cells(3, 3))    ' Цикл от первого до последнего
символа «вводимого значения»
    simvol = Mid(Cells(3, 3), x, 1)    ' Берем очередной символ
    Cells(Cells(5, 3), Cells(6, 3) + x - 1) = simvol ' Записываем на
                                                "Лист1 " со смещением по столбцу
    Worksheets ("Лист2").Cells(.Cells(5, 3), Cells(6, 3) + x - 1) = simvol
                                                ' Записываем на "Лист2" со смещением по столбцу
Next x

End Sub
```

Программа:

```
Sub RaskladkaVer_Click()  ' Раскладка текста по вертикали  
  
For x = 1 To Len(Cells(3, 3))      ' Цикл от первого до последнего  
                                   ' символа «вводимого значения»  
    simvol = Mid(Cells(3, 3), x, 1)      ' Берем очередной символ  
    Cells(Cells(5, 3) + x - 1, Cells(6, 3)) = simvol      ' Записываем на  
                                   ' "Лист1" со смещением по строке  
    Worksheets ("Лист2").Cells(Cells(5, 3) + x - 1, Cells(6, 3)) = simvol  
                                   ' Записываем на "Лист2" со смещением по строке  
  
Next x  
  
End Sub
```

Урок

Дата и время

Date() — возвращает текущую системную дату.

MsgBox "Системная дата " & Date()

Установить системную дату можно при помощи одноименного оператора, например, так:

Date = #5/12/2006#

Time() возвращает текущее системное время,
а ***Now()*** — дату и время вместе.

**MsgBox "Системное время " & Time() & "
Системная дата и время " & Now()**

DateAdd() — возможность добавить к дате указанное количество лет, кварталов, месяцев и так далее — вплоть до секунд.

DateAdd("interval", number, "date")

yyyy - Год

q - Квартал

m - Месяц

y - День года

d - День месяца

w - День недели

ww - Неделя

h - Часы

n - Минуты

s - Секунды

DateAdd("m", 1, "31-Jan-95")

Пример 1: *' Вычисляем дату на 15 дней позднее текущей*

Dim Today

Dim NewDate As Date

Today=Date

' узнаем текущую системную дату

NewDate=DateAdd("y",15,Today)

' Преобразуем в строку и

' выводим в заголовке формы

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

DateDiff(Interval, Date1, Date2)

Interval – имеет те же параметры, как и ***DateAdd()***

Пример 2: *' Вычисляем сколько дней осталось до 21 века*

<i>Dim Today as Date</i>	<i>' сегодняшний день</i>
<i>Dim Millenium As Date</i>	<i>' 1 день 21 века</i>
<i>Dim retval</i>	<i>' возвращаемое значение</i>

```
Today=Date                    ' узнаем текущую системную дату  
Millenium="01.01.01"  
retval=DateDiff("d", Today, Millenium)
```

DatePart() — функция, которая возвращает указанную вами часть даты (например, только год, только месяц или только день недели).

DatePart(Interval, Date)

Interval — имеет те же параметры, как и ***DateAdd()***

Пример 3: *' Вычисляем какой сейчас месяц*

MsgBox "Сейчас " & DatePart("m", Now) & " месяц года "

DateSerial() — возможность создать значение даты на основе передаваемых символьных значений.

То же самое делает ***DateValue()***, отличия — в формате принимаемых значений.

Аналогичным образом (для времени) работают ***TimeSerial()*** и ***TimeValue()***.

Пример 4: *' Вычисляем дату, предшествующую*
Dim MyDate, DVal ' на 5 дней дате 1 августа 1966 года
MyDate = DateSerial(1966, 8, 1-4) ' возвращает 27.07.66
DVal = DateValue("25 Июня 1969") ' возвращает 25.06.69
MsgBox MyDate & " " & DVal

Пример 5: *' При вычислении даты, возможно*
' вычисление ее прямо в формуле
DateSerial(1990 - 10, 8 - 2, 1 - 1) ' возвращает 05.31.80

Day() (а также ***Year()***, ***Month()***, ***Weekday()***, ***Hour()***, ***Minute()***, ***Second()***) — специализированные заменители функции ***DatePart()***, которые возвращают нужную вам часть даты.

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.

Запуск процедуры через определенное время

Application.OnTime EarliestTime, Procedure

EarliestTime — время запуска процедуры

Procedure — имя процедуры

В следующем примере демонстрируется, как запустить процедуру «Очистка» на выполнение через 15 секунд от текущего времени:

Application. OnTime Now +TimeValue("00:00:15"), "Очистка"

Практическая работа:

Помещение в ячейку электронных часов

```
Sub UpdateTime() ' Макрос должен быть записан в разделе Module
    Dim VarNextCall As Variant
    Cells(1,1).Value = Now      ' Запись в ячейку (A1) текущей даты и времени
    Cells(2,1).Value = Format(Now, "Long Time") ' Дата и время в длинном формате
    Cells(1,3) = "Cells(1,1).Value = Now" ' В ячейке C1 и C2 выводим формулы,
        ' благодаря которым были выведены данные значения
    Cells(2, 3) = "Cells(2, 2).Value = Format(Now, " & Chr(34) & "Long Time" + Chr(34) + ")"
        ' Чтобы напечатать кавычки, в формуле указываем Chr(34)
    VarNextCall = TimeSerial(Hour(Now), Minute(Now), Second(Now) + 1) ' Записываем в
    ' VarNextCall время, когда вызвать этот макрос, в следующий раз (через 1 секунду)
    Application.OnTime VarNextCall, " UpdateTime" ' Уведомляем Excell в необхо-
End Sub                                ' димости вызова макроса
```

Рекурсивный запуск процедуры из самой же себя

*Запустить макрос **UpdateTime()** можно через кнопку или вызвав из другой процедуры*

Остановить макрос возможно только в редакторе кода, нажав кнопку Pause

Практическая работа:

Будильник

```
Sub Clock() ' Макрос должен быть записан в разделе Module  
Application.OnTime Cells(4,2), "Alarm" ' Уведомляем Excell в необходимости  
' вызова макроса во время указанное в ячейке  
End Sub
```

```
Sub Alarm() ' Макрос должен быть записан в разделе Module  
MsgBox Cells(4,2) ' Сообщение, выводимое в будильнике  
End Sub
```

*Запуск макроса "Clock" в модуле Module1 возможно с помощью кнопки,
не являющейся объектом ActiveX*

Выполнение практического задания