

Microsoft® Small Basic

Условия и циклы

Предполагаемое время работы с этим уроком: 2 часа



Условия и циклы

В этом уроке вы изучите следующее.

Написание программ, содержащих различные инструкции на основе действительности одного или нескольких логических условий.

Написание программ, повторяющих инструкции до наступления определенного события.



Условия в программах Small Basic

Вы хотите указать условия, контролирующее выполнение программы (или даже того, будет ли она выполняться)?

Посмотрим на следующую программу.

```
If Clock.Day = 1 And Clock.Month = 1 Then  
    TextWindow.WriteLine("С Новым Годом!")  
EndIf
```

Эта программа указывает компьютеру на отображение строки «С Новым Годом!», только если сегодня 1 января.

Обратите внимание, что программа содержит ключевые слова **If**, **Then** и **EndIf**.

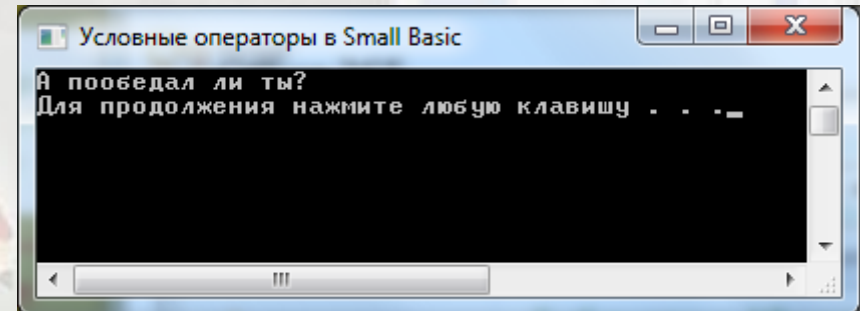
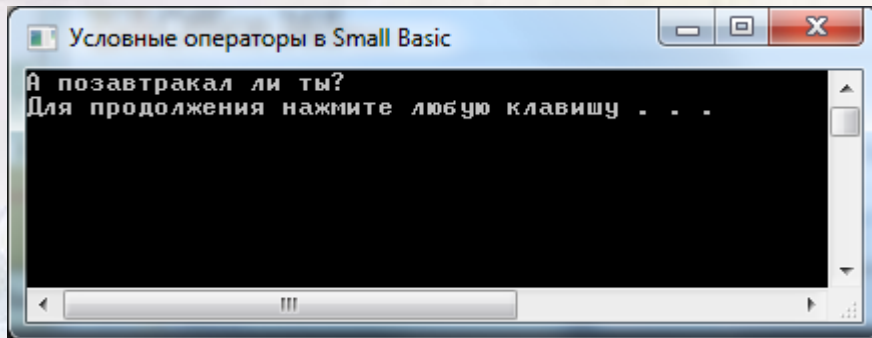
Условия в программах Small Basic

Теперь напишем программу, в которой будет указано альтернативное действие для выполнения в случае ложности условия.

```
If Clock.Hour < 12 Then
    TextWindow.WriteLine("А позавтракал ли ты?")
EndIf

If Clock.Hour > 12 Then
    TextWindow.WriteLine("А пообедал ли ты?")
EndIf
```

В зависимости от времени выполнения программы компьютер отображает один из следующих результатов:



Условия в программах Small Basic

В программировании одного результата можно добиться несколькими способами. Вы, как программист, выбираете лучший путь.

В этом примере вы могли заметить, что второе условие в программе повторяет множество информации из первого условия.

```
If Clock.Hour < 12 Then
    TextWindow.WriteLine("А позавтракал ли ты?")
EndIf

If Clock.Hour > 12 Then
    TextWindow.WriteLine("А пообедал ли ты?")
EndIf
```

Уменьшим повторения, используя ключевое слово **Else**.

```
If Clock.Hour < 12 Then
    TextWindow.WriteLine("А позавтракал ли ты?")
Else
    TextWindow.WriteLine("А пообедал ли ты?")
EndIf
```

Результат обеих программ один, но можно использовать меньшее число ключевых слов **If**, **Then** и **EndIf** при применении ключевого слова **Else**.

Условия в программах Small Basic

Рассмотрим другой пример...

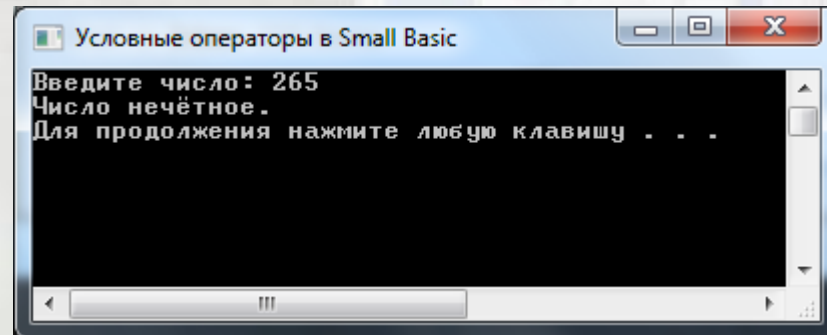
```
TextWindow.Write("Введите число: ")
number = TextWindow.ReadNumber()

remainder = Math.Remainder(number, 2)
If remainder = 0 Then
    TextWindow.WriteLine("Число чётное.")
Else
    TextWindow.WriteLine("Число нечётное.")
EndIf
```

Вы пишете сложную программу и хотите узнать, является ли введенное пользователем число четным или нечетным.

ВЫВОД

Обратите внимание на использование **If**, **Then**, **Else** и **EndIf** в программе.



```
Условные операторы в Small Basic
Введите число: 265
Число нечётное.
Для продолжения нажмите любую клавишу . . .
```

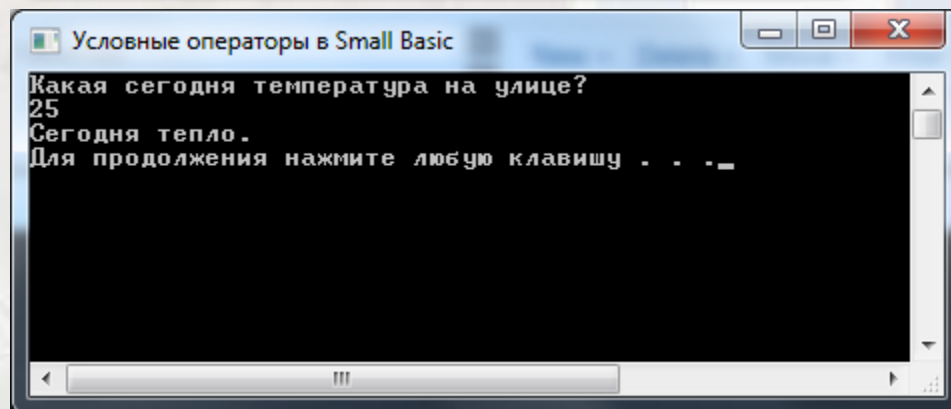
Условия в программах Small Basic

При написании программы можно указать любое количество условий, используя ключевое слово **Elseif**. Также можно указать одну или несколько операций, выполняемых в зависимости от истинного условия при запуске программы.

Рассмотрим это на примере.

В этом примере каждое условие содержит уникальную инструкцию, оцениваемую компьютером. Если компьютер оценивает инструкцию как истинную, он выполняет операцию для этого условия и переходит к концу программы.

```
If temp <= 5 Then
    TextWindow.WriteLine("Сегодня холодно.")
ElseIf temp <= 15 Then
    TextWindow.WriteLine("Сегодня прохладно.")
ElseIf temp <=25 Then
    TextWindow.WriteLine("Сегодня тепло.")
Else
    TextWindow.WriteLine("Сегодня жарко.")
EndIf
```



Циклы в программах Small Basic

Цикл можно использовать для указания выполнения одной или нескольких инструкций более одного раза.

Цикл **For** можно использовать, если известно, сколько раз необходимо повторить инструкции.

Цикл **While** можно использовать, если необходимо повторять выполнение инструкций, пока определенное условие не станет истинным.



Рассмотрим несколько операторов цикла...

Циклы в программах Small Basic

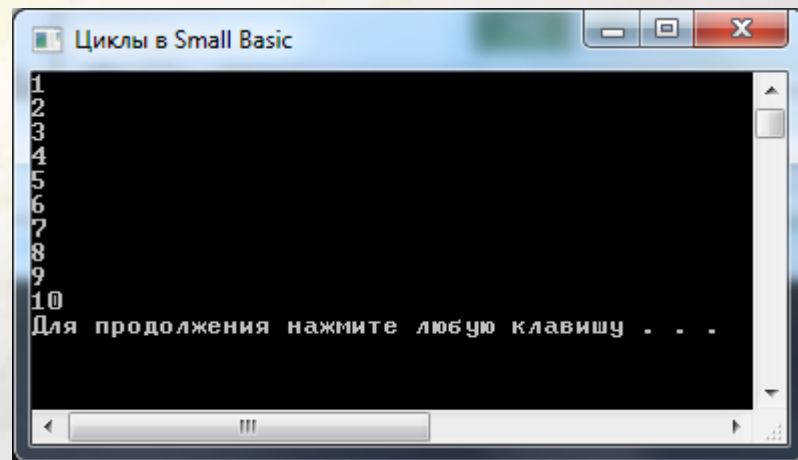
Начнем с программы, содержащей цикл **For ... EndFor**.

В общем, цикл **For ... EndFor** используется для выполнения кода определенное число раз. Для управления этим типом цикла создается переменная для отслеживания числа выполнений цикла.

```
For a = 1 To 10  
    TextWindow.WriteLine(a)  
EndFor
```

Нажмите кнопку  на панели управления.

Запуск (F5)



```
Циклы в Small Basic  
1  
2  
3  
4  
5  
6  
7  
8  
9  
10  
Для продолжения нажмите любую клавишу . . .
```

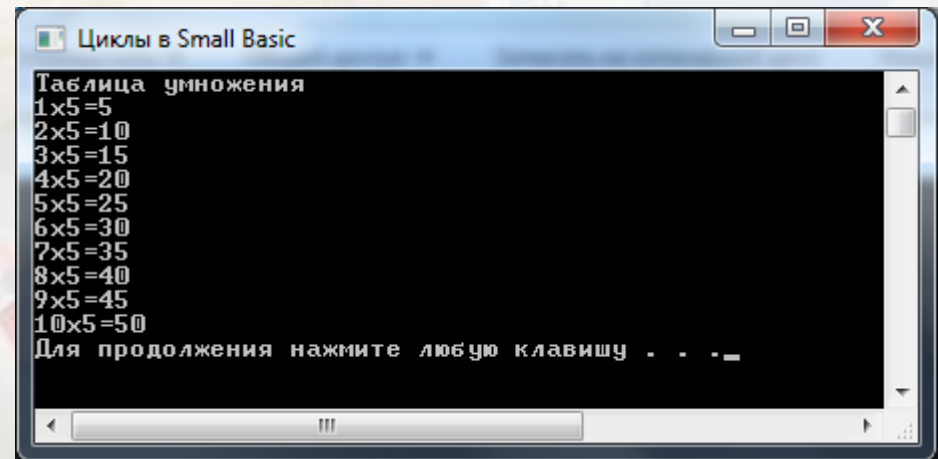
В этом примере переменная содержит значение, увеличивающееся на 1 при каждом выполнении цикла.

Циклы в программах Small Basic

Используем этот подход для вычисления и вывода таблицы умножения для числа 5.

```
TextWindow.WriteLine("Таблица умножения")
number = 5
For a = 1 To 10
    TextWindow.WriteLine(a + "x" + number + "=" + a * number)
EndFor
```

ВЫВОД



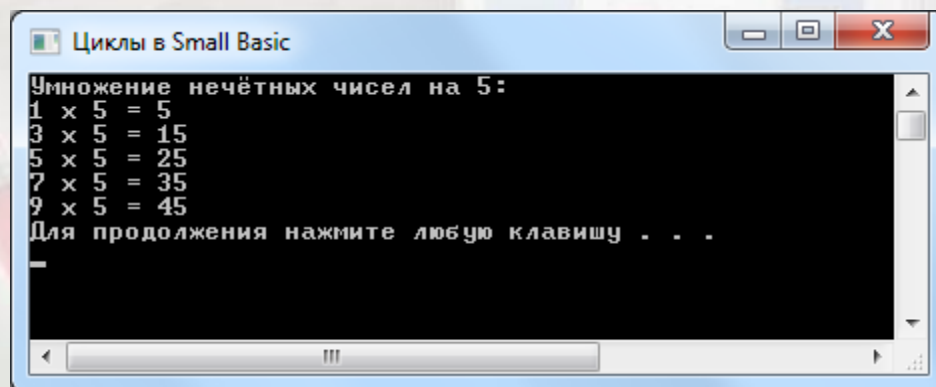
```
Циклы в Small Basic
Таблица умножения
1x5=5
2x5=10
3x5=15
4x5=20
5x5=25
6x5=30
7x5=35
8x5=40
9x5=45
10x5=50
Для продолжения нажмите любую клавишу . . .
```

Циклы в программах Small Basic

В предыдущем примере значение переменной счетчика в цикле **For** увеличивалось на 1 при каждом выполнении цикла. Однако можно увеличить это значение на другое число, если использовать ключевое слово **Step**.

Например, можно увеличивать значение на 2, если написать следующий код:

```
TextWindow.WriteLine("Умножение нечётных чисел на 5:")  
number = 5  
For a = 1 to 10 Step 2  
    TextWindow.WriteLine(a + " x " + number + " = " + a * number)  
EndFor
```



```
Циклы в Small Basic  
Умножение нечётных чисел на 5:  
1 x 5 = 5  
3 x 5 = 15  
5 x 5 = 25  
7 x 5 = 35  
9 x 5 = 45  
Для продолжения нажмите любую клавишу . . .  
_
```

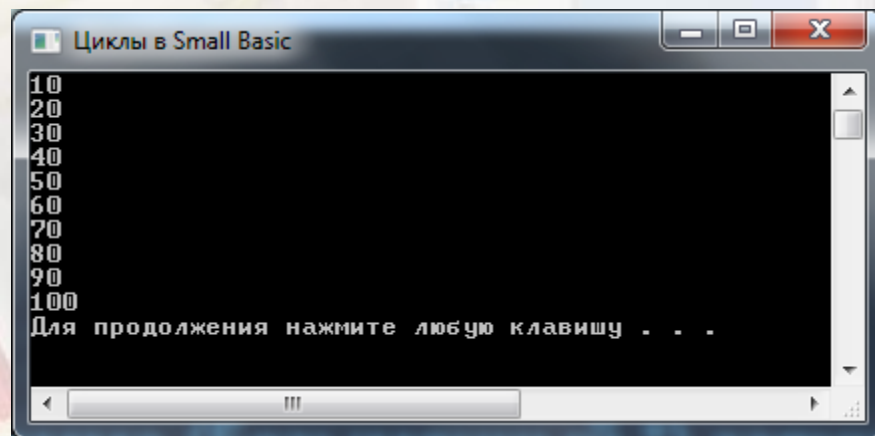
Циклы в программах Small Basic

Если счетчик цикла до написания программы неизвестен, вместо цикла **For** можно создать цикл **While**.

При создании цикла **While** указывается условие, истинное при начале цикла. Но компьютер оценивает условие при каждом повторе цикла. Когда условие становится ложным, выполнение цикла останавливается.

Напишем следующую программу, чтобы продемонстрировать цикл **While**.

```
a = 10
While (a <= 100)
    TextWindow.WriteLine(a)
    a = a + 10
EndWhile
```



```
Циклы в Small Basic
10
20
30
40
50
60
70
80
90
100
Для продолжения нажмите любую клавишу . . .
```



Поздравляем! Вы изучили следующее.

- + Написание программы, оценивающей логические условия и выполняющей операции на основе этих результатов.
- + Написание программ, повторяющих одну или несколько инструкций определенное число раз или на основе логического условия.

Продемонстрируйте свои знания

Создание программы для преобразования одной или нескольких оценок учащихся из процентной в буквенную оценку. Прежде всего, попросите пользователя указать, сколько оценок будет вычислено. Затем попросите пользователя указать первый процент и преобразуйте его в буквенную оценку на основе следующего критерия:

- ❖ Если процент больше 75, он преобразовывается в А.
- ❖ Если процент меньше 75, но больше или равен 60, он преобразовывается в В.
- ❖ Если процент меньше 60, но больше или равен 35, он преобразовывается в С.
- ❖ Если процент меньше 35, он преобразовывается в D.

