

ВВОД И ВЫВОД

1. Ввод данных с помощью функции `InputBox`

Задайтесь вопросом: «как можно задать значение переменной»? Ответ на этот вопрос зависит от некоторых обстоятельств.

Если переменная должна принимать это конкретное значение при каждом запуске проекта, то задание ее значения следует предусмотреть в программном коде. Для этого можно предусмотреть инициализацию переменной при ее объявлении или задание ее значения инструкцией присвоения.

Если же необходимо значение переменной задать на этапе выполнения проекта и при этом иметь возможность установить, какое конкретное значение она должна получить (эта операция называется вводом значения переменной), то можно воспользоваться функцией с именем `InputBox`. Эта функция выводит на экран диалоговое окно, содержащее сообщение и поле ввода, возвращает значение типа `String`, содержащее текст, введенный в поле ввода.

Упрощенный синтаксис обращения к функции:

```
InputBox(Сообщение[, [Заголовок])
```

У этой функции первый слева аргумент обязательный, а все остальные – необязательные. Назначение аргументов:

Обязательный аргумент `Сообщение` – выражение типа `String`, отображаемое в диалоговом окне служит для напоминания пользователю о том, значение какой переменной должно быть задано. Необязательный аргумент `Заголовок` – выражение типа `String`, отображаемое в заголовке диалогового окна.

Пример. На рис. 1 показано окно функции `InputBox` для программного кода:

```
Dim f As String  
f = InputBox("Пожалуйста, введите Ваше имя", «ВводВывод»)
```

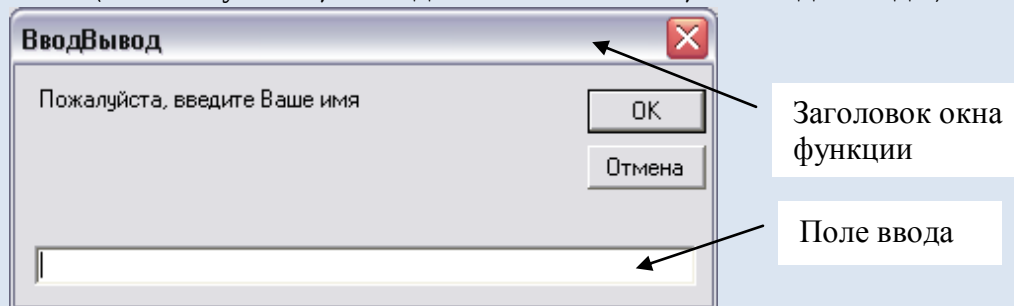


Рис. 1. Окно функции `InputBox`

После ввода пользователем строки в поле ввода и нажатия на кнопку `OK` переменная `f` получит значение, введенное пользователем в поле ввода. Если переменная `f` имеет тип, отличающийся от `String`, то произошло бы преобразование типа `String` строки в поле ввода к типу переменной `f`.

2. Вывод данных

Операция вывод позволяет пользователю увидеть, какое значение имеет переменная в конкретный момент выполнения проекта. Для отображения значений выводимых данных рекомендуется использовать управляющий элемент `TextBox` (текстовое поле).

Ниже приведен код событийной подпрограммы `Form1_Click`, которая выполняется при щелчке на форме `Form1`.

```
Private Sub Form1_Click(ByVal sender As Object, _  
    ByVal e As System.EventArgs) Handles MyBase.Click  
    Dim a As String = "Строка символов"
```

End Sub

[illegible]

Константа vbTab, действует как клавиша табуляции. Она при выводе вставляет поле пробелов между значением переменной a и строкой "b = ".

Константы `vbTab` и `vbCrLf` позволяют управлять расположением выводимых результатов.

Чтобы текстовое поле могло содержать не одну, а несколько строк, его свойство Multiline должно иметь значение True.

Вот упрощенный синтаксис функции `Format: Format(Выражение, Формат)`. Аргумент `Выражение` определяет значение, которое требуется преобразовать. Параметр `Формат` – это текстовая строка, сформированная из символов: (0) – в этой позиции должен быть напечатан замыкающий или лидирующий ноль; (#) – в этой позиции не следует печатать замыкающий или лидирующий ноль; (.), (,), (-), (+), (\$), ((,)), (пробел) – эти символы печатаются в той позиции, где они указаны. Ниже приводятся примеры преобразования чисел дат и времени функцией `Format`:

Обращение к функции	Результат
Format (315.4,"00000.00")	00315,40
Format (315.4,"#####.##")	315,4
Format (6315.4,"##,##0.00")	6 315,40
Format (315.4,"\$##0.00")	\$315,40
Format ("09.10.2001","dd.MM.yy")	09.10.01

Format ("09.10.2001", "dddd dd MMMM yyyy года")	вторник 09 октября 2001 года
Format ("07.12.1958", "M/d/yy")	12/7/58
Format ("07.12.1958", "d-MMM")	7-дек
Format ("07.12.1958", "d-MMMM-yy")	7-декабря-58
Format ("07.12.1958", "dd-MMMM-yy")	07-декабря-58
Format ("07.12.1958", "M/d/yyyy H:mm")	12/7/1958 20:50

В этих примерах регистр символов в параметре Формат важен.

Достоинством применения текстового поля для вывода данных является то обстоятельство, что содержащаяся в нем информация всегда доступна пользователю, пока выполняется проект.

3. Вывод данных с помощью функции MsgBox


Для вывода значения некоторого выражения может быть применена функция MsgBox. Для этого следует обратиться к функции MsgBox, пользуясь, например, упрощенным синтаксисом:

```
MsgBox (Сообщение [, , Заголовок])
```

Здесь Сообщение – это выражение типа строки символов, которое будет выведено в окне MsgBox, а Заголовок – строка символов, отображаемая в заголовке этого окна.

Недостаток этого способа вывода состоит в том, что после закрытия окна MsgBox (а это пользователь вынужден сделать после ознакомления с содержащимся в нем информацией) вся отображавшееся в нем информация становится недоступной.

4. Проект ВводВывод

1. Создайте новый проект с именем ВводВывод, следуя приложению 1.
2. Если окно конструктора форм не открыто, то откройте его щелчком на кнопке  View Designer (просмотреть конструктор), расположенной на панели инструментов окна обозревателя решений. Если же эта кнопка на панели инструментов отсутствует, то предварительно щелкните в окне обозревателя решений на компоненте проекта Form1.vb.
3. Разместите на форме Form1 три управляющих элемента: надпись Label1, текстовое поле TextBox1 и кнопку Button1.
4. Задайте свойству AutoSize надписи значение False.
5. Задайте свойству MultiLine текстового поля значение True. Это позволит отображать в текстовом поле не только одну, но и несколько строк.
6. Установите положение и размеры управляющих элементов, а также значения их свойств Text такими, чтобы форма соответствовала рис. 3.

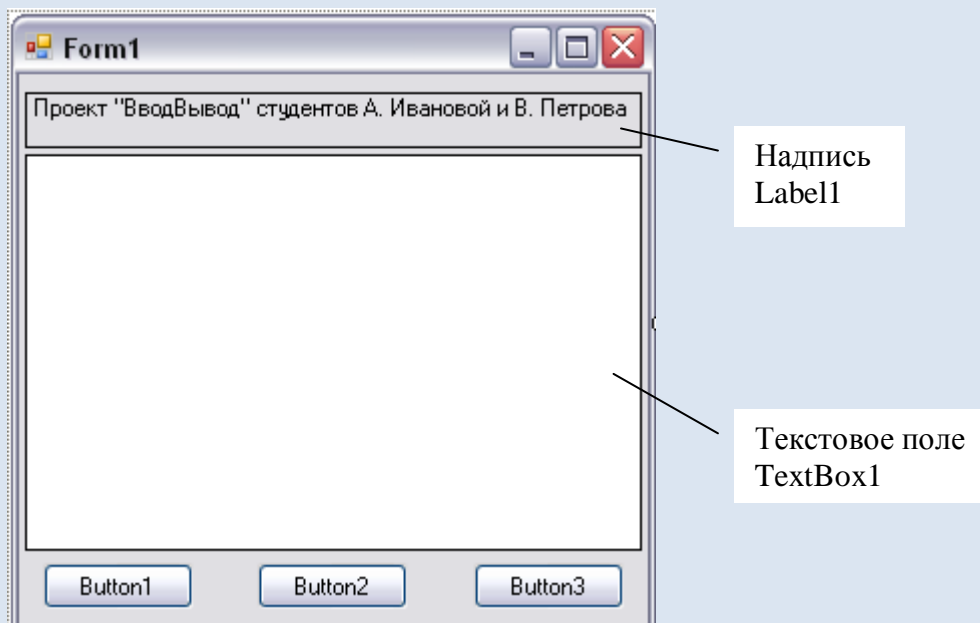


Рис. 3. Положение объектов на форме

7. Выполните двойной щелчок на кнопке Button1. Это приведет к созданию заготовки подпрограммы Button1_Click – обработчика события Click объекта Button1.

8. Дополните код подпрограммы Button1_Click, чтобы он соответствовал листингу 1.

Листинг 1. Код подпрограммы Button1_Click

```
Private Sub Button1_Click(ByVal sender As Object, _
    ByVal e As System.EventArgs) Handles Button1.Click
    Dim i As Integer, s As Single, x As String
    x = InputBox("Пожалуйста, введите Ваше имя и отчество")
    i = InputBox("Задайте значение i")
    s = InputBox("Задайте значение s")
5:   TextBox1.Clear()
    TextBox1.AppendText("x=" & x & vbTab)
    TextBox1.AppendText("i=" & i & vbTab)
    TextBox1.AppendText("s=" & s)
End Sub
```

В подпрограмме Button1_Click объявлены переменная *x* типа строки символов, переменная *i* целого типа и переменная *s* с дробной частью одинарной точности. Этим переменным также задаются значения с помощью обращений к функции InputBox. В строке 5 метод Clear выполняет очистку текстового поля TextBox1. Полученные значения переменными затем в инструкции в строках 6 – 8 отображают в текстовом поле TextBox1.

9. Сохраните проект и выполните его прогон.

10. Щелкните на кнопке Button1. Начнет выполняться подпрограмма Button1_Click. Теперь диалоговое окно функции InputBox последовательно появится на экране три раза.

11. Введите Ваше имя и отчество и щелкните на кнопке ОК. Затем введите произвольное значение переменной *i* и щелкните на кнопке ОК. Наконец, введите произвольное значение (обязательно содержащее дробную часть) переменной *s* и щелкните на кнопке ОК. При вводе значения переменной *s* не забудьте, что для отделения целой части от дробной части следует применять символ запятая (,), если в операционной системе именно она установлена в качестве разделителя целой и дробной частей числа (это типичная настройка). Иначе выполнение программы закончится генерацией исключения с сообщением о

невозможности преобразования типа. Если ввод данных выполнен корректно, то введенные значения будут отображены в одной строке текстового поля TextBox1.

12. Измените код проекта так, чтобы значения каждой переменной при выводе отображалось в отдельной строке. При этом придется применить константы VB vbCrLf и vbTab.

13. Двойным щелчком на кнопке Button2, создайте заготовку событийной подпрограммы Button2_Click и введите ее код (листинг 2).

Листинг 2. Код подпрограммы Button2_Click

```
Private Sub Button2_Click(ByVal sender As System.Object, _  
    ByVal e As System.EventArgs) Handles Button2.Click  
    Dim i As Integer, s As Single, d As Date  
    i = InputBox("Задайте значение i")  
    s = Rnd() 'Функция Rnd возвращает случайное число  
    'из диапазона [0,1], т.е. 0 <= Rnd <1  
    d = InputBox("Задайте дату в формате дд.мм.гг")  
5:    TextBox1.Clear()  
    TextBox1.AppendText("i = " & i & vbCrLf)  
    TextBox1.AppendText("s = " & s & vbCrLf)  
    TextBox1.AppendText("d = " & d & vbCrLf)  
End Sub
```

В строке 5 выполняется очистка текстового окна TextBox1. Инструкции 6 – 8 обеспечивают вывод в отдельных строках значений переменных i, s и d в текстовом поле TextBox1. В инструкциях 6 – 8 каждое завершение добавляемых строк константой vbCrLf делает текущей первую позицию следующей строки.

14. Сохраните проект и начните его выполнение.

15. Щелкните на кнопке Button2. Начнет выполняться процедура Button2_Click.

16. Введите данные и проанализируйте полученные результаты.

17. Остановите выполнение проекта.

18. Примените функцию Format для управления форматом вывода данных. Для этого замените инструкции в 7 и 8 строках подпрограммы Button2_Click:

```
TextBox1.AppendText("s = " & s & vbCrLf)  
TextBox1.AppendText("d = " & d & vbCrLf)
```

на инструкции:

```
TextBox1.AppendText("s = " & _  
    Format(s, "+0.0000000000") & vbCrLf)  
TextBox1.AppendText("d = " & _  
    Format(d, "dddd, dd MMMM, yyyy год") & vbCrLf)
```

19. Сохраните проект и выполните его. Обратите внимание на изменение формата выводимых данных.

20. Еще раз выполните проект. Введите данные. В качестве значения переменной d задайте дату своего рождения. Проанализируйте полученные результаты. Если Вы не знали раньше, в какой день недели Вы родились, то теперь Вам это известно.

21. Остановите выполнение приложения.



22. Сделайте так, чтобы значения переменных i и s выводились не в разных строках, а в одной строке. Для этого замените строку 6 кода

```
TextBox1.AppendText("i = " & i & vbCrLf)
```

следующей строкой

```
TextBox1.AppendText("i = " & i & vbTab)
```

Вместо задаваемого константой vbCrLf перехода к началу следующей строки, константа vbTab задает здесь интервал перед следующим выводом в этой строке.

23. Сохраните проект и выполните его.
 24. Щелкните на кнопке Button2. Введите данные и убедитесь, что теперь значения переменных *i* и *s* выводятся в одной строке.
 25. Остановите выполнение проекта.
 26. Дважды щелкните на кнопке Button3 на Form1 проекта. В окне кода отобразится заготовка подпрограммы Button3_Click.
 27. Скопируйте в эту подпрограмму код инструкций подпрограммы Button2_Click. Для этого выделите все инструкции подпрограммы Button2_Click, кроме первой и последней, щелкните на кнопке  Copy (копировать) стандартной панели инструментов. Затем в коде подпрограммы Button3_Click установите курсор в начало первой после заголовка подпрограммы строки и щелкните на кнопке  Paste (вставить) стандартной панели инструментов.
 28. Замените в подпрограмме Button3_Click три указанные ниже инструкции кода:


```

      TextBox1.AppendText("i = " & i & vbTab)
      TextBox1.AppendText("s = " & _
      Format(s, "+0.0000000000") & vbCrLf)
      TextBox1.AppendText("d = " & _
      Format(d, "dddd, dd MMMM, yyyy год"))
      
```

 на три инструкции:


```

      MsgBox("i = " & i, , "Вывод переменной i")
      MsgBox("s = " & Format(s, "+0.####"), _
      , "Вывод переменной s")
      MsgBox("d = " & Format(d, "dddd dd MMMM yyyy год"), _
      , "Вывод переменной d")
      
```
- Эти инструкции поочередно осуществляют вызов функции MsgBox и отобразят содержание переменных *i*, *s*, *d* в соответствующем окне сообщения.
29. Запустите проект и проанализируйте результаты вывода.
 30. Попробуйте ответить на вопросы для контроля.
 31. Покажите полученный результат преподавателю.
 32. Копируйте рабочую папку на собственную флэш-панель, либо на сетевой диск *o* в папку вашей учебной группы.
 33. Удалите на диске *d* свою рабочую папку.

5. Вопросы для контроля

1. Каково назначение функции InputBox?
 2. Какие символы применяется для отделения целой части числа от дробной части во входном потоке и в программном коде?
 3. Каково назначение метода Clear().?
 4. Каково назначение константы vbCrLf?
 5. Каково назначение константы vbTab?
 6. В чем состоит назначение функции Format?
 7. Каково назначение функции MsgBox?
 8. Какой вариант окна функции InputBox соответствует выполнению приведенной ниже программы:
- ```

Private Sub Form1_Click(ByVal sender As Object, _
ByVal e As System.EventArgs) Handles MyBase.Click
 Dim k As Integer
 k = InputBox("k = ?", "Ввод исходных данных")
End Sub

```



Вариант 1

Вариант 2

9. Какой вариант окна функции MsgBox соответствует результату выполнения приведенной ниже программы:

```
Private Sub Form1_Click(ByVal sender As Object, _
 ByVal e As System.EventArgs) Handles MyBase.Click
 Dim st1, st2 As String, k As Integer
 st1 = 12
 st2 = 15
 k = st1 + st2
 MsgBox("k = " & k, , "Результат")
End Sub
```

Вариант 1

Вариант 2

10. Какой вариант вывода будет получен в результате выполнения приведенной ниже программы:

```
Private Sub Form1_Click(ByVal sender As Object, _
 ByVal e As System.EventArgs) Handles MyBase.Click
 TextBox1.Clear()
 TextBox1.AppendText("Великолепная " & vbCrLf & "идея")
End Sub
```

Вариант 1

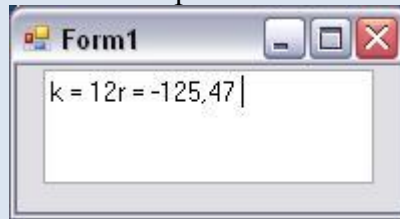
Вариант 2

11. Показаны три варианта вывода в текстовом поле значений переменных k и r:

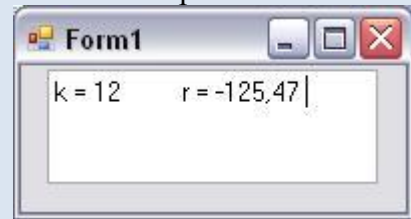
Вариант 1



Вариант 2



Вариант 3



11.1. Какому варианту соответствует результат вывода при применении указанного кода?

```
Private Sub Form1_Click(ByVal sender As Object, _
 ByVal e As System.EventArgs) Handles MyBase.Click
 Dim k As Integer, r As Single
 k = 12
 r = -125.47
 TextBox1.AppendText("k = " & k)
 TextBox1.AppendText("r = " & r)
End Sub
```

11.2. Какому варианту соответствует результат вывода при применении указанного кода?

```
Private Sub Form1_Click(ByVal sender As Object, _
 ByVal e As System.EventArgs) Handles MyBase.Click
 Dim k As Integer, r As Single
 k = 12
 r = -125.47
 TextBox1.AppendText("k = " & k & vbCrLf & "r = " & r)
End Sub
```

11.3. Какому варианту соответствует результат вывода при применении указанного кода?

```
Private Sub Form1_Click(ByVal sender As Object, _
 ByVal e As System.EventArgs) Handles MyBase.Click
 Dim k As Integer, r As Single
 k = 12
 r = -125.47
 TextBox1.AppendText("k = " & k & vbTab)
 TextBox1.AppendText("r = " & r)
End Sub
```

11.4. Какому варианту соответствует результат вывода при применении указанного кода?

```
Private Sub Form1_Click(ByVal sender As Object, _
 ByVal e As System.EventArgs) Handles MyBase.Click
 Dim k As Integer, r As Single
 k = 12
 r = -125.47
 TextBox1.AppendText("k = " & k)
 TextBox1.AppendText(vbTab & "r = " & r)
End Sub
```



11.5. Какому варианту соответствует результат вывода при применении указанного кода?

```
Private Sub Form1_Click(ByVal sender As Object, _
ByVal e As System.EventArgs) Handles MyBase.Click
 Dim k As Integer, r As Single
 k = 12
 r = -125.47
 TextBox1.AppendText("k = " & k & vbCrLf)
 TextBox1.AppendText("r = " & r)
End Sub
```

11.6. Какому варианту соответствует результат вывода при применении указанного кода?

```
Private Sub Form1_Click(ByVal sender As Object, _
ByVal e As System.EventArgs) Handles MyBase.Click
 Dim k As Integer, r As Single
 k = 12
 r = -125.47
 TextBox1.AppendText("k = " & k)
 TextBox1.AppendText(vbCrLf & "r = " & r)
End Sub
```

11.7. Какому варианту соответствует результат вывода при применении указанного кода?

```
Private Sub Form1_Click(ByVal sender As Object, _
ByVal e As System.EventArgs) Handles MyBase.Click
 Dim k As Integer, r As Single
 k = 12
 r = -125.47
 TextBox1.AppendText("k = " & k & vbTab & "r = " & r)
End Sub
```