



## Модификаторы доступа

Последнее обновление: 30.10.2015



Каждый из членов класса, будь то переменная или метод, да и сам класс имеют **модификаторы доступа**. Модификаторы доступа позволяют задать область видимости для членов класса. В VB.NET используются следующие модификаторы:

- **Public** - публичный или открытый класс или член класса. Доступ к члену класса можно получить из любого места в коде, а к открытому классу - из других программ и сборок.
- **Private** - закрытый класс или член класса. Доступ к данному классу или члену класса, можно получить только из кода в том же классе или контексте.
- **Friend** - класс или член класса с данным модификатором доступен из текущей сборки или только в пределах данной программы.
- **Protected** - применяется либо к членам класса, либо к вложенным классам. Такие классы или члены класса только из самого класса, либо из наследующих классов.
- **Protected Friend** - применяется либо к членам класса, либо к вложенным классам. Данный модификатор объединяет контексты Friend и Protected. Поэтому классы и члены класса с данным модификатором доступны либо из текущей программы, либо из наследующих классов.

Объявление переменных с оператором **Dim** аналогично объявлению этих переменных с модификатором **Private**.

Посмотрим на примере использование модификаторов доступа. Создадим следующий класс State:

```
1 Public Class State
2
3     Dim a As Integer 'Все равно что Private a As Integer
4     Private b As Integer 'Доступна только из текущего класса
5     Protected c As Integer 'Доступна из текущего класса и классов-наследников
6     Friend d As Integer 'Доступна в любом месте программы
```

```
7 Protected Friend e As Integer 'Доступна в любом месте программы и из класса
8 Public f As Integer 'Доступна везде
9
10 Private Sub Display_f()
11     Console.WriteLine("Переменная f = {0}", f)
12 End Sub
13
14 Public Sub Display_a()
15     Console.WriteLine("Переменная a = {0}", a)
16 End Sub
17
18 Friend Sub Display_b()
19     Console.WriteLine("Переменная b = {0}", b)
20 End Sub
21
22 Protected Sub Display_e()
23     Console.WriteLine("Переменная e = {0}", e)
24 End Sub
25
26 End Class
```

Поскольку этот класс объявлен как **Public**, он будет доступен из любого места программы. У нас создано пять переменных для каждого уровня доступа и одна переменная с оператором Dim. Также у нас созданы четыре метода, которые будут выводить значения переменных на экран. Обратите внимание, что так как все модификаторы позволяют использовать члены класса внутри данного класса, то и все переменные класса у нас доступны его методам. Теперь посмотрим, как мы сможем использовать переменные нашего класса в программе:

```
1 Module Module1
2
3     Sub Main()
4
5         Dim statel As New State()
6
7         'присвоить значение переменной a у нас не получится,
8         'так как она закрытая и модуль ее не видит
9         'И данную строку среда подчеркнет как неправильную
10        statel.a = 4 Ошибка, получить доступ нельзя
11
12        'то же самое относится и к переменной b
13        statel.b = 3 Ошибка, получить доступ нельзя
14
15        'присвоить значение переменной c то же не получится,
16        'так как модуль не является классом-наследником класса State
17        statel.c = 1 Ошибка, получить доступ нельзя
18
19        'переменная d доступна из любого места программы
20        'поэтому спокойно присваиваем ей значение
```

```
21     statel.d = 5
22
23     'переменная e так же доступна из любого места программы
24     statel.e = 8
25
26     'переменная f общедоступна
27     statel.f = 8
28
29     'Попробуем вывести значения переменных
30     'Так как этот метод объявлен как Private, мы можем использовать его
31     statel.Display_f()      Ошибка, получить доступ нельзя
32
33     'Так как этот метод объявлен как Protected, а модуль не является насл
34     statel.Display_e()      Ошибка, получить доступ нельзя
35
36     'Общедоступный метод
37     statel.Display_a()
38     'Метод доступен из любого места программы
39     statel.Display_b()
40
41     Console.ReadLine()
42 End Sub
43 End Module
```

Таким образом, мы смогли установить только переменные `d`, `e` и `f`. И нам оказались доступны только два метода: `statel.Display_a()` и `statel.Display_b()`. Но так как мы переменные `a` и `b` не смогли установить, то эти методы вывели нули, так как значение переменных типа `Integer` по умолчанию равно нулю.

Несмотря на то, что модификаторы `Friend` и `Public` похожи по своему действию, но они имеют большое отличие. Классы и члены класса с модификатором `Public` также будут доступны и другим программам, если данный класс поместить в динамическую библиотеку `dll` и потом ее использовать в этих программах.

Благодаря такой системе модификаторов мы можем скрыть некоторые моменты реализации класса от других частей программы, такое сокрытие называется **инкапсуляцией**.

[Назад](#) [Содержание](#) [Вперед](#)



ТАКЖЕ НА METANIT.COM

<b>Встроенные компоненты ввода</b>  5 месяцев назад · 1 коммен... Встроенные компоненты ввода Blazor из пространства имен ...	<b>Взаимодействие с кодом Python</b>  5 месяцев назад · 4 коммен... Взаимодействие с кодом Python в программе на языке Си, установка Qt, ...	<b>Отправка запросов на сервер. HttpClient</b>  5 месяцев назад · 1 коммен... Отправка запросов на сервер HttpServer с помощью класса ...	<b>ListVi</b>  2 меся ListVi созда получ
--	---	--	---

0 Комментариев

1 Войти ▼

G

Начать обсуждение...

войти с помощью

или через DISQUS ?

Имя



Поделиться

Лучшие Новые Старые

Прокомментируйте первым.

Подписаться

О защите персональных данных

Помощь сайту

YooMoney:  
410011174743222

Qiwi:  
[qiwi.com/n/METANIT](https://qiwi.com/n/METANIT)

Перевод на карту  
Номер карты:  
4048415020898850

[Вконтакте](#) | [Телеграм](#) | [Twitter](#) | [Помощь сайту](#)

Контакты для связи: [metanit22@mail.ru](mailto:metanit22@mail.ru)

Copyright © metanit.com, 2023. Все права защищены.