



Microsoft Partner
Silver Learning

C# Стартовый

ПРОЦЕДУРНОЕ ПРОГРАММИРОВАНИЕ НА ЯЗЫКЕ C#

Параметры методов



ITVDN
IT VIDEO DEVELOPERS NETWORK

Introduction



Александр Шевчук



MCID: 9230440

Тема урока

Параметры методов

Аргументы (Параметры) методов бывают:

Позиционные

При вызове метода, фактические параметры передаются в соответствии с заданной позицией и типом формальных параметров. Количество фактических параметров должно совпадать с количеством формальных.

Создание метода:

```
void Method(int a, string b)
{
    Console.WriteLine(a + b);
}
```

Вызов метода:

```
Method(5, "Hello");
```

Именованные

При вызове метода, фактические параметры передаются в соответствии с именем и типом формальных параметров. Позиция фактических параметров может не совпадать с позицией формальных, но количество фактических параметров должно совпадать с количеством формальных параметров.

Создание метода:

```
void Method(int a, string b)
{
    Console.WriteLine(a + b);
}
```

Вызов метода:

```
Method(b: "Hello", a: 5);
```

Опциональные (необязательные)

При создании метода, формальным параметрам можно присваивать значения по умолчанию. При вызове метода, фактические параметры можно передавать так, как передаются позиционные или именованные. Допускается не передавать формальным параметрам никаких фактических параметров, так как формальные параметры уже имеют значение по умолчанию.

Создание метода:

```
void Method(int a = 1, string b = "Привет")
{
    Console.WriteLine(a + b);
}
```

Вызов метода:

```
Method (b : "Hello");
Method(5);
Method();
```

ПРОЦЕДУРНОЕ ПРОГРАММИРОВАНИЕ НА ЯЗЫКЕ C#



Используйте именованные параметры

Некоторые языки позволяют явно сопоставить формальные параметры с фактическими. Это делает применение параметров более ясным и помогает избегать ошибок, обусловленных неправильным сопоставлением параметров...

Стр. 175

Данный подход (с использованием именованных параметров) особенно полезен при использовании длинных списков параметров одинакового типа, потому что в этом случае вероятность неправильного сопоставления параметров более высока, а компилятор эту ошибку определить не может. Во многих средах явное сопоставление параметров может оказаться пальбой из пушки по воробьям, но в средах, от которых зависит безопасность людей, или других средах с повышенными требованиями к надежности дополнительный способ гарантии правильного сопоставления параметров не помешает.

Стр. 176

Аргументы (Параметры) методов бывают:

ref

- ref-аргумент передается по ссылке, а не по значению.

Создание метода:

```
void Method(ref int parameter)
{
    parameter++;
    Console.WriteLine(parameter);
}
```

Вызов метода:

```
int variable = 0;
Method(ref variable);
```

in

- in-аргумент передается по ссылке, а не по значению.
- in-аргументы не могут быть изменены вызываемым методом.

Создание метода:

```
void Method(in int parameter)
{
    // parameter++;
    Console.WriteLine(parameter);
}
```

Вызов метода:

```
int variable = 0;
Method(in variable);
```

out

- out-аргумент передается по ссылке, а не по значению.
- out-аргумент не требуется инициализировать перед передачей в метод, но требуется инициализировать в теле метода.

Создание метода:

```
void Method(out int parameter)
{
    parameter = 7;
    parameter++;
    Console.WriteLine(parameter);
}
```

Вызов метода:

```
int variable;
Method(out variable);
```

Ключевые слова **ref**, **in** и **out** не считаются частью сигнатуры метода

ПРОЦЕДУРНОЕ ПРОГРАММИРОВАНИЕ НА ЯЗЫКЕ C#

Спасибо за внимание! До новых встреч!



Александр Шевчук



OLEKSANDR SHEVCHUK

Has successfully completed the requirements to be recognized as a Trainer.

Date of achievement: October 25, 2012
Certification number: E207-8382
Valid until: April 04, 2019

Satya Nadella
Chief Executive Officer

Microsoft
CERTIFIED
Trainer

MCID: 9230440

Информационный видеоресурс для разработчиков программного обеспечения

