

PRA1 PROGRAMAÇÃO PARA APLICATIVOS 1

SÉRIE 1

PROFESSOR MARCOS ANTONIO DOS SANTOS

SEQUENCIA 003

DATA 10.08.2020

/**

*

* Creative Commons (CC) 2019 Marcos Vinícius da Silva Santos and Marcos Antonio dos Santos

*

* Licensed under the Creative Commons, Version 4.0;

* you may not use this file except in compliance with the License.

* You may obtain a copy of the License at

*

* <https://creativecommons.org/licenses/by-nc-sa/4.0/>

* <https://creativecommons.org/licenses/by-nc-sa/4.0/legalcode>

* Attribution-NonCommercial-ShareAlike 4.0 International (CC BY-NC-SA 4.0)

*

* https://creativecommons.org/licenses/by-nc-sa/4.0/deed.pt_BR

* <https://creativecommons.org/licenses/by-nc-sa/4.0/legalcode.pt>

* Atribuição-NãoComercial-CompartilhaIgual 4.0 Internacional (CC BY-NC-SA 4.0)

*

* Unless required by applicable law or agreed to in writing, software

* distributed under the License is distributed on an "AS IS" BASIS,

* WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.

* See the License for the specific language governing permissions and

* limitations under the License.

*

*

*/

ARGUMENTO VS PARAMETROS

Baseado em um post de @java_eduyear

Parâmetro é a variável que irá receber um valor em uma função (ou método) enquanto que um **argumento** é o valor (que pode originar de uma variável ou expressão) que você passa para a função (ou método).

Você não passa parâmetros, você passa argumentos. Você recebe argumentos também, mas recebe em forma de parâmetros. Você parametriza sua função com informações que virão posteriormente. Você argumenta com o que deseja executar uma função devidamente parametrizada.

Pode haver menos ou mais argumentos para cada parâmetro já que existem parâmetros que são opcionais e outros que podem ser listas variáveis de dados. Portanto não há uma relação de um para um e a distinção entre eles é importante.

Frequentemente é confundido por todos e eu mesmo admito que faço uso indevido destes termos erroneamente, mas para uma boa comunicação é importante todos saberem o correto.

Argumento são os valores que passamos ao chamar a função, enquanto parâmetros são as variáveis que definimos ao definir uma função.

```
void Func1(int i, bool flag = true) { //declarou dois parâmetros. O inteiro i sem valor definido e o booleano flag com valor definido com true
```

```
    // execução do código
}
```

```
void Func2( params int[] lista ) { //declarou um parâmetro como sendo uma lista de números inteiros

    // execução do código
}
```

```
void Func3( bool x, bool y ) {

    int z = 0;

    Func1(1); //chamou a função com 1 argumento

    Func1(z, x && y); //chamou a função com dois argumentos vindos da variável z e do resultado da expressão x && y

    Func2(1, 2, 3); //chamou a função com 3 argumentos

    Func1(2, true);

}
```

A função Func3 tem quantos parâmetros declarados?

Qual é o tipo de dado destes parâmetros?

Poderíamos utilizar a Func2 informando 5 argumentos, por exemplo Func2(13, 12, 68, 51, 3)?

Poderíamos utilizar a Func1 informando seus dois argumentos assim Func1('m', "true")?

parâmetros

/ \

```
function add(x, y) {
  return x + y;
}
```

sum = add(4, 1);

/ \

argumentos

Por analogia, podemos considerar o parâmetro como um espaço de estacionamento e o argumento como um automóvel.

Portanto ...

Parâmetros

São definidos na declaração (ou assinatura) da função/método;

Os parâmetros de uma função recebem os argumentos;

O nome de cada parâmetro serve como uma variável local dentro da função;

Uma boa mnemónica é pensar que um Parâmetro é como um Placeholder para um Potencial valor.

Argumentos

Representam os valores/variáveis atuais passadas para os parâmetros da função, quando é invocada;

Cada argumento corresponde a um parâmetro (na mesma posição);

Uma boa mnemónica é pensar que um Argumento é o valor Atual.

Inverter uma string