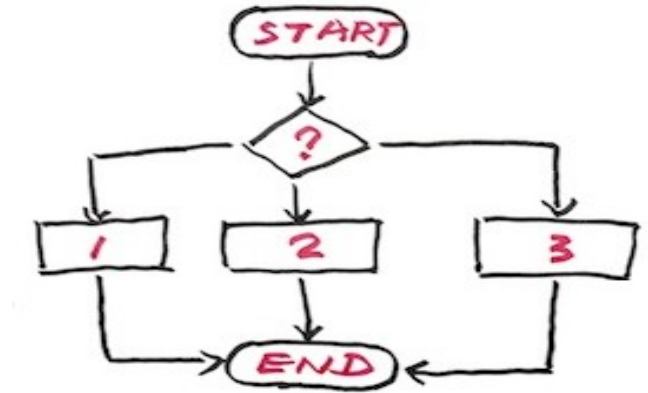


Introdução à Programação

Aula 15 *Módulos*



Aula anterior

Funções

Aula de hoje

- Escopo
- Módulos

Escopo local

Definição: Variável

- Uma variável criada dentro de uma função pertence ao **escopo local** dessa função e só pode ser usada dentro dessa função.

A variable created inside a function is available inside that function:

```
def myfunc():  
    x = 300  
    print(x)  
  
myfunc()
```

Escopo local

Função dentro da função

- A variável `x` não está disponível fora da função, mas está disponível para qualquer função dentro da função:

The local variable can be accessed from a function within the function:

```
def myfunc():  
    x = 300  
    def myinnerfunc():  
        print(x)  
    myinnerfunc()  
  
myfunc()
```

Escopo Global

Definição

- Uma variável criada no **corpo principal** do código Python é uma **variável global** e pertence ao escopo global.
- Variáveis globais estão disponíveis em **qualquer escopo**, global e local.

```
x = 300

def myfunc():
    print(x)

myfunc()

print(x)
```

Escopo Global

Variáveis

- Se declarar o mesmo nome de variável dentro e fora de uma função, o Python as tratará como duas variáveis separadas, uma disponível no escopo global (fora da função) e outra disponível no escopo local (dentro da função):

```
x = 300

def myfunc():
    x = 200
    print(x)

myfunc()

print(x)
```

Escopo Global

Global Keyword

- Se precisar **criar** uma **variável global**, mas estiver preso no escopo local, poderá usar a palavra-chave global.
- A palavra-chave global torna a variável global.

```
def myfunc():  
    global x  
    x = 300  
  
myfunc()  
  
print(x)
```


SeuNome_Aula15.py

Ex01: Escreva uma função que receba um número como parâmetro e multiplique-o por uma variável global. Em seguida, exiba o resultado fora da função.

Ex02: Escreva uma função que solicite ao usuário um número e armazene-o em uma variável local. Em seguida, exiba o valor dessa variável local.

Ex03: Escreva uma função que receba uma lista como parâmetro e modifique um elemento específico dessa lista usando uma variável global. Em seguida, exiba a lista modificada fora da função.

Ex04: Escreva uma função que solicite ao usuário um nome e armazene-o em uma variável local. Em seguida, crie uma função secundária que exiba esse nome, sem passá-lo como parâmetro.

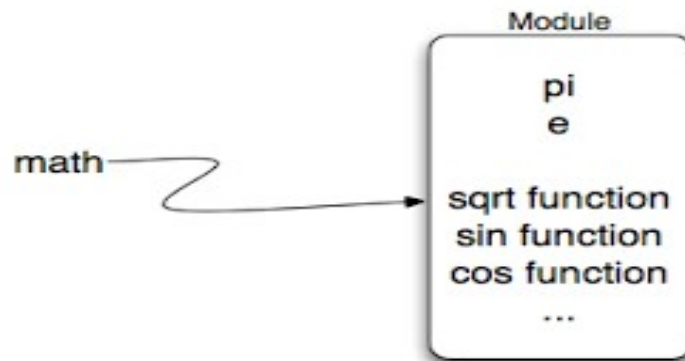
Ex05: Escreva uma função que declare uma variável local e uma variável global com o mesmo nome. Em seguida, exiba o valor de ambas as variáveis.

Módulos

Módulos

O que é uma módulo?

- Considere um módulo igual a uma biblioteca de códigos.
- Um arquivo que contém um conjunto de funções que você deseja incluir no seu aplicativo.



Módulos

Criar um módulo

- Para criar um módulo, salve o código desejado em um arquivo com a extensão .py:

Exemplo:

Save this code in a file named `mymodule.py`

```
def greeting(name):  
    print("Hello, " + name)
```

Módulos

Criando um módulo

Save this code in a file named `mymodule.py`

```
def greeting(name):  
    print("Hello, " + name)
```

Usando o módulo

Import the module named mymodule, and call the greeting function:

```
import mymodule  
  
mymodule.greeting("Jonathan")
```

Módulos

Variáveis em módulos

- O módulo pode conter também variáveis de todos os tipos (matrizes, dicionários, objetos etc.):

Save this code in the file `mymodule.py`

```
person1 = {  
    "name": "John",  
    "age": 36,  
    "country": "Norway"  
}
```

Import the module named mymodule, and access the person1 dictionary:

```
import mymodule  
  
a = mymodule.person1["age"]  
print(a)
```

Módulos

Nomeando um módulo

- Você pode nomear o arquivo do módulo como desejar, mas ele deve ter a extensão .py

Create an alias for `mymodule` called `mx`:

```
import mymodule as mx

a = mx.person1["age"]
print(a)
```

Lista de exercício 05

SeuNome_Aula15.py

Ex06: Crie um módulo chamado "math_utils" que contenha uma função chamada "dobro" que receba um número como parâmetro e retorne o dobro desse número. Em seguida, importe esse módulo em um programa principal e utilize a função para calcular o dobro de um número fornecido pelo usuário.

Ex07: Crie um módulo chamado "string_utils" que contenha uma função chamada "inverter" que receba uma string como parâmetro e retorne a string invertida. Em seguida, importe esse módulo em um programa principal e utilize a função para inverter uma frase fornecida pelo usuário.

Ex08: Crie um módulo chamado "counter_utils" que contenha uma variável global chamada "counter" e duas funções: "incrementar" e "exibir_contagem". A função "incrementar" deve incrementar o valor da variável "counter" em 1, enquanto a função "exibir_contagem" deve exibir o valor atual da variável "counter". Importe esse módulo em um programa principal e utilize as funções para incrementar e exibir a contagem.

SeuNome_Aula15.py

Ex09: Crie um módulo chamado "random_utils" que contenha uma função chamada "gerar_numero_aleatorio" que retorne um número aleatório entre dois valores fornecidos como parâmetros. Importe esse módulo em um programa principal e utilize a função para gerar um número aleatório dentro de um intervalo definido pelo usuário.

Ex10: Crie um módulo chamado "data_utils" que contenha uma variável global chamada "data_atual" e uma função chamada "exibir_data_atual" que exiba o valor da variável "data_atual". Importe esse módulo em um programa principal e utilize a função para exibir a data atual.