

Introdução à Programação

01 {

[Funções e
Subprogramas]



[SubProgramas - Conceito]

- Subprogramas são trechos de programa que realizam uma tarefa específica;
- Podem ser chamados pelo nome a partir do programa principal ou de trechos de outros subprogramas, até mesmo ele próprio (chamada recursiva).



[Tipos]

- **Funções (functions)**

- Retornam um valor em seu nome

- **Procedimentos (procedures)**

- Não retornam valor



[SubProgramas em Python]

“uma função é uma sequência nomeada de instruções que executa uma operação de computação. Ao definir uma função, você especifica o nome e a sequência de instruções. Depois, pode “chamar” a função pelo nome”



[Chamando uma Função]

Função

```
print('Eu sou Wulldson!')
```

Nome da Função

Argumento



[Chamando uma Função]

- Nome da função é print;
- A String entre parênteses é o argumento;
- Resultado desta função impressão do argumento no terminal.



[Exemplo]

```
1
2
3
4 type('64')
5
6 int(3.15)
7
8 float(32)
9
10 str(3.14)
11
12 input('Digite seu nome')
13
14 print('Star Wars Day')
```

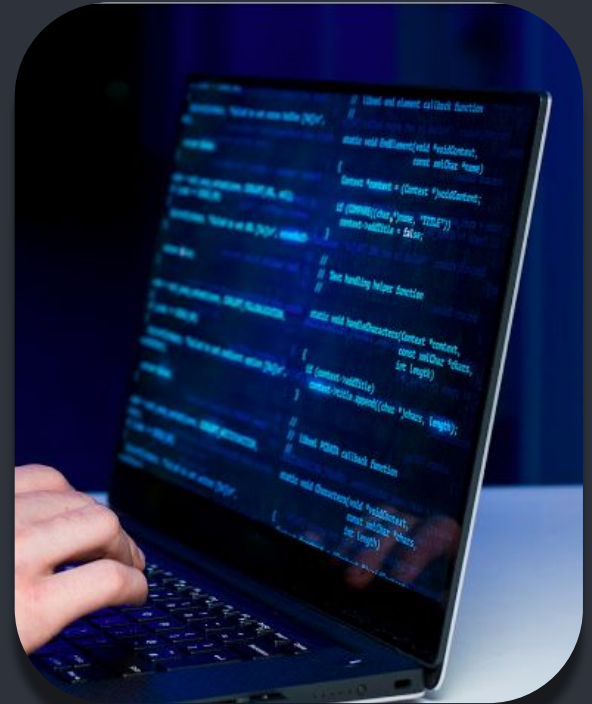


02 {

[Importação de

(Funções)

Bibliotecas]





[Importação de (Funções) Bibliotecas]

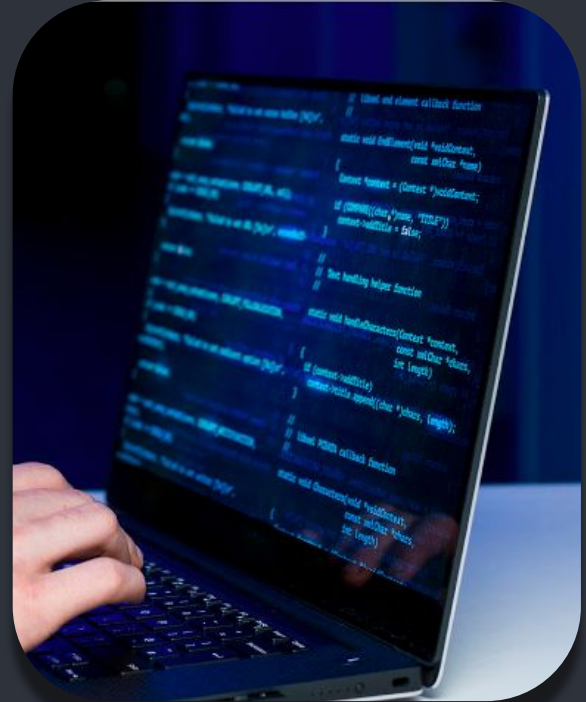
```
1
2
3
4     import random
5
6     import math
7
8     import re
9
10
11     import datetime
12
13
14
```

[Exemplos]

```
1  
2  
3  
4  
5  
6 from random import randint  
7  
8 from math import pi  
9  
10  
11  
12  
13  
14
```

03 {

[Criação de Funções]



[Criação de Funções]

```
1  
2  
3  
4  
5  
6  
7 def soma(parametro1, parametro2):  
8     return(parametro1 + parametro2)  
9  
10  
11  
12  
13  
14
```

[Parâmetros e Argumentos]

- Funções podem necessitar de argumentos;
- Argumentos são atribuídos a variáveis;
- Essas variáveis são chamadas de parâmetros;
 - Argumento é o valor
 - Parâmetro é a variável



**Importante lembrar..
Variáveis e parâmetros são locais**

[Tipos de Parâmetros – Formais]

- São aqueles passados na declaração da função;
- É onde informamos quais as variáveis que a função irá receber quando chamada e quais os seus tipos (são informados como uma declaração da variável;
- Esses parâmetros são considerados como variáveis locais da função;
- Se a função não precisa receber nenhum parâmetro, colocamos entre os parênteses a palavra void;

[Tipos de Parâmetros – Reais]

- São aqueles passados na chamada da função;
- É quando informamos quais os valores que os parâmetros terão dentro da função;
- Se a função não espera receber nenhum parâmetro, na sua chamada colocamos o abre e fecha parênteses vazio.

04 {

[Por que
Utilizar
SubProgramas?]

- 1
- 2
- 3 • Criar uma nova função dá a oportunidade de nomear um grupo de
- 4 instruções, o que deixa o seu programa mais fácil de ler e de
- 5 depurar;
- 6 • As funções podem tornar um programa menor, eliminando o código
- 7 repetitivo. Depois, se fizer alguma alteração, basta fazê-la em
- 8 um lugar só;
- 9 • Dividir um programa longo em funções permite depurar as partes
- 10 uma de cada vez e então reuni-las em um conjunto funcional;
- 11 • As funções bem projetadas muitas vezes são úteis para muitos
- 12 programas. Uma vez que escreva e depure uma, você pode
- 13 reutilizá-la.
- 14



uniesp

Centro Universitário