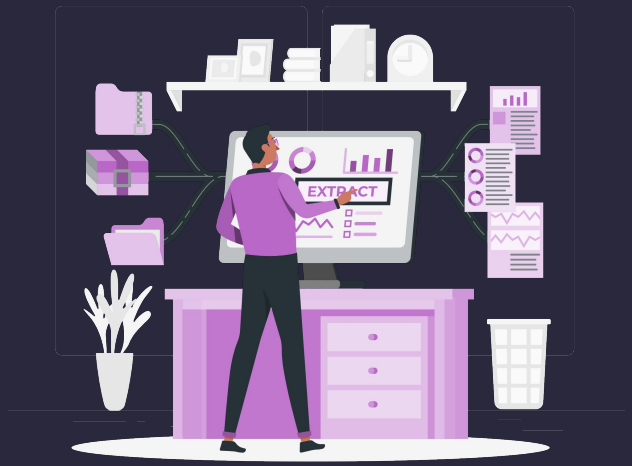
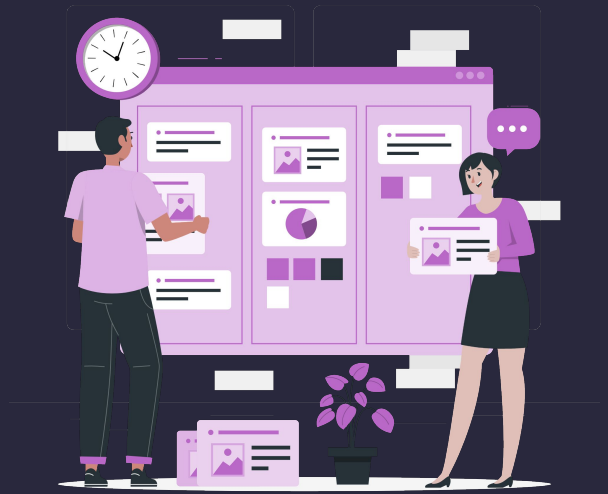
SUBPROGRAMAS: FUNCTIONS AND PROCEDURES



Introdução a Programação

/01

/SUBPROGRAMAS



/CONCEITO

Definição

Bloco separado de instruções que pode ser "chamado" pelo nome;

Tipos

- **Funções**: retorna um valor;
- **Procedimentos**: Realizam uma ação, não retornam valor.

VANTAGENS DOS SUBPROGRAMAS

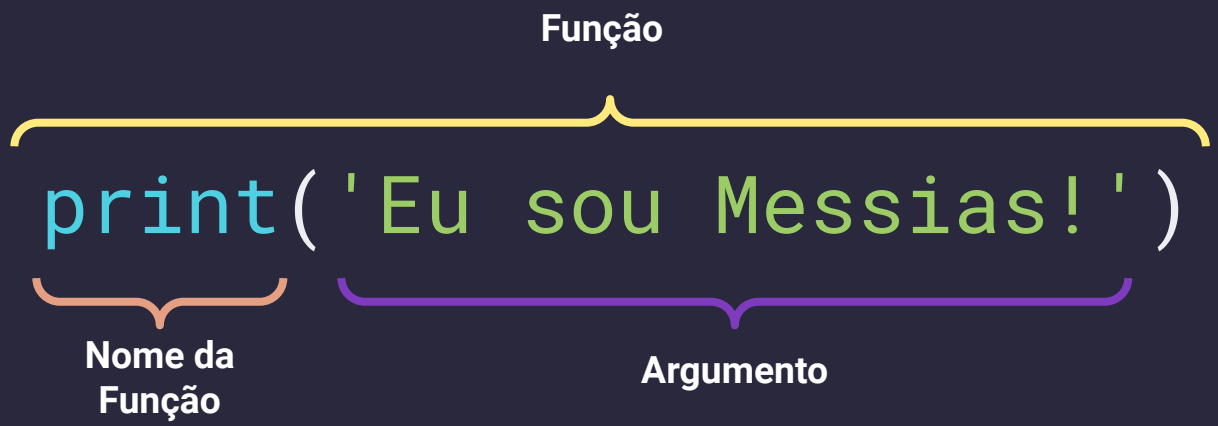
1. **Reutilização de Código:** Evita repetição.
2. **Modularidade:** Torna o código mais gerenciável.
3. **Abstração:** Pensamento em ações de alto nível.
4. **Colaboração:** Permite trabalho simultâneo em diferentes partes.

EXEMPLO 1 - FUNÇÃO SOMAR

```
# Isso é uma função chamada 'somar'
def somar(a, b):
    return a + b

# Usando a função
resultado = somar(5, 3)
print(resultado)  # Isso imprimirá 8
```

EXEMPLO 2 - CHAMANDO A FUNÇÃO PRINT

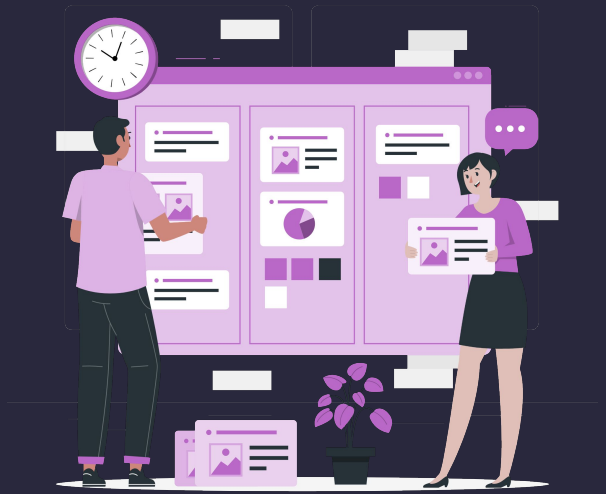


EXEMPLO 3 - OUTRAS FUNÇÕES

```
type('64')
int(3.15)
float(32)
str(3.14)
input('Digite seu nome')
print('Vascão')
```

02

IMPORTAÇÃO DE FUNÇÕES BIBLIOTECAS



IMPORTAÇÃO DE BIBLIOTECAS

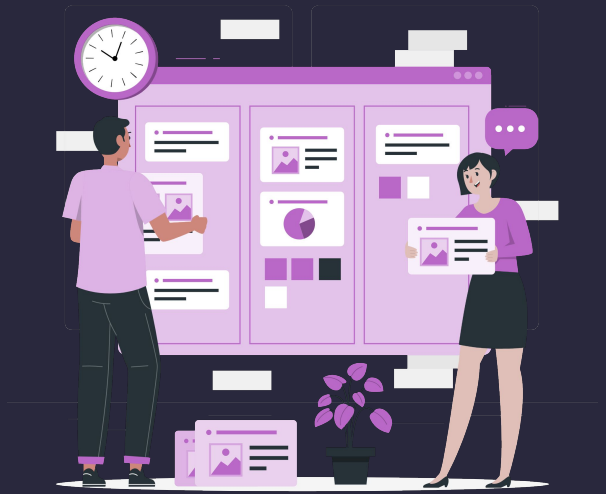
```
import random
import math
import re
import datetime
```

IMPORTAÇÃO DE FUNÇÕES DE BIBLIOTECAS

```
from random import randint
from math import pi
from bs4 import BeatifulSoup
from urllib.request import urlopen
```

03

CRIAÇÃO DE FUNÇÕES



/CRIANDO UMA FUNÇÃO DE SOMA

```
def NOME (PARAMETROS):
  COMANDOS
```

EXEMPLO 1 - FUNÇÃO SOMAR

```
# Isso é uma função chamada 'somar'
def somar(a, b):
    return a + b
```

ESCOPO DE VARIÁVEL

Refere-se à região do código onde essa variável pode ser acessada ou modificada.

Tipos:

- **Locais:** quando visíveis apenas dentro do subprograma;
- **Globais:** visível em todo o programa.

VARIÁVEL LOCAL

- Definidas dentro de uma função.
- Só podem ser acessadas dentro dessa função.
- Existem apenas enquanto a função está sendo executada.

VARIÁVEL LOCAL

```
def minha_funcao():
    variavel_local = "Eu sou local!"
    print(variavel_local)

minha_funcao()  # Imprime "Eu sou local!"
```

VARIÁVEL GLOBAL

- Definidas fora de todas as funções.
- Podem ser acessadas globalmente no arquivo/script.
- Existem enquanto o programa estiver em execução.

VARIÁVEL GLOBAL

```
variavel_global = "Eu sou global!"


def minha_funcao():
    print(variavel_global)


minha_funcao()  # Imprime "Eu sou global!"
```

/PARÂMETROS E ARGUMENTOS

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

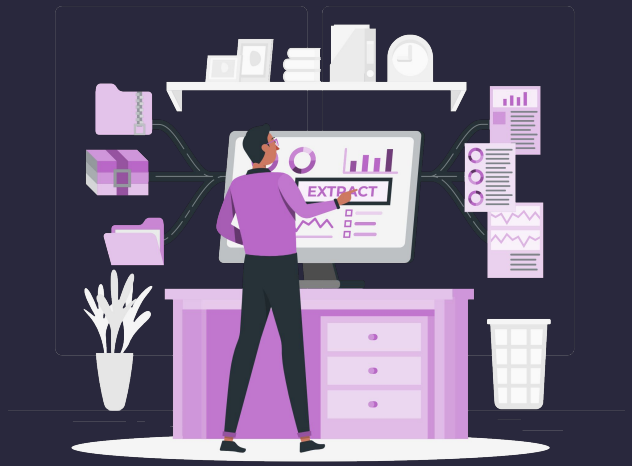
Importante lembrar..
Variáveis e parâmetros são locais

EXEMPLO 1 - FUNÇÃO SOMAR

```
# Isso é uma função chamada 'somar'
def somar(a, b):
    return a + b

# Usando a função
resultado = somar(5, 3)
print(resultado)  # Isso imprimirá 8
```

SUBPROGRAMAS: FUNCTIONS AND PROCEDURES



Introdução a Programação