

Módulo 03 | Python: Fluxo Condicional & Repetição

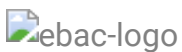
Caderno de **Aula**

Desenvolvedor [Enzo Schitini](#)

▼ Tópicos

1. Estrutura condicional if / else / elif;
 2. Estrutura condicional try / catch / finally;
 3. Estrutura de repetição for / in.
-

▼ Aulas



▼ 1. Estrutura condicional if / else / elif

▼ 1.1. if / else

Estrutura de alteração de fluxo lógico do código, avalia um valor booleano ou uma comparação lógica. **Note** a indentação do código.

```
if <booleano / comparação lógica> == True:
    <execute este código>
else:
    <senão execute este código>
```

```

if (codigo_de_seguranca == codigo_de_seguranca_cadastro) & (senha == senha_cadastro):
    print("Pagamento efetuado")
else:
    print("Erro: Pagamento não efetuado")

```

Pagamento efetuado

```

if (codigo_de_seguranca != codigo_de_seguranca_cadastro) | (senha != senha_cadastro):
    print("Erro: Pagamento não efetuado")
else:
    print("Pagamento efetuado")

```

Pagamento efetuado

▼ 1.2. if / elif / else

Podemos também avaliar múltipla condições.

```

if <1º booleano / 1ª comparação lógica> == True:
    <execute este código se a primeira condição for verdade>
elif <2º booleano / 2ª comparação lógica> == True:
    <execute este código se a segunda condição for verdade>
else:
    <senão execute este código>

```

```

codigo_de_seguranca = '802'
codigo_de_seguranca_cadastro = '852'

```

```

senha = '7703'
senha_cadastro = '7783'

```

CÓDIGO	SENHA	CÓDIGO AND SENHA	MENSAGEM
TRUE	TRUE	TRUE	Pagamento efetuado
TRUE	FALSE	FALSE	Erro: Senha inválida
FALSE	FALSE	FALSE	Erro: Código de segurança e senha inválidos
FALSE	TRUE	FALSE	Erro: Código de segurança inválido

```

if (codigo_de_seguranca == codigo_de_seguranca_cadastro) & (senha == senha_cadastro):
    print("Pagamento efetuado")

```

```

elif (codigo_de_seguranca != codigo_de_seguranca_cadastro) & (senha == senha_cadastro):
    print("Erro: Código de segurança inválido")

```

```

elif (codigo_de_seguranca == codigo_de_seguranca_cadastro) & (senha != senha_cadastro):
    print("Erro: Senha inválida inválida")

```

```
ano_atual = anos[3]
```

```
-----  
IndexError                                Traceback (most recent call last)  
c:\Users\Intel i5\Downloads\Python\Modulos\module_03.ipynb Cella 39 line 1  
----> <a href='vscode-notebook-  
cell:/c%3A/Users/Intel%20i5/Downloads/Python/Modulos/module_03.ipynb#X53sZmlsZQ%3D%3  
line=0'>1</a> ano_atual = anos[3]
```

IndexError: list index out of range

```
cursos = {  
    'python': {  
        'nome': 'Python para Análise de Dados', 'duracao': 2.5  
    },  
    'sql': {  
        'nome': 'SQL para Análise de Dados', 'duracao': 2  
    }  
}
```

```
curso_atual = cursos['analista']  
  
{'nome': 'Python para Análise de Dados', 'duracao': 2.5}
```

▼ 2.2. try / except

Estrutura para tratar exceções:

```
preco = 132.85  
pessoas = 2  
  
try:  
    valor_por_pessoa = preco / pessoas  
    print(valor_por_pessoa)  
except ZeroDivisionError:  
    print('Número de pessoas inválido. Espera-se um valor maior que 0 e obteve-se um valor :')
```

```
anos = [2019, 2020, 2021]  
  
try:  
    ano_atual = anos[3]  
    print(ano_atual)  
except Exception:  
    print('Lista de anos é menor que o valor escolhido. Espera-se um valor entre 0 e ' + str(len(anos)))
```

Lista de anos é menor que o valor escolhido. Espera-se um valor entre 0 e 2

```
anos = [2019, 2020, 2021]
```

▼ 3. Estrutura repetição for / in

▼ 3.1. for / in

Estrutura que permite a execução repetida de um bloco de código repetidas vezes.

```
for variavel_temporaria in coleção:  
    <execute este código>
```

▼ 3.2. for / in / range

Estrutura que permite a execução repetida de um bloco de código **n** vezes.

```
for valor in range(6):  
    print(valor)
```

```
soma = 0
```

```
for valor in range(0, 100000):  
    soma = soma + valor  
    # print(soma)
```

```
print(soma)
```

```
for multiplo_dois in range(2, 10, 3):  
    print(multiplo_dois)
```

▼ 3.3. for / in / list

Estrutura que permite a execução de um bloco de código para todos os elementos de uma lista.

```
frutas = ['maca', 'banana', 'laranja', 'uva', 'pera']
```

```
for fruta in frutas:  
    print(fruta)
```

```
frase = 'Fala pessoal, meu nome é André Perez.'
```

```
for caracter in frase:
```

750

O valor do escore de crédito é 750, mas não temos mais as chaves :(.

812

O valor do escore de crédito é 812, mas não temos mais as chaves :(.

980

O valor do escore de crédito é 980, mas não temos mais as chaves :(.

▼ 3.5. break / continue

Estrutura que permite a quebra ou o avanço de um laço de repetição.

```
for i in range(0, 10*10*10*10*10*10):
    print(i)
    if i == 10:
        break
```

```
numero = 3
```

```
if numero % 2 == 0:
    print(f'O numero {numero} é par')
else:
    print(f'O numero {numero} é impar')
```

```
numeros = [361, 553, 194, 13, 510, 33, 135]
```

```
for numero in numeros:

    if numero % 2 == 0:
        print(f'O numero {numero} é par')
        break
    else:
        print(f'O numero {numero} é impar')
```

```
numeros = [361, 553, 194, 13, 510, 33, 135]
```

```
for numero in numeros:

    if numero % 2 == 0:
        print(f'O numero {numero} é par')
        break
    else:
        continue
    print(f'O numero {numero} é impar')
```