

Python

Curso Intensivo

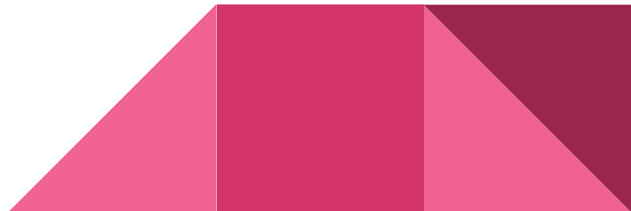
Prof. Cláudio Fleury
Abr-22

Curso baseado no
livro de Eric Mattes,
**Python Crash
Course**

Cap.5 – if's
e Listas

Conteúdo

1. Valor específico de uma lista
2. Testes Condicionais
3. Sentenças **if**
4. Sentenças **if** com Listas
5. Sentenças **if** com Abrangência de Listas
6. Resumo



Dividindo o Conteúdo de uma String em Palavras

- ◆ Nomes de fabricantes de carros são nomes próprios, então os nomes da maioria deles deve começar com letra maiúscula. No entanto, o nome 'bmw' deve ser impresso com todas as letras em maiúsculas
- ◆ A sigla BMW significa *Bayerische Motoren Werke* (Fábrica Bávara de Motores)

File Edit Selection Find View Goto Tools Project Preferences Help

```
FOLD  ▶  platypus_doc.py  x  dimensoes2.py  x  carros3.py  x
1  fabr_carros = ['bmw', 'audi', 'toyota', 'subaru']
2  for fabr in fabr_carros:
3      if fabr == 'bmw':
4          print(fabr.upper())
5      else:
6          print(fabr.title())
```

```
BMW
Audi
Toyota
Subaru
[Finished in 178ms]
```

Testes Condicionais

- No centro de cada instrução **if** está uma expressão que pode ser avaliada como **True** (Verdadeiro) ou **False** (Falso) e é chamada de teste condicional

```
>>> fabr = 'bmw'  
>>> fabr == 'bmw'  
True  
>>>
```

```
>>> fabr = 'bmw'  
>>> fabr == 'audi'  
False  
>>>
```

```
>>> fabr = 'Bmw'  
>>> fabr == 'bmw'  
False  
>>>
```

```
>>> fabr = 'Bmw'  
>>> fabr.lower() == 'bmw'  
True  
>>>
```



Testes Condicionais

Idênticos aos operadores relacionais da Linguagem C

- Operadores de comparação: **==, !=, >, <, >=, <=**

```
>>> idade = 30
>>> idade == 30
True
>>>
```

```
>>> idade = 16
>>> idade != 30
True
>>>
```

```
>>> idade = 41
>>> idade <= 40
False
>>>
```

```
>>> idade = 27
>>> idade >= 18
True
>>>
```



Testes Condicionais

- Operadores lógicos: **and, or, not**

```
>>> idade = 30
>>> idade >= 18 and idade <= 50
True
>>>
```

```
>>> id_1 = 16
>>> id_2 = 20
>>> id_1 != 30 or id_2 < 10
True
>>>
```



Testes Condicionais

- Operador de pertinência: **in**

```
>>> fabr = ['bmw', 'audi', 'fiat']
>>> 'ford' in fabr
False
>>> 'fiat' in fabr
True
>>>
```

File Edit Selection Find View Goto Tools Project Preferences Help

```
carros3.py x lista_negra.py x
1 lista_negra = ['andrew', 'carolina', 'david'] # clientes devedores
2
3 usuario = 'maria' # usuário do pedido
4 if usuario not in lista_negra:
5     print(f"{usuario.title()}, você pode pagar seu pedido com cartão de crédito...")
```

```
Maria, você pode pagar seu pedido com cartão de crédito...
[Finished in 382ms]
```

Exercícios

1. Testes condicionais: escreva uma série de testes condicionais. Imprimir uma declaração descrevendo cada teste e sua previsão para os resultados de cada teste. Seu código deve ficar algo assim:

```
carro = 'subaru'
print('Carro é 'subaru'? Eu prevejo True.')
print(carro == 'subaru')
print('Carro é 'audi'? Eu prevejo False.')
print(carro == 'audi')
```

- Observe atentamente seus resultados e certifique-se de entender por que cada linha avalia como Verdadeiro ou Falso.
- Crie pelo menos dez testes. Tenha pelo menos cinco testes avaliados como Verdadeiro e outros cinco testes avaliados como Falso.

2. Mais testes condicionais: Crie pelo menos um resultado Verdadeiro e um Falso para cada um dos testes seguintes:

- Igualdade e desigualdade com strings
- Usando o método lower()
- Numéricos envolvendo igualdade e desigualdade, maior que menor que, maior ou igual a, e menor ou igual a
- Usando a palavra-chave **and** e a palavra-chave **or**
- Se um item está em uma lista
- Se um item não está em uma lista

Exercícios

- 1. Olá Admin:** Faça uma lista de cinco ou mais nomes de usuário, incluindo o nome 'admin'. Imagine que você está escrevendo um código que imprimirá uma saudação para cada usuário depois de fazer login em um site. Percorra a lista e imprima uma saudação para cada usuário:
 - Se o nome de usuário for 'admin', imprima uma saudação especial, como Olá admin, você gostaria de ver um relatório de status?
 - Caso contrário, imprima uma saudação genérica, como Olá Jaden, obrigado por **logar** novamente.
- 2. Nenhum usuário:** adicione um teste if a `hello_admin.py` para garantir que a lista de usuários não esteja vazia.
 - Se a lista estiver vazia, imprima a mensagem Precisamos encontrar alguns usuários!
 - Remova todos os nomes de usuário da sua lista e certifique-se de que a mensagem é impressa.
- 3. Verificando nomes de usuário:** Faça o seguinte para criar um programa que simule como os sites garantem que todos tenham um nome de usuário exclusivo.
 - Faça uma lista de cinco ou mais nomes de usuário chamados `current_users`.
 - Faça outra lista de cinco nomes de usuário chamados `new_users`. Certifique-se de que um ou dois dos novos nomes de usuário também estão na lista `current_users`.
 - Percorra a lista `new_users` para ver se cada novo nome de usuário já sido usado. Se tiver, imprima uma mensagem informando que a pessoa precisará inserir um novo usuário. Se um nome de usuário não foi usado, imprima uma mensagem dizendo que o nome de usuário está disponível.
 - Certifique-se de que sua comparação não diferencia maiúsculas de minúsculas. Se 'joão' foi usado, 'JOÃO' não deve ser aceito. (Para fazer isso, você precisará fazer uma cópia do `current_users` contendo as versões minúsculas de todos os usuários existentes.)
- 4. Números Ordinais:** Os números ordinais indicam sua posição em uma lista, como como 1º ou 2º. A maioria dos números ordinais termina em ésimos, exceto 1, 2 e 3.
 - Armazene os números de 1 a 9 em uma lista.
 - Percorra a lista.
 - Use uma cadeia if-elif-else dentro do loop para imprimir a terminação ordinal apropriada para cada número. Sua saída deve ser '1st 2nd 3rd 4th 5th 6th 7th 8th 9th', e cada resultado deve estar em uma linha separada.

Sentenças **if**

- Alternativa simples
- Alternativa composta

```
if teste_condicional:  
    Faz alguma coisa
```

```
idade = 22  
if idade >= 16:  
    print("Você pode votar!")
```

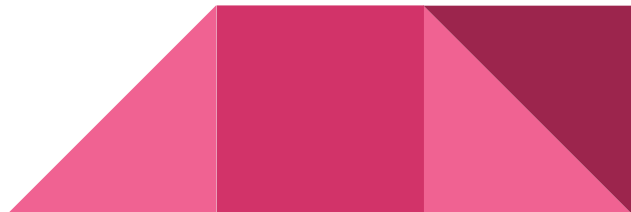
```
idade = 22  
if idade >= 16:  
    print("Você pode votar!")  
    print("Você já tirou o título?")
```

```
idade = 22  
if idade >= 16:  
    print("Você pode votar!")  
    print("Você já tirou o título")  
else:  
    print("Desculpe, você não vota!")
```

Sentenças **if**

- Cadeia de Alternativas

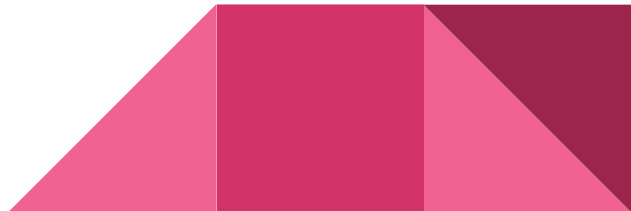
```
idade = 22
if idade <= 10:
    print("Ingresso = isento!")
elif idade < 18:
    print("Ingresso = R$ 20,00")
elif idade < 60:
    print("Ingresso = R$ 40,00")
else:
    print("Ingresso = isendo!")
```



Sentenças **if** com Listas

- Verificando se a lista está vazia

```
acessorios = []  
#['relógio', 'óculos', 'chapéu', 'anel']  
if acessorios:  
    for item in acessorios:  
        print("Coloque ", item)  
else:  
    print("Vc não tem acessório!")
```



Sentenças **if** com Abrangência de Listas

- Filtrando componentes de uma lista

```
usuarios = ['admin','joao','maria','carla','walter']
usuarios_comuns = []
for usuario in usuarios:
    if usuario != 'admin':
        usuarios_comuns.append(usuario)
print(usuarios_comuns)
```

```
usuarios = ['admin','joao','maria','carla','walter']
usuarios_comuns = [u for u in usuarios if u != 'admin']
print(usuarios_comuns)
```

Resumo

- Testes condicionais
- Constantes booleanas: **True**, **False**
- Sentença **if** simples, composta e cadeia
- Sentença **if** com listas
- Filtros em Abrangência de Lista

