

Estruturas de Decisão

Introdução a Programação

Casting

O que é *casting*?

Casting é um método para converter ou forçar que uma variável seja de um determinado tipo (`string` , `int` , `float` , etc). O Python realiza o *casting* automaticamente em algumas situações. Porém, em outros casos o programador precisa fazê-lo manualmente.

Como fazer *casting*?

- Existem algumas funções úteis relacionadas com o *casting*
 - `type()`
 - `int()`
 - `float()`
 - `str()`

Exemplo 1

```
1  # Python converte automaticamente
2  # a para int
3  a = 7
4  print(type(a))
5
6  # Python converte automaticamente
7  # b para float
8  b = 3.0
9  print(type(b))
10
11 # Python converte automaticamente
12 # c para float por conta da soma
13 c = a + b
14 print(c)
15 print(type(c))
16
17 # Python converte automaticamente
18 # d to float por conta da multiplicação
19 d = a * b
20 print(d)
21 print(type(d))
```

Exemplo 2

```
1  # O que acontece ao executar o
2  # código abaixo
3
4  idade = input('Qual sua idade?')
5  idade++
6  print('No próximo ano minha idade será', idade)
```

- O que acontece ao executar o código?
- Qual erro apresentado?
- Como corrigir?

Exemplo 2 - Corrigido

```
1  # O que acontece ao executar o
2  # código abaixo
3
4  # a funcao int() força que idade
5  # do tipo int
6  idade = int(input('Qual sua idade?'))
7  # variável idade é incrementada
8  idade++
9  # a mensagem é exibida
10 print('No próximo ano minha idade será', idade)
```

- Na linha 6 é realizado o *casting*
- Agora pode-se realizar a operação de incremento (linha 8)
- Não há mais erro

Estruturas de decisão

Declaração Condicional

Declarações condicionais em Python são utilizadas para executar certas partes do código apenas se determinadas condições forem atendidas. Estas declarações controlam o fluxo do código.

if

A declaração `if` executa um bloco de código caso a condição declarada seja atendida (`True`) e outro bloco caso não seja atendida (`False`).

Exemplo 3

```
1  idade = 20
2
3  if idade >= 16:
4      print("Cidadão pode votar.")
```

Exemplo 4

```
1  age = 10
2
3  if age <= 12:
4      print("Viajante gratuito.")
5  else:
6      print("Viajante pagante.")
```

Exemplo 5

```
1  nota = 45
2  res = "Aprovado" if nota ≥ 60 else "Reprovado"
3
4  print(f"Result: {res}")
```

elif

A declaração `elif` significa `else + if`. Permite a checagem de condições múltiplas e executar diferentes blocos de código de acordo com a condição verdadeira. **Cuidado**, apenas uma condição pode ser verdadeira, um único bloco de código é executado.

Exemplo 6

```
1  idade = 25
2
3  if idade <= 12:
4      print("Criança.")
5  elif age <= 19:
6      print("Adolescente.")
7  elif age <= 35:
8      print("Jovem adulto.")
9  else:
10     print("Adult.")
```

if-else aninhados

Pode-se criar estruturas de if-else dentro de outras if-else criando um aninhamento. Utiliza-se if-else aninhados para checar condições em sequência.

Exemplo 7

```
1  idade = 70
2  e_membro = True
3
4  if idade >= 60:
5      if e_membro:
6          print("Desconto senior, 30%")
7      else:
8          print("Desconto senior 20%")
9  else:
10     print("Não elegível para desconto senior.")
```

Exercícios

Exercício 1

Modificar a condição do `if-else` para que o código dentro do `else` seja executado. A saída do *script* deve ser '2'.

```
option = 2

if option > 0:
    var = 1
else:
    var = 2
print(var)
```

Exercício 2

Criar um *script* em Python que recebe dois números do usuário e a operação aritmética desejada (soma, subtração, multiplicação e divisão). Em seguida a operação é realizada com os números e o resultado exibido ao usuário.

Referências

- <https://www.geeksforgeeks.org/type-casting-in-python/>
- <https://www.geeksforgeeks.org/conditional-statements-in-python/>