

Computação I - Python

Aula 3 - Tipos de Dados

Apresentado por: Bernardo F. Costa

Produção DCC-UFRJ

Metodologia de referência <https://doi.org/10.5753/wei.2016.9683>



Tipos de dados: introdução

- O que são tipos de dados?
- Dados compartilham representação, propriedades e operações
- Tipos diferentes podem ter dificuldade em operar
- Casting: conversão entre tipos
- Conversão: implícita \times explícita
- Como tipos funcionam em python?

Tipos de dados: introdução

- Programador deve ter preocupação mínima com tipos
- Python escolhe o tipo automaticamente
- Python tenta converter tipos automaticamente
- Podem ocorrer falhas: ambiguidade, incompatibilidade, etc
- Programador precisa entender tipos para agir quando necessário
- Sem um bom motivo, deixe o python fazer as escolhas de tipo

Tipos de dados: números

- Tipos numéricos: três possibilidades
- Inteiro ou **int**
- Ponto flutuante ou **float**
- Número complexo ou **complex**

```
# inteiro
>>> 100
100
>>> -9
-9
```

```
# ponto flutuante
>>> -9.
-9.0
>>> .4
0.4
>>> 1e3
1000.0
>>> 0.3e2
30.0
# numero complexo
>>> 1j
1j
>>> 3.2+1.1j
(3.2+1.1j)
>>> 2+0j
(2+0j)
>>> 0.1e2 + 2.2e3j
(10+2200j)
```

Tipos de dados: números

- Conversão entre numéricos:
implícita para o tipo mais abrangente
- **int** \subset **float** \subset **complex**
- Conversão explícita é possível pelo construtor mas pode falhar
- Operadores: divisão exata e exponenciação podem mudar o tipo
- Pode haver perda na conversão para menos abrangente

```
# construtores
>>> int(100)
100
>>> float(100)
100.0
>>> complex(100)
(100+0j)
# operação exige tipo mais abrangente
>>> 1/2
0.5
>>> 1+(-1)**(1/2)-1
1j
# perda na conversão
>>> int(9.9)
9
```

Tipos de dados: string

- Tipo string ou **str** representa texto
- Operadores: concatenação (+) e autoconcatenação sucessiva (*)
- Concatenação com outros tipos tem que ser explícita usando construtor

string

```
>>> 'texto'
```

```
'texto'
```

```
>>> "outro texto"
```

```
'outro texto'
```

```
>>> 'texto \com\' "duas" aspas'
```

```
'texto \com\' "duas" aspas'
```

concatenação entre strings

```
>>> 'a'+'b'
```

```
'ab'
```

autoconcatenação sucessiva de uma string

```
>>> 'dcc'*3
```

```
'dccdccdcc'
```

falha na interpretação

```
>>> '11'+3
```

```
>>> 'ab'*3.1
```

correções possíveis

```
>>> int('11')+3
```

```
14
```

```
>>> '11'+str(3)
```

```
'113'
```

```
>>> 'ab'*3
```

```
'ababab'
```

Tipos de dados: booleano

- Tipo booleano ou **bool** conhece apenas dois valores: **True** ou **False**
- Origem: resultado de comparações, expressões com outros booleanos
- Conversão é implícita para inteiros em expressões aritméticas

```
# booleano
>>> 3>2
True
>>> 10<=5
False
# conversão em expressão aritmética
>>> (3>2)+(10<=5)
1
>>> (3>2)+(10>=5)
2
>>> (3<2)+(10<=5)
0
```

Tipos de dados: booleano

- Conversão explícita é possível com **bool()**
- Números: zero é **False**, **True** é todo resto
- String: "" ou " é **False**, **True** é todo resto

```
# conversão explícita a partir de número
```

```
>>> bool(3)
```

```
True
```

```
>>> bool(-1.2)
```

```
True
```

```
>>> bool(0+0j)
```

```
False
```

```
# conversão explícita a partir de string
```

```
>>> bool("")
```

```
False
```

```
>>> bool(" False")
```

```
True
```

```
>>> bool('uma string')
```

```
True
```

```
# falha na interpretação
```

```
>>> 'booleano = '+True
```

```
# correções possíveis
```

```
>>> 'booleano = '+str(True)
```

```
'booleano = True'
```


Tipos de dados: revisão

- Tipos de dados: classificações para uso e tratamento distinto
- Em python, tipos são atribuídos automaticamente
- Erro: conversões que não conseguem ser feitas automaticamente
- Conversão explícita corrige alguns destes erros e ambiguidade
- Dicas:
 - Saiba reconhecer o tipo ao ver um dado
 - Cuidado ao usar string, demais tipos e o operador (+) juntos
 - Função **type()** retorna o tipo

Autores

- **João C. P. da Silva** ▶ Lattes
- **Carla Delgado** ▶ Lattes
- **Ana Luisa Duboc** ▶ Lattes

Colaboradores

- **Anamaria Martins Moreira** ▶ Lattes
- **Fabio Mascarenhas** ▶ Lattes
- **Leonardo de Oliveira Carvalho** ▶ Lattes
- **Charles Figueiredo de Barros** ▶ Lattes
- **Fabício Firmino de Faria** ▶ Lattes