

85ª EDIÇÃO

SEQ UFRJ

20 a 24 de agosto



Introdução à programação para ciência e engenharia em *Python*

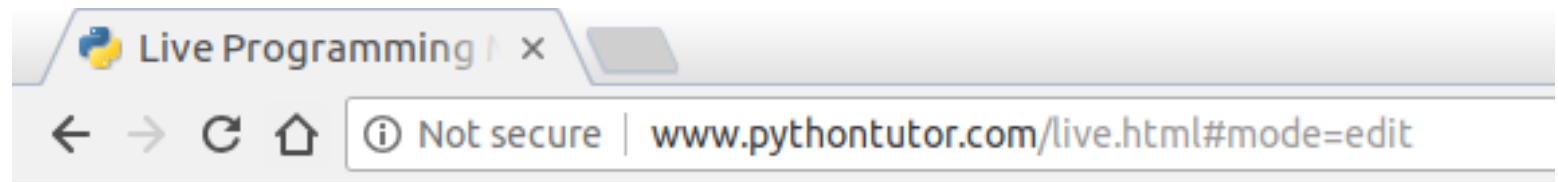
Iuri Soter Viana Segtovich

Parte 2: Lógica e Sintaxe

Gerenciamento de exceções (`try`, `except`, `finally`, `raise`)

python tutor

[www.pythontutor.com/
live.html#mode=edit](http://www.pythontutor.com/live.html#mode=edit)



Write code in Python 3.6

(drag lower right corner to resize code editor)

```
1 |
```

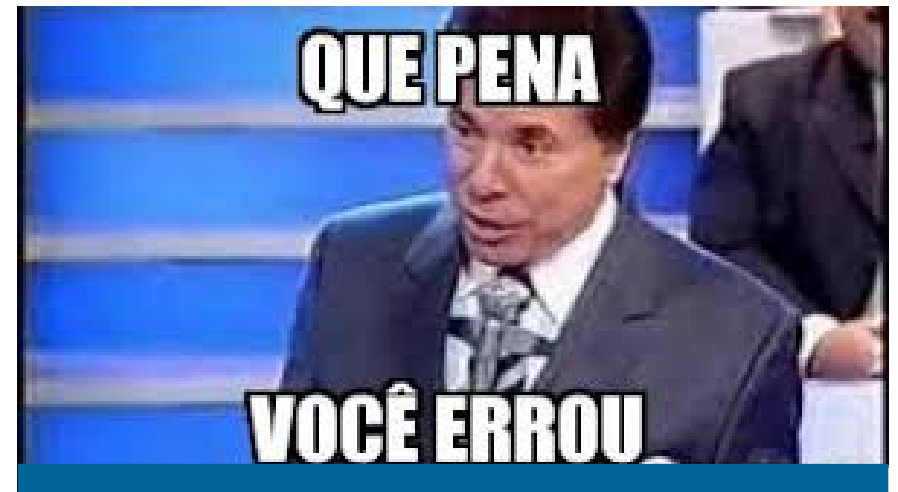
→ line that has just executed

→ next line to execute

Gerenciamento de exceções

```
x=0
a=10
b=0
try:
    temp=x
    x+=a
    x/=b
except Exception as error:
    print("Que pena, você errou.")
    print(error)
finally:
    x=temp
    print("restaurando x a partir de temp")

print("a vida continua")
```



```
Que pena, você errou.
division by zero
restaurando x a partir de temp
a vida continua
```


Exemplo

```
x=[]  
for i in range(-3,4):  
    try:  
        x.append('{:.2f}'.format(1/i))  
    except Exception:  
        x.append("INFINITO")  
    finally:  
        n=len(x)
```

Global frame

x	
i	3
n	7

list



0	1	2	3	4	5	6
"-0.33"	"-0.50"	"-1.00"	"INFINITO"	"1.00"	"0.50"	"0.33"

Prepare sua exceção

```
def funcao_bugada(x):  
    if x<0:  
        raise ValueError('Deu ruim aqui, x não pode ser negativo!')  
    else:  
        return x**.5+x**1.5+x**2.5  
  
y1=funcao_bugada(2.)  
y2=funcao_bugada(-2.)
```

ValueError: Deu ruim aqui, x não pode ser negativo!

Referências principais

[https://www.tutorialspoint.com/
python3/
python_basic_syntax.htm](https://www.tutorialspoint.com/python3/python_basic_syntax.htm)

[https://stackoverflow.com/
search](https://stackoverflow.com/search)

perguntas

