

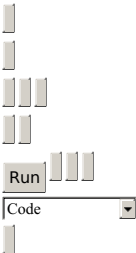
Menu

Python 3

Trusted

- [File](#)
 - [New Notebook](#)
 - [Python 3](#)
 - [Open...](#)
 - [Make a Copy...](#)
 - [Save as...](#)
 - [Rename...](#)
 - [Save and Checkpoint](#)
 - [Revert to Checkpoint](#)
 - [Quinta-feira, 3 de janeiro de 2019 às 12:07](#)
 - [Print Preview](#)
 - [Download as](#)
 - [Notebook \(.ipynb\)](#)
 - [Python \(.py\)](#)
 - [HTML \(.html\)](#)
 - [Reveal.js slides \(.html\)](#)
 - [Markdown \(.md\)](#)
 - [reST \(.rst\)](#)
 - [LaTeX \(.tex\)](#)
 - [PDF via LaTeX \(.pdf\)](#)
 - [asciidoc \(.asciidoc\)](#)
 - [custom \(.txt\)](#)
 - [custom \(.html\)](#)
 - [latex \(.tex\)](#)
 - [markdown \(.md\)](#)
 - [notebook \(.ipynb\)](#)
 - [pdf \(.tex\)](#)
 - [python \(.py\)](#)
 - [rst \(.rst\)](#)
 - [custom \(.txt\)](#)
 - [slides \(.slides.html\)](#)
 - [Deploy as](#)
 - [Trusted Notebook](#)
 - [Close and Halt](#)
- [Edit](#)
 - [Cut Cells](#)
 - [Copy Cells](#)
 - [Paste Cells Above](#)
 - [Paste Cells Below](#)
 - [Paste Cells & Replace](#)
 - [Delete Cells](#)
 - [Undo Delete Cells](#)
 - [Split Cell](#)
 - [Merge Cell Above](#)
 - [Merge Cell Below](#)
 - [Move Cell Up](#)
 - [Move Cell Down](#)
 - [Edit Notebook Metadata](#)
 - [Find and Replace](#)
 - [Cut Cell Attachments](#)
 - [Copy Cell Attachments](#)
 - [Paste Cell Attachments](#)
 - [Insert Image](#)
- [View](#)
 - [Toggle Header](#)
 - [Toggle Toolbar](#)
 - [Toggle Line Numbers](#)
 - [Cell Toolbar](#)
 - [None](#)
 - [Edit Metadata](#)
 - [Raw Cell Format](#)
 - [Slideshow](#)
 - [Attachments](#)
 - [Tags](#)
- [Insert](#)
 - [Insert Cell Above](#)
 - [Insert Cell Below](#)
- [Cell](#)
 - [Run Cells](#)
 - [Run Cells and Select Below](#)
 - [Run Cells and Insert Below](#)
 - [Run All](#)
 - [Run All Above](#)
 - [Run All Below](#)
 - [Cell Type](#)
 - [Code](#)
 - [Markdown](#)
 - [Raw NBConvert](#)
 - [Current Outputs](#)
 - [Toggle](#)
 - [Toggle Scrolling](#)
 - [Clear](#)
 - [All Output](#)
 - [Toggle](#)
 - [Toggle Scrolling](#)
 - [Clear](#)
- [Kernel](#)

- [Interrupt](#)
 - [Restart](#)
 - [Restart & Clear Output](#)
 - [Restart & Run All](#)
 - [Reconnect](#)
 - [Shutdown](#)
 - [Change kernel](#)
 - [Python 3](#)
- [Widgets](#)
 - [Save Notebook Widget State](#)
 - [Clear Notebook Widget State](#)
 - [Download Widget State](#)
 - [Embed Widgets](#)
- [Help](#)
 - [User Interface Tour](#)
 - [Keyboard Shortcuts](#)
 - [Edit Keyboard Shortcuts](#)
 - [Notebook Help](#)
 - [Markdown](#)
 - [Python Reference](#)
 - [JPython Reference](#)
 - [NumPy Reference](#)
 - [SciPy Reference](#)
 - [Matplotlib Reference](#)
 - [SymPy Reference](#)
 - [pandas Reference](#)
 - [About](#)



Teste de avaliação de declarações

Vamos testar seu conhecimento!

Teste de avaliação de declarações¶

Vamos testar seu conhecimento!

**** Use for, split() e if para criar uma declaração que imprima as palavras que começam com 's': ****

**** Use for, split() e if para criar uma declaração que imprima as palavras que começam com 's': ****

```
In [3]:
x

st
In [2]: 'Print only the words that start with s in this sentence'

for palavra in st.split():
    if palavra[0].lower() == 's':
        print(palavra)
start
s
sentence
```

**** Use range() para imprimir todos os números pares de 0 a 10. ****

**** Use range() para imprimir todos os números pares de 0 a 10. ****

```
In [4]:

a = range(0,11,2)
list(a)
Out[4]:
[0, 2, 4, 6, 8, 10]
```

**** Use a compreensão de lista para criar uma lista de todos os números entre 1 e 50 que são divisíveis por 3. ****

**** Use a compreensão de lista para criar uma lista de todos os números entre 1 e 50 que são divisíveis por 3. ****

```
In [6]:

a = [i for i in range(1,51) if i%3==0]
list(a)
Out[6]:
[3, 6, 9, 12, 15, 18, 21, 24, 27, 30, 33, 36, 39, 42, 45, 48]
```

**** Percorra a string abaixo e se o comprimento de uma palavra for par imprima "é par!" ****

**** Percorra a string abaixo e se o comprimento de uma palavra for par imprima "é par!" ****

```
In [6]:

str_r01. 'Print every word in this sentence that has an even number of letters'
```

In [0]:

```
for i in st.split():

    if len(i) % 2 == 0:

        print (i + ": É par!")# Código aqui
word: É par!
in: É par!
this: É par!
sentence: É par!
that: É par!
an: É par!
even: É par!
number: É par!
of: É par!
```

**** Escreva um programa que imprima os números inteiros de 1 a 100. Para múltiplos de três imprima "Fizz" ao invés do número, e para os múltiplos de cinco imprima "Buzz". Para números que são múltiplos de três e cinco imprima "FizzBuzz". ****

In [9]:

```
for i in range(1,101):

    if(i % 3 == 0 and i % 5 == 0):

        print("FizzBuzz")

    elif(i % 5 == 0):

        print("Buzz")

    elif(i % 3 == 0):

        print("Fizz")

    else:
```

```
        print (i)
1
2
Fizz
4
Buzz
Fizz
7
8
Fizz
Buzz
11
Fizz
13
14
FizzBuzz
16
17
Fizz
19
Buzz
Fizz
22
23
Fizz
Buzz
26
Fizz
28
29
FizzBuzz
31
32
Fizz
34
Buzz
Fizz
37
38
Fizz
Buzz
41
Fizz
43
44
FizzBuzz
46
47
Fizz
49
Buzz
Fizz
52
53
Fizz
Buzz
56
Fizz
58
59
FizzBuzz
61
62
Fizz
64
Buzz
Fizz
67
68
Fizz
Buzz
71
Fizz
73
74
FizzBuzz
76
77
Fizz
79
Buzz
Fizz
82
83
Fizz
Buzz
86
Fizz
88
89
FizzBuzz
91
92
Fizz
94
Buzz
```

```
Fizz
97
98
Fizz
Buzz
```

~~** Use Compreensão em listas para criar uma lista das primeiras letras de cada palavra na string abaixo: **~~

**** Use Compreensão em listas para criar uma lista das primeiras letras de cada palavra na string abaixo: ****

In [8]:

```
st = 'Create a list of the first letters of every word in this string'
```

In [11]:

```
st = 'Create a list of the first letters of every word in this string'
```

```
a = [palavras[0] for palavras in st.split()]
```

```
list(a)
```

Out[11]:

```
['C', 'a', 'l', 'o', 't', 'f', 'l', 'o', 'e', 'w', 'i', 't', 's']
```

```
### Muito bom!
```

Muito bom! 