



INSTITUTO FEDERAL
Bahia
Campus Vitória da Conquista



Introdução a programação em Python

Flávio Filipe L. S. Porto

O que é Python?

Linguagem de programação criada por **Guido Van Rossum** em 1991 com foco em produtividade e legibilidade.



Por quê Python?

- Fácil e intuitiva
- Multiplataforma
- Linguagem de propósito geral
- Orientada a Objeto
- Diversas bibliotecas
- Grande comunidade
- Livre



Zen of Python

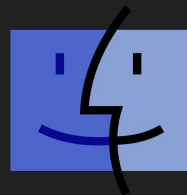


Tim Peters

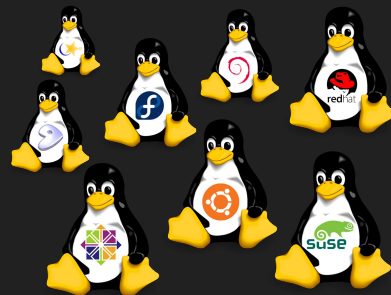


- **Bonito é melhor que feio.**
- **Explícito é melhor que implícito.**
- **Simples é melhor que complexo.**
- **Complexo é melhor que complicado.**
- **Linear é melhor do que aninhado.**
- **Esparsos é melhor que denso.**
- **Legibilidade conta.**
- **Casos especiais não são especiais o bastante para quebrar as regras.**
- **Ainda que praticidade vença a pureza.**
- **Erros nunca devem passar silenciosamente.**
- **A menos que sejam explicitamente silenciados.**
- **Diante da ambiguidade, recuse a tentação de adivinhar.**
- **Deveria haver um – e preferencialmente só um – modo óbvio para fazer algo.**
- **Embora esse modo possa não ser óbvio a princípio a menos que você seja holandês.**
- **Agora é melhor que nunca.**
- **Embora nunca frequentemente seja melhor que *já*.**
- **Se a implementação é difícil de explicar, é uma má idéia.**
- **Se a implementação é fácil de explicar, pode ser uma boa idéia.**
- **Namespaces são uma grande idéia – vamos ter mais dessas!**

Quem utiliza Python



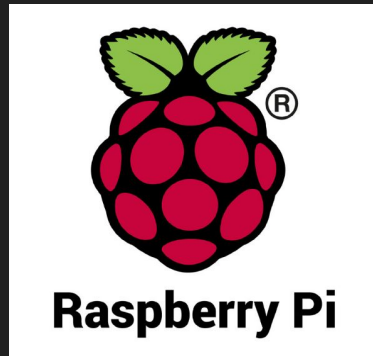
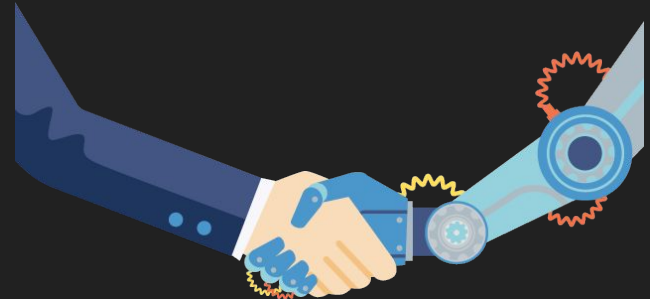
Mac OS



Week-IT
Semana de Tecnologia da Informação

O que fazer com Python

django



Python VS Java VS C++

```
print('Hello World')
```

```
Class HelloWorld{  
    public static void Main(String[] args){  
        System.out.println("Hello World");  
    }  
}
```

```
#include <stdio.h>  
  
int main(){  
    printf("Hello World");  
    return 0;  
}
```

Python VS Java VS C++

```
print('Hello World' * 50)
```

```
Class HelloWorld{  
    public static void Main(String[] args){  
        for(int i = 0; i < 50; i++){  
            System.out.println("Hello World");  
        }  
    }  
}
```

```
#include <stdio.h>  
  
int main(){  
    for(int i=0; i<50; i++){  
        printf("Hello World");  
    }  
    return 0;  
}
```


Instalação

python.org

The screenshot shows the python.org website with a dark blue header and a lighter blue main content area. The header includes navigation links: Python, PSF, Docs, PyPI, Jobs, and Community. Below the header is the Python logo and a search bar. The main content area features a horizontal menu with links: About, Downloads, Documentation, Community, Success Stories, News, and Events. The 'Downloads' link is active, and a dropdown menu is open, listing options: All releases, Source code, Windows, Mac OS X, Other Platforms, License, and Alternative Implementations. The 'Windows' option is selected, leading to a 'Download for Windows' section. This section contains buttons for 'Python 3.6.3' and 'Python 2.7.14'. Below the buttons, a note states: 'Note that Python 3.5+ cannot be used on Windows XP or earlier.' It also mentions that Python can be used on many operating systems and environments, with a link to 'View the full list of downloads.' At the bottom of the page, a text block reads: 'Python is a programming language that lets you work quickly and integrate systems more effectively. >>> [Learn More](#)'.

Python

PSF

Docs

PyPI

Jobs

Community

python™

Search

GO

Socialize

About

Downloads

Documentation

Community

Success Stories

News

Events

```
# Python 3:
>>> def fib(n):
>>>     a, b = 0, 1
>>>     while a < n:
>>>         print(a, end=' ')
>>>         a, b = b, a+b
>>>     fib(1000)
0 1 1 2 3 5
987
```

All releases

Source code

Windows

Mac OS X

Other Platforms

License

Alternative Implementations

Download for Windows

Python 3.6.3

Python 2.7.14

Note that Python 3.5+ cannot be used on Windows XP or earlier.

Not the OS you are looking for? Python can be used on many operating systems and environments.

[View the full list of downloads.](#)

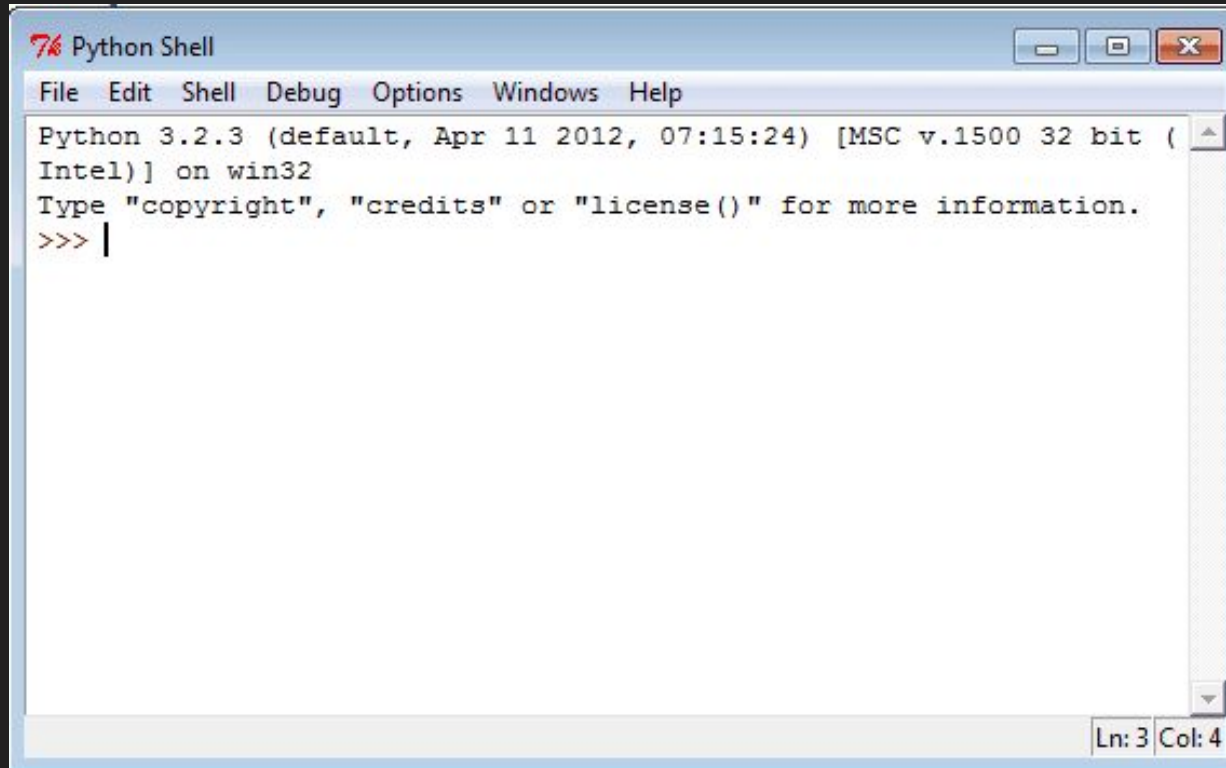
Python is a programming language that lets you work quickly and integrate systems more effectively. >>> [Learn More](#)

Instalação



Instalação

IDLE



The image shows a screenshot of the Python Shell window within the IDLE (Integrated Development and Learning Environment) application. The window has a title bar that reads "Python Shell" with a red Python logo icon on the left and standard Windows window controls (minimize, maximize, close) on the right. Below the title bar is a menu bar with the following options: File, Edit, Shell, Debug, Options, Windows, and Help. The main text area of the window displays the following text: "Python 3.2.3 (default, Apr 11 2012, 07:15:24) [MSC v.1500 32 bit (Intel)] on win32". This is followed by a prompt line: "Type 'copyright', 'credits' or 'license()' for more information." Below this, the prompt ">>>" is shown with a vertical cursor line to its right. At the bottom right of the window, a status bar indicates the current cursor position: "Ln: 3 Col: 4".

```
Python 3.2.3 (default, Apr 11 2012, 07:15:24) [MSC v.1500 32 bit (Intel)] on win32
Type "copyright", "credits" or "license()" for more information.
>>> |
```

Olá Mundo

```
>>> print('Olá Mundo')
```



Operadores

OPERADOR	DESCRIÇÃO	EXEMPLO
+	Adição	$2 + 3$
-	Subtração	$2 - 3$
*	Multiplicação	$2 * 3$
/	Divisão	$5 / 2$
//	Divisão Inteira	$5 / 2$
**	Exponencial	$5 ** 2$
%	Módulo	$5 \% 2$

Tipos de dados

type('Olá Mundo')

TIPOS	EXEMPLOS
Inteiro	42
Ponto Flutuante	42.5
STRING	'Olá Mundo'
Tuplas	(1,2,'a')
Listas	[1,2,['a',3], 'b']
Dicionários	{'a':1, 'b':[1,2,3], 'c': 'nome'}
Booleanos	True , False

String - str()
Inteiro - int()
Float - float()

Variáveis

```
>>> a = 10
>>> b = 2
>>> a + b
>>> 12
>>> a ** b
>>> 100
```

```
>>> nome = 'Flávio Filipe'
>>> nome[0]
>>> 'F'
>>> nome[1:4]
>>> 'láv'
```

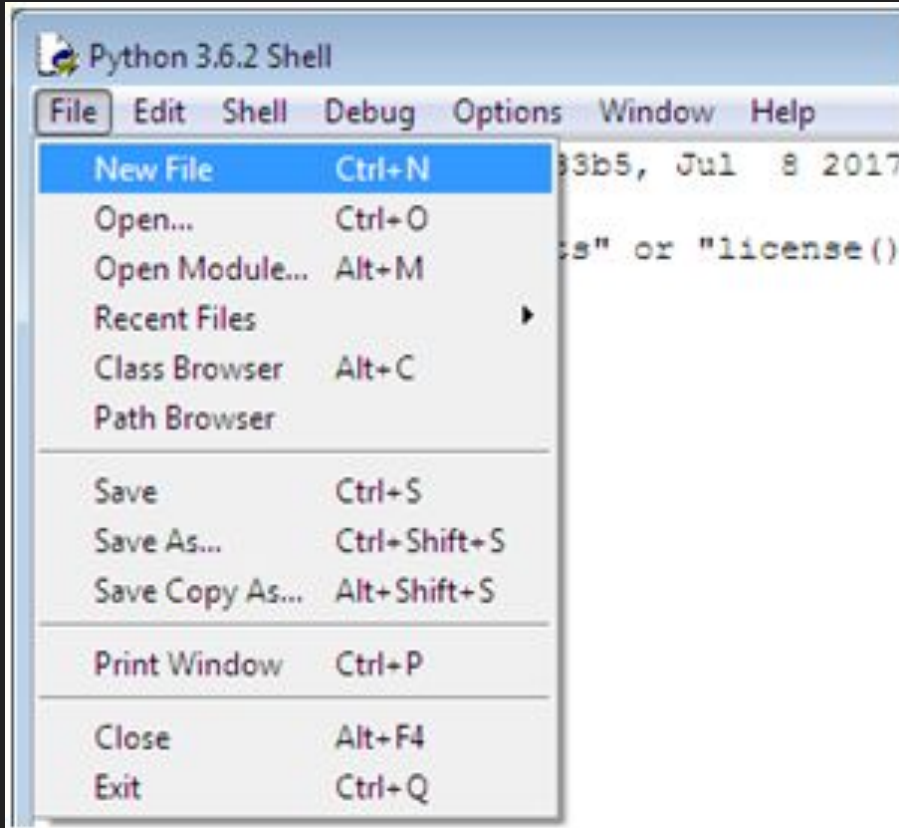
```
>>> animais = ['Gato', 'Cachorro', 'Coelho']
>>> animais[0]
>>> 'Gato'
```

Strings

```
>>> nome = 'Flávio Filipe'
>>> len(nome)
>>> nome.replace('Filipe', 'Teste')
>>> nome.count('i')
>>> nome.upper()
>>> nome.lower()
>>> nome.title()
```

```
>>> nome[1:3]
>>> nome[:3]
>>> nome[-1]
>>> nome[-5:]
>>> nome[1:6:2]
>>> n = nome.split(' ')
```


Script



`print("Primeiro programa")`

Executar: F5

Comentários em Python

```
"""  
Comentários em Python  
"""  
  
#print('Linha 1')  
  
print('Linha 2')
```

Entrada e Saída de dados

```
nome = input(' Digite seu nome ')\nprint('Olá ' + nome)\nou\nprint('Olá {}'.format(nome))
```

Para concatenar variáveis em uma frase utiliza-se o operador + ou o método *.format()*.

As variáveis contidas no método substituem os { } na frase pela ordem de precedência

Entrada e Saída de dados

Crie um programa que leia 3 notas, apresente a média. Escreva aprovado se a média for maior ou igual a 7 ou reprovado se for menor que 7.



Programa calc_media.py

```
n1 = float( input('Primeira nota: ' ) )  
n2 = float( input('Segunda nota: ' ) )  
n3 = float( input('Terceira nota: ' ) )  
media = ( n1+n2+n3 ) / 3  
  
print( 'Sua média é ' + media )
```

Para receber um valor é necessário convertê-lo para um número inteiro (*int*) ou decimal (*float*).

Programa calc_media.py

```
n1 = float( input('Primeira nota: ' ) )  
n2 = float( input('Segunda nota: ' ) )  
n3 = float( input('Terceira nota: ' ) )  
media = ( n1+n2+n3 ) / 3  
  
print( 'Sua média é ' + str(media) )
```

Para juntar um número a uma string é necessário convertê-lo para o tipo texto.

Programa calc_media.py

```
n1 = float(input('Primeira nota: '))
n2 = float(input('Segunda nota: '))
n3 = float(input('Terceira nota: '))
media = (n1+n2+n3)/3

if(media > 7):
    print('Sua média é {}. Você foi aprovado!'.format(media))
else:
    print('Sua média é {}. Você foi reprovado!'.format(media))
```

Funções

Uma função é um conjunto de códigos pré definidos que podem ser chamadas a qualquer momento.

Ex:

$f(x) = x^2 + 2x + 1$

$f(2) = 8$

`print('Olá Mundo')`

`input('Digite seu nome: ')`

`exit()`

`ola(nome)`

```
>>> def f(x):  
        print(x**2+2*x+1)
```

```
>>> f(2)
```

```
>>> def ola(nome):  
        print('Olá '+nome)
```

```
>>> ola('Flávio')
```


Funções

Crie uma calculadora com as funções de adição, subtração, divisão e multiplicação. Solicite o primeiro e o segundo valor e a operação ser executado entre eles.



Programa calculadora.py

```
n1 = float(input('Primeiro valor: '))
n2 = float(input('Segundo valor: '))
print("""QUAL OPERAÇÃO DESEJA?
1- Adição
2- Subtração """)
op = int(input('>: '))

print('{} {} {} = x'.format(n1, op, n2))
```

Programa calculadora.py

```
n1 = float(input('Primeiro valor: '))
n2 = float(input('Segundo valor: '))
print("""QUAL OPERAÇÃO DESEJA?
1- Adição
2- Subtração """)
op = int(input('>: '))

if(op == 1):
    print('{} + {} = {}'.format(n1, n2, n1+n2))
elif(op == 2):
    print('{} - {} = {}'.format(n1, n2, n1-n2))
```

Programa calculadora.py

```
def add(n1, n2):  
    print('{} + {} = {}'.format(n1, n2, n1+n2))
```

```
def sub(n1, n2):  
    print('{} - {} = {}'.format(n1, n2, n1-n2))
```

Programa calculadora.py

```
n1 = float(input('Primeiro valor: '))
n2 = float(input('Segundo valor: '))
print("""QUAL OPERAÇÃO DESEJA?
1- Adição
2- Subtração """)
op = int(input('>: '))
{ Funções aqui }
if(op == 1):
    add(n1, n2)
elif(op == 2):
    sub(n1, n2)
```

Módulos

Módulos são uma classe com uma ou várias funções que podem ser reaproveitadas em vários projetos.

No Python existem milhares de módulos para diversas funções como:

- Cálculos - Math
- Cálculo com números complexos - Cmath
- Criar jogos - PyGame
- Criar interfaces Gráficas - TKInter, Turtle

Para usar um módulo nativo basta importá-lo:
import nome



Módulo Random

```
>>> import random  
>>> random.randint(5,10)  
>>> cores = ['vermelho', 'amarelo', 'azul']  
>>> cores[random.randint(0,2)]
```

Módulo Turtle

- Avançar x pixels - `turtle.forward(x)` ou `turtle.fd(x)`
- Virar a esquerda x graus - `turtle.left(x)` ou `turtle.lt(x)`
- Virar para direita x graus - `turtle.right(x)` ou `turtle.rt(x)`
- Alterar cor de fundo - `turtle.bgcolor('orange')`
- Largura da reta - `turtle.pensize(x)`
- Cor da reta - `turtle.pencolor('red')`
- Seta tartaruga - `turtle.shape('turtle')`
- Velocidade - `turtle.speed(10)`

Módulo Turtle

```
import turtle
```

```
turtle.forward(100)
```

```
turtle.left(45)
```

```
turtle.forward(100)
```

```
turtle.left(45)
```

```
from turtle import *
```

```
forward(100)
```

```
left(45)
```

```
forward(100)
```

```
left(45)
```

Módulo Turtle

```
import turtle

for i in range(8):
    turtle.forward(100)
    turtle.left(45)
```

Módulo Turtle

```
import turtle
import random
colors = ['red','yellow','blue','green','pink','black','purple','sienna','gray']
turtle.pensize(5)
for i in range(8):
    num = random.randint(0,8)
    turtle.pencolor(colors[num])
    turtle.forward(100)
    turtle.left(45)
```

Módulo Turtle

```
import turtle  
def spiral(voltas, avancar):  
    turtle.speed(10)  
    for i in voltas:  
        turtle.forward(frente)  
        turtle.left(90)  
        avancar = avancar + 10  
  
spiral(50, 10)
```

Módulo Turtle

```
import turtle
turtle.speed(10)
turtle.pensize(5)
turtle.pencolor('red')
turtle.bgcolor('black')
turtle.shape('turtle')
turtle.forward(100)
turtle.left(90)
turtle.right(90)
```

```
import turtle
avancar = 20
turtle.pencolor('red')
turtle.bgcolor('black')
turtle.pensize(5)
for i in range(30):
    turtle.forward(avancar)
    turtle.left(90)
    avancar = avancar+5
```

Dúvidas?

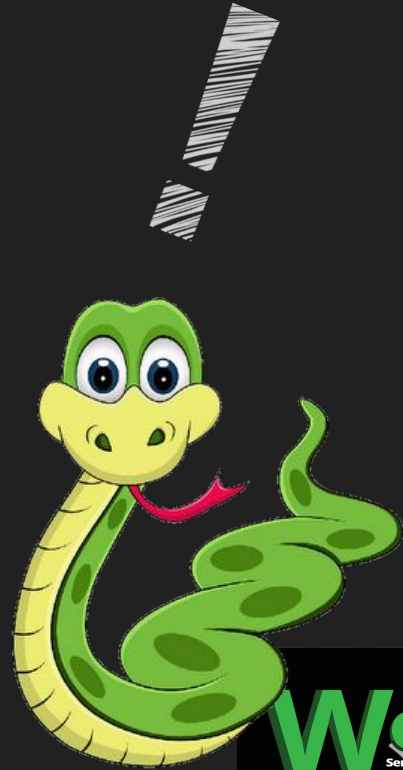
Email: flaviofilipe97@gmail.com



Week-IT
Semana de Tecnologia da Informação

Obrigado!

Email: flaviofilipe97@gmail.com



Week-IT
Semana de Tecnologia da Informação