

Fundamentos do Bootcamp

Desenvolvedor Python

Capítulo 1.

Prof. Túlio Philipe Ferreira e Vieira



Fundamentos do Bootcamp

Desenvolvedor Python

Aula 1.1. Introdução ao Python

Prof. Túlio Philipe Ferreira e Vieira

Nesta aula

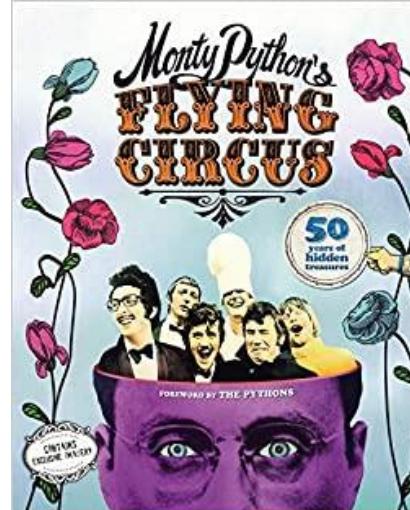
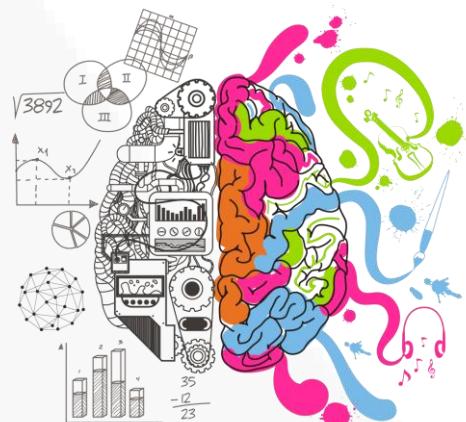


1. A linguagem Python.
2. Como o programa é executado?

Linguagem Python

IGTI

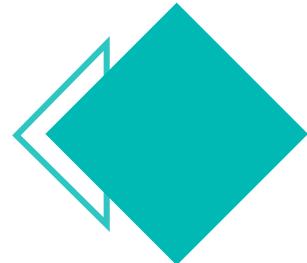
- Criada no final dos anos 80.
- Guido Van Rossum.





1994

Lançamento do Python 1.0



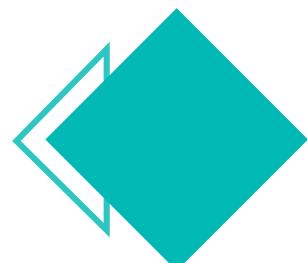
2000

Lançamento do Python 2.0



Dezembro de 2008

Python 3.0

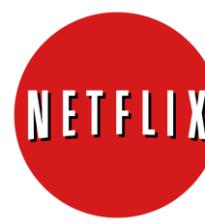


Outubro de 2019

Python 3.8

Por que Python?

1. Alto nível.
2. Sintaxe simples e intuitiva.
3. Code Readability.
4. Interpreted – velocidade.
5. Desenvolvida para matemáticos e ciência da computação.
6. Código aberto.



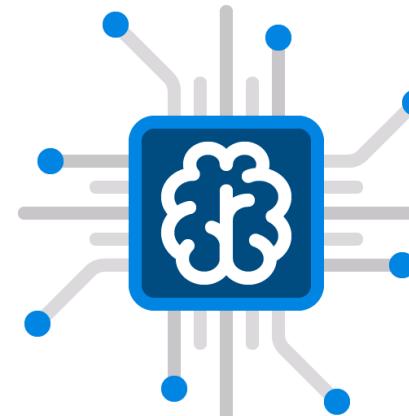
Áreas de aplicação



Aprendizado de Máquina



Ciência de dados



Inteligência Artificial



Flask

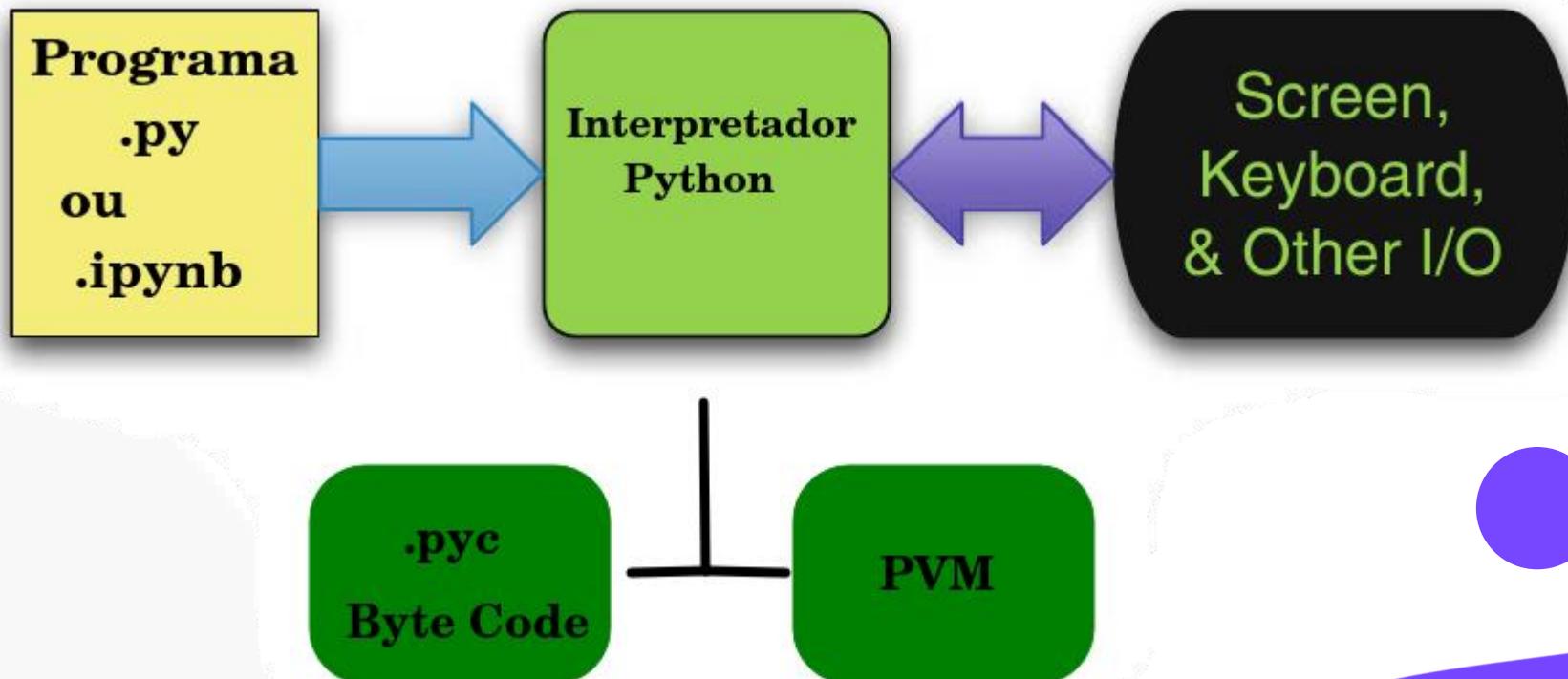
Web



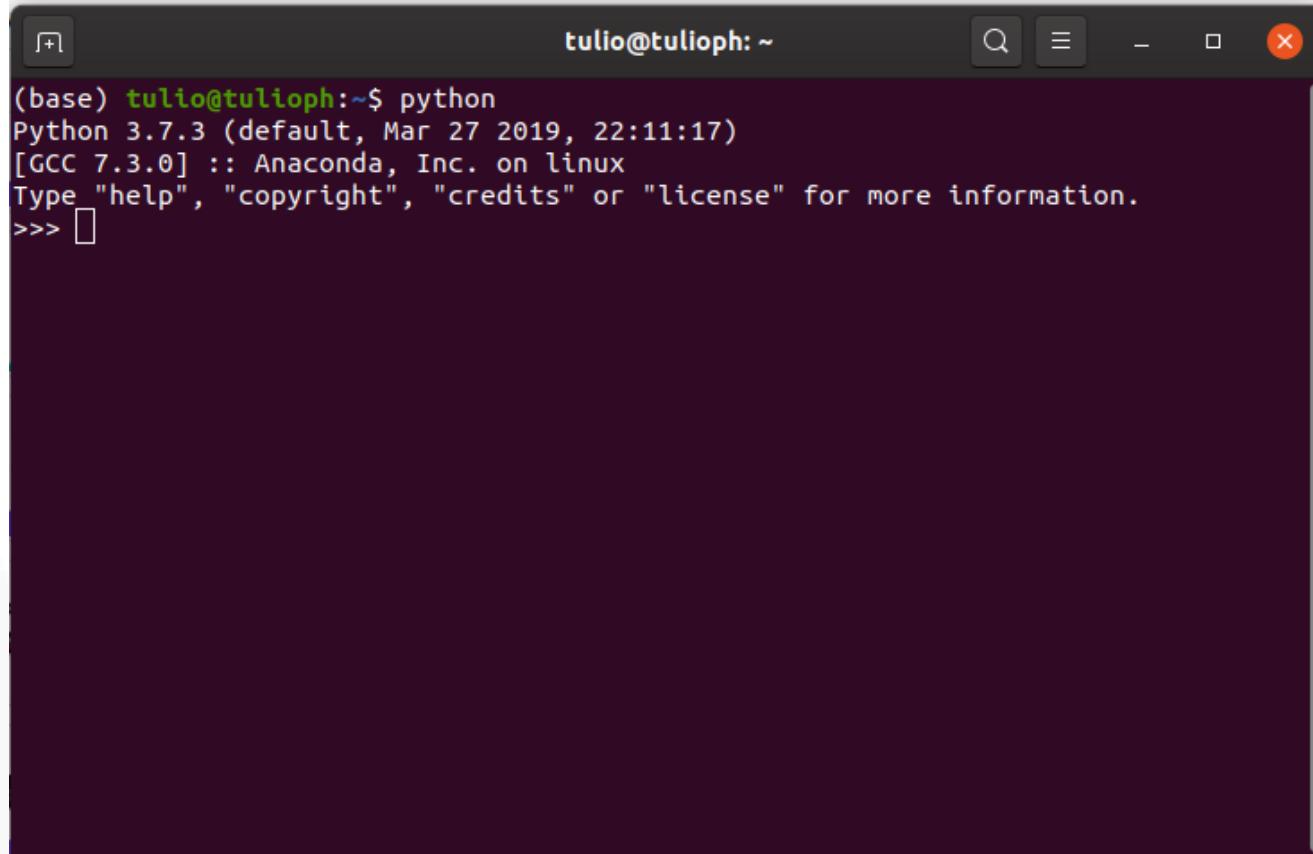
Jogos

IGTI

Como o código “roda”



Como rodar o código?



```
tulio@tulioph:~$ python
Python 3.7.3 (default, Mar 27 2019, 22:11:17)
[GCC 7.3.0] :: Anaconda, Inc. on linux
Type "help", "copyright", "credits" or "license" for more information.
>>> 
```

Prompt Interativo

Como rodar o código?



Spyder (Python 3.7)

Arquivo Editar Pesquisar Código Executar Depurar Consoles Projetos Ferramentas Ver Ajuda

Editor - /home/tulio/Documents/Scraping/recipe_2.py

modo_segmentacao_parceiros_automatizado.py X main_segmentacao.py X recipe_2.py X

```
1 #!/usr/bin/env python3
2 # -*- coding: utf-8 -*-
3 """
4 Created on Fri May 22 13:04:14 2020
5
6 @author: tulio
7 """
8 from selenium import webdriver
9
10 def get_upcoming_events(url):
11     driver = webdriver.Firefox()
12     #driver.add_argument('headless')
13     #driver = webdriver.PhantomJS(executable_path='/home/tulio/Downloads/phantomjs-2.1.1-linux-x86_64/bin/phantomjs')
14     driver.get(url)
15     events = driver.find_elements_by_xpath("//ul[contains(@class, "list-recent-events")]//li")
16     for event in events:
17
18         event_details = dict()
19         event_details['name'] = event.find_element_by_xpath('h3[@class="event-title"]/a').text
20         event_details['location'] = event.find_element_by_xpath('p/span[@class="event-location"]').text
21         event_details['time'] = event.find_element_by_xpath('p/time').text
22
23         print(event_details)
24     driver.close()
25 get_upcoming_events('https://www.python.org/events/python-events/')
```

Explorador de variáveis

Nome	Tipo	Tamanho	Valor
bus_cardCity	str	1	5d420092b374490035a51f90
data_final_previsao	str	1	2019-09-30 23:59:59
data_final_referencia	str	1	2019-04-30 23:59:59
data_inicial_previsao	str	1	2019-09-01 00:00:00
data_inicial_referencia	str	1	2019-04-01 00:00:00
df_recargas_medias_periodo	DataFrame	(2, 35)	Column names: oid_id, oid_busCard, date,...
df_verifica_catracadas_prev	DataFrame	(209886, 35)	Column names: oid_id, oid_busCard, date,...
df_verifica_catracadas_ref	DataFrame	(196421, 35)	Column names: oid_id, oid_busCard, date,...

Explorador de variáveis Explorador de arquivos Ajuda

Console IPython

Console 3/A X

```
gerar_arquivos_metricas: Métricas de catracadas dentro e fora pico calculadas
gerar_arquivos_metricas: Extração de dados concluída. Total de linhas: 4837
Aplica a previsão para o mês de Previsão
```

```
In [5]: runfile('/home/tulio/Documents/analise_desespero/dados_novas_catracadas/main_segmentacao.py', wdir='/home/tulio/Documents/analise_desespero/dados_novas_catracadas')
Reloaded modules: modelo_segmentacao_parceiros_automatizado
Conexão com Dremio estabelecida com sucesso.

Não existem dados suficientes do parceiro: 5d10dff7c6f3b900106814e4 para o intervalo informado:
2019-04-01 00:00:00 -- 2019-04-30 23:59:59
Conexão com Dremio estabelecida com sucesso.

Retira as métricas para o mês de referência 04_2019
```

IDE - Integrated Development Environment

Conclusão



1. História do python.
2. Como um programa é executado.

Próxima aula



1. Como instalar o python.
2. IDE para python.



Fundamentos do Bootcamp

Desenvolvedor Python

Aula 1.2. Configuração do Ambiente Python

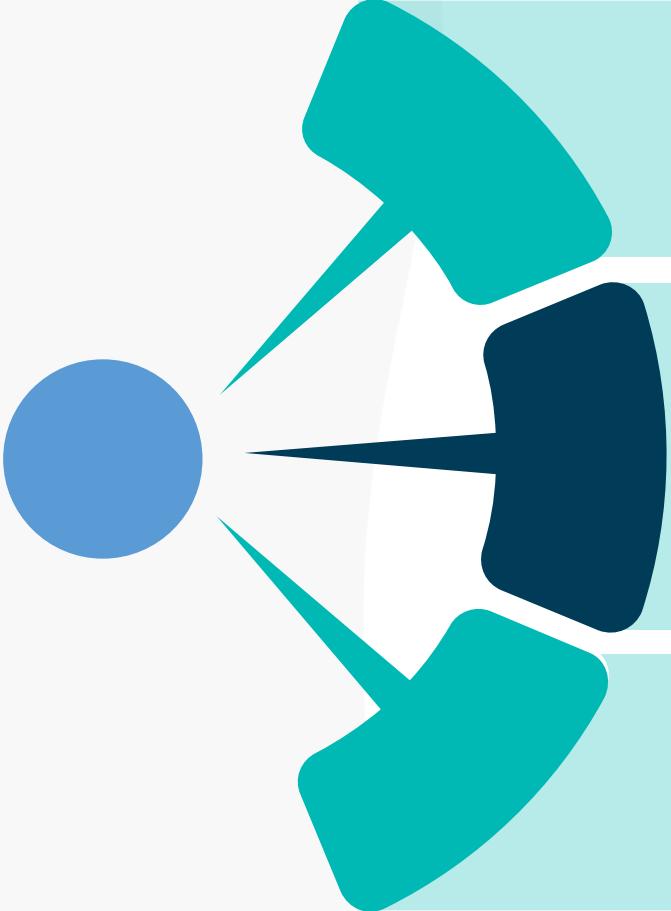
Prof. Túlio Philipe Ferreira e Vieira

Nesta aula



1. Preparação do ambiente.
2. IDE para Python.
3. Python 2 x Python 3.

Instalação para Windows



Acesse o link:

<https://www.python.org/downloads/.>

Execute o arquivo baixado.

Selecione o local padrão de instalação.

Selecione os pacotes de instalação padrão.

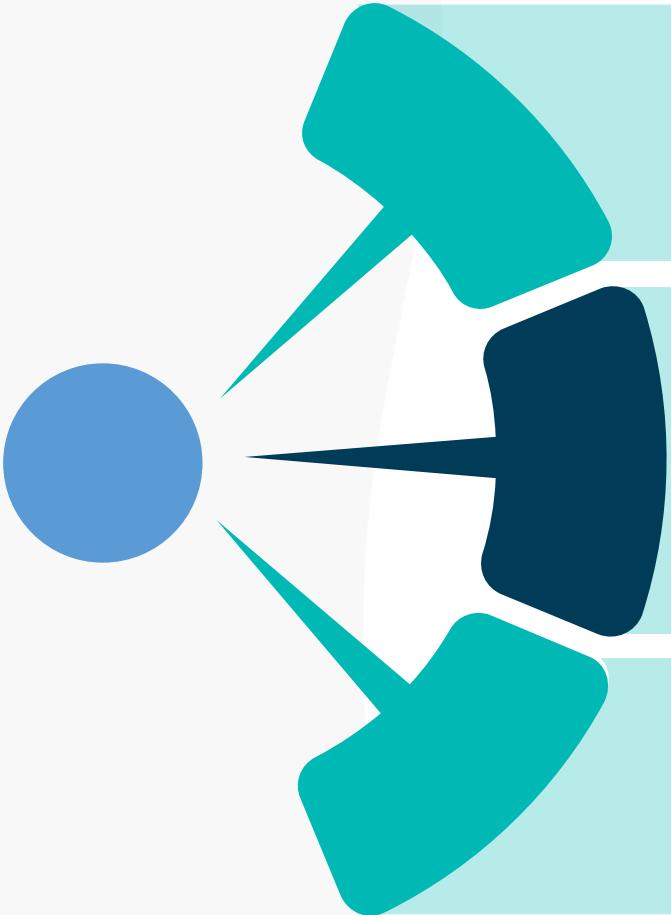
Prossiga com a instalação.

Instalação no windows



The screenshot shows the Python website's main navigation bar at the top, with links for Python, PSF, Docs, PyPI, Jobs, and Community. Below the navigation is the Python logo and a search bar with a "GO" button. The main content area has a blue header with tabs: About, Downloads, Documentation, Community, Success Stories, News, and Events. The "Downloads" tab is highlighted with a red box. A dropdown menu is open under "Downloads", listing "All releases", "Source code", "Windows" (which is also highlighted with a red box), "Mac OS X", "Other Platforms", "License", and "Alternative Implementations". To the right of the dropdown, there's a section titled "Python Source" showing "Python 3.8.3". Below it, text says "Not the OS you are looking for? Python can be used on many operating systems and environments." and a link "View the full list of downloads.". At the bottom of the page, a banner states "Python is a programming language that lets you work quickly and integrate systems more effectively." followed by a "Learn More" link.

Instalação para OS X



A decorative graphic in the bottom left corner features a large blue circle at the bottom left, overlaid by several smaller, semi-transparent shapes: a teal triangle pointing up, a dark teal circle, a white circle, and a teal circle pointing down. A thin white horizontal line extends from the center of the graphic towards the right side of the slide.

Instale o MacPorts

[MacPorts \(<http://www.macports.org>\)](http://www.macports.org)

Realize o download do arquivo dmg
Ou utilize o comando `python`

Utilize o comando para instalação

`sudo port install pythonXX`

Instalação no MAC OS

A screenshot of the Python.org website's "Downloads" page. The top navigation bar includes links for Python, PSF, Docs, PyPI, Jobs, and Community. Below the header is a search bar with a magnifying glass icon and a "Search" button. A "Donate" button is also present. The main content area features a sidebar with Python code snippets and a list of download options for Mac OS X. The central column displays the "Python Source" section, which includes a "Python 3.8.3" button and a note about Python's compatibility across various operating systems. A call-to-action at the bottom encourages users to learn more about Python.

Python 3: Li
>>> fruits = [
>>> loud_fruit
fruits]
>>> print(loud_fruit)
['BANANA', 'APPEL']

List and the
>>> list(enum
[(0, 'Banana')]

All releases
Source code
Windows
Mac OS X
Other Platforms
License
Alternative Implementations

Python Source

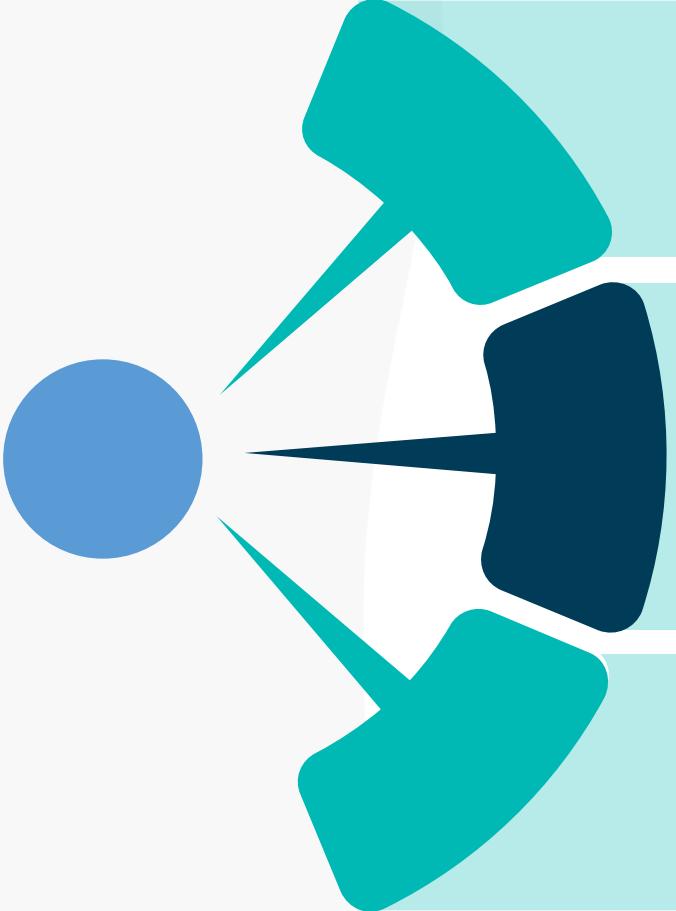
Python 3.8.3

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 para Linux



A large decorative graphic on the left side of the slide features a blue circle at the bottom left, three teal teardrop shapes pointing towards it from the top, and two dark blue teardrop shapes pointing towards it from the middle-left. A thin white horizontal line extends from the center of the graphic across the slide.
Abra o terminal
Ctrl+Alt+t

Verifique a versão instalada
`python -V`

```
sudo apt-get install python3.7  
sudo apt-get install idle-python3.7
```

Instalação no Linux



The screenshot shows the Python.org homepage. At the top, there is a navigation bar with tabs: Python, PSF, Docs (which is highlighted in yellow), PyPI, Jobs, and Community. Below the navigation bar is the Python logo and a search bar with buttons for "Donate", "Search", "GO", and "Socialize". A secondary navigation bar below the logo includes links for About, Downloads, Documentation, Community, Success Stories, News, and Events. The main content area features a code snippet in a terminal window:

```
# For loop on a list
>>> numbers = [2, 4, 6, 8]
>>> product = 1
>>> for number in numbers:
...     product = product * number
...
>>> print('The product is:', product)
The product is: 384
```

To the right of the code, there is a section titled "All the Flow You'd Expect" with the following text:

Python knows the usual control flow statements that other languages speak — `if`, `for`, `while` and `range` — with some of its own twists, of course. [More control flow tools in Python 3](#)

At the bottom of the main content area, there is a footer with a call to action: "Python is a programming language that lets you work quickly and integrate systems more effectively. [» Learn More](#)".

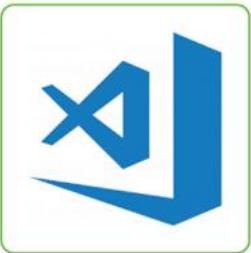
IDE Python

IGTI

PyCharm



Visual Studio Code



Sublime Text



Vim



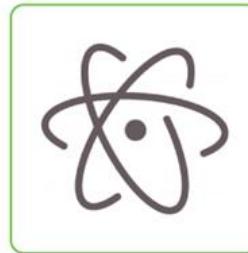
GNU Emacs



IDLE

Python's Integrated Development and Learning Environment

Atom



Spyder



JuPyter



Eclipse



Eric Python



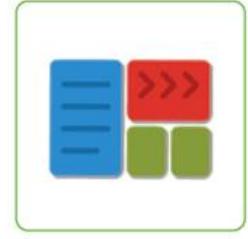
Wing



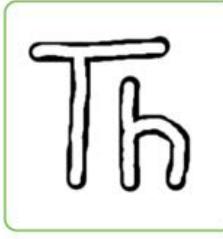
PyScripter



Pyzo



Thonny



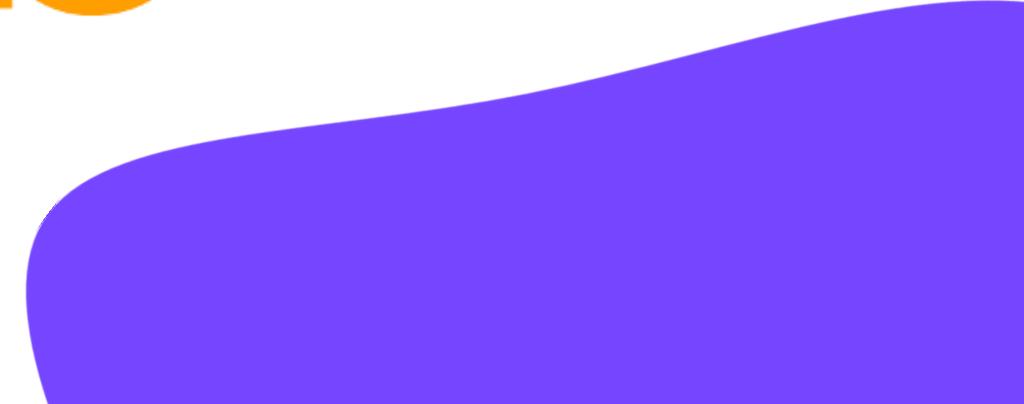
colab

Nosso primeiro programa



Olá, mundo!!!

colab



Conclusão



1. Instalação em diferentes plataformas.
2. IDE.
3. Olá, mundo!

Próxima aula



1. Conceitos fundamentais.



Fundamentos do Bootcamp

Desenvolvedor Python

Aula 1.3. Conceitos Fundamentais

Prof. Túlio Philipe Ferreira e Vieira

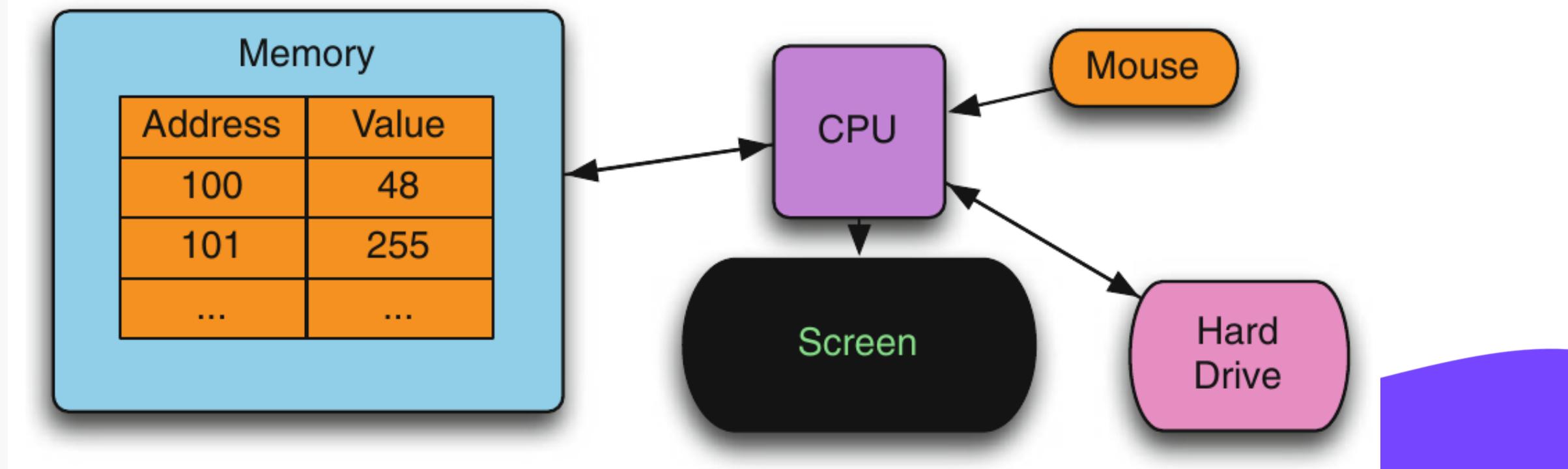
Nesta aula



1. Componentes do computador.
2. O que é uma linguagem de programação?
3. Erros.

Componentes de um computador

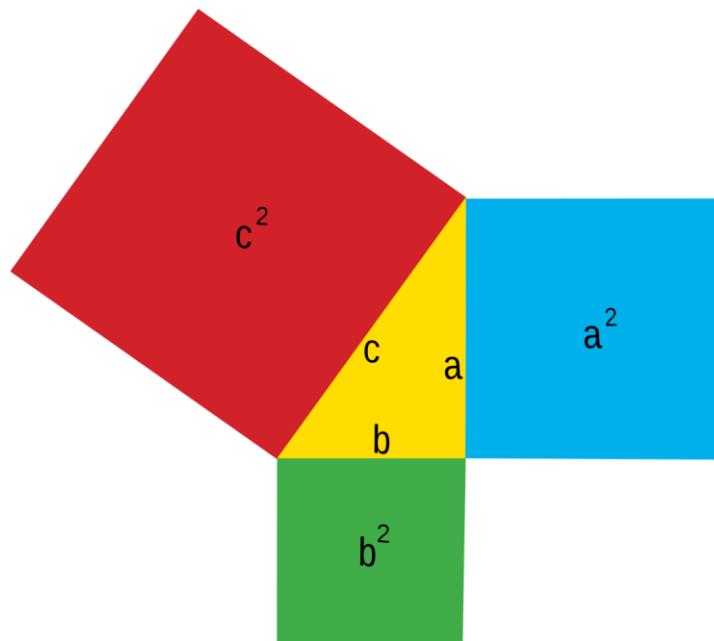
IGTi



Linguagens de programação



É um conjunto de ferramentas que nos permite construir programas em um nível maior de abstração que a escrita em uma sequência de 0's e 1's.



Interpretador da linguagem

A screenshot of a terminal window titled "tulio@tulioph: ~". The window shows the Python 3.7.3 interpreter starting up. The text displayed is:

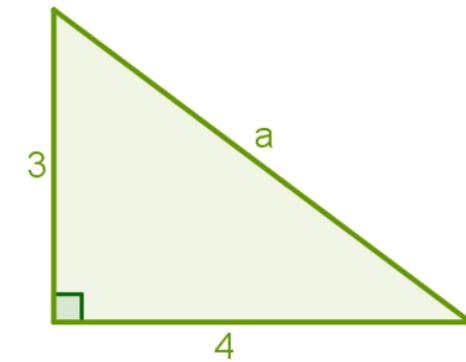
```
(base) tulio@tulioph:~$ python
Python 3.7.3 (default, Mar 27 2019, 22:11:17)
[GCC 7.3.0] :: Anaconda, Inc. on linux
Type "help", "copyright", "credits" or "license" for more information.
>>> █
```

The terminal has a dark background and light-colored text. The title bar includes standard window controls (minimize, maximize, close).

Interpretador da linguagem



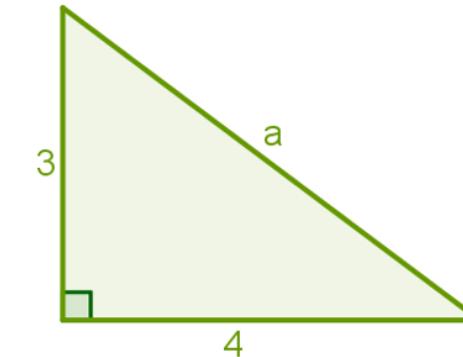
```
tulio@tulioph: ~
(base) tulio@tulioph:~$ python
Python 3.7.3 (default, Mar 27 2019, 22:11:17)
[GCC 7.3.0] :: Anaconda, Inc. on linux
Type "help", "copyright", "credits" or "license" for more information.
>>> b=3
>>> c=4
>>> a2=(b*b)+(c*c)
>>> a2
25
>>> a=a2**0.5
>>> a
5.0
>>> □
```



IDE - Integrated Development Environment



colab



Erros de sintaxe



Ocorrem antes do programa ser executado (run).

São mais fáceis de serem identificados.

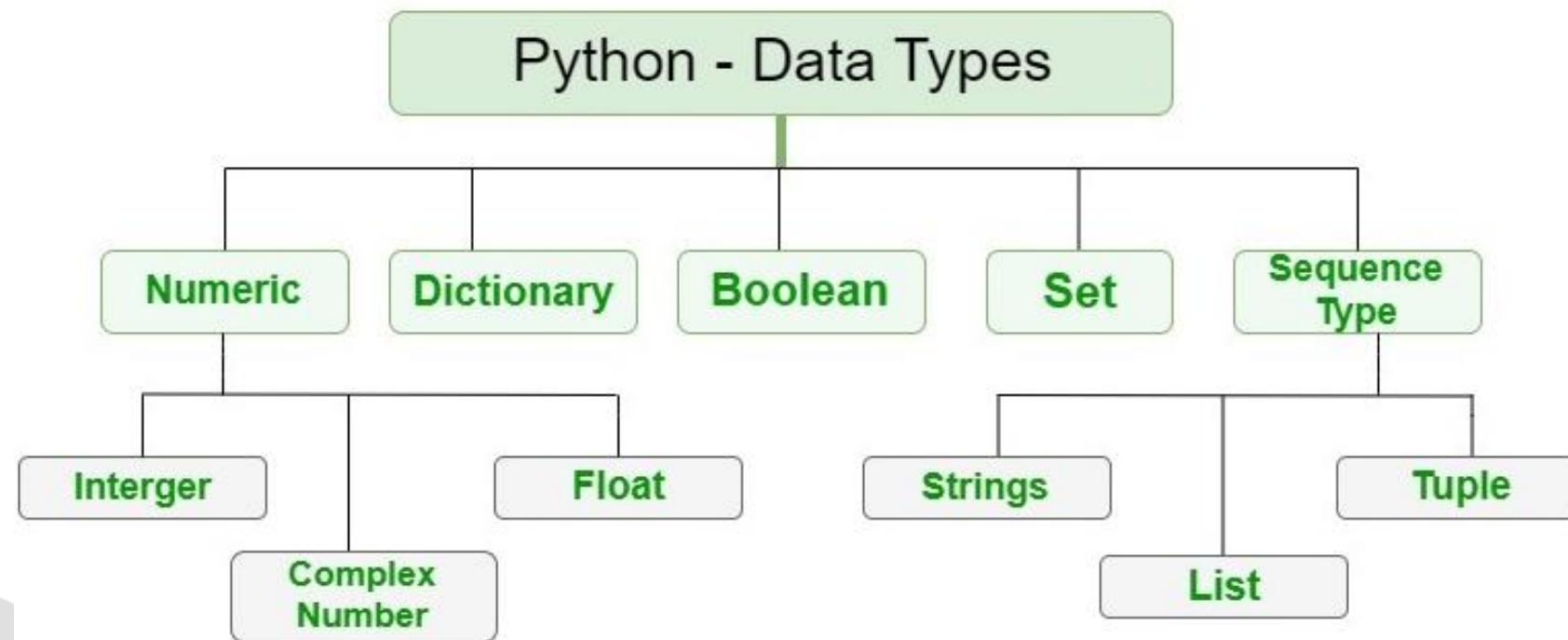
A screenshot of a terminal window titled "tulio@tulioph: ~". The window shows a Python session. The user has typed several commands: "b=3", "c=4", "a2=(b*b)+(c*c)", "a2", "a=a2**0.5", "a", and "print a". The last command, "print a", results in a syntax error: "SyntaxError: Missing parentheses in call to 'print'. Did you mean print(a)?". A red rectangular box highlights this error message.

```
(base) tulio@tulioph:~$ python
Python 3.7.3 (default, Mar 27 2019, 22:11:17)
[GCC 7.3.0] :: Anaconda, Inc. on linux
Type "help", "copyright", "credits" or "license" for more information.
>>> b=3
>>> c=4
>>> a2=(b*b)+(c*c)
>>> a2
25
>>> a=a2**0.5
>>> a
5.0
>>> print a
  File "<stdin>", line 1
    print a
      ^
SyntaxError: Missing parentheses in call to 'print'. Did you mean print(a)?
>>> 
```

Tipos de valores

IGTI

Todos os valores em python são objetos.



Fonte: [geeksforgeeks.org](https://www.geeksforgeeks.org/python-data-types/)

Conclusão



1. Componentes do computador.
2. O que é uma linguagem de programação.
3. Tipos de erros.

Próxima aula



1. Tipos de dados em python.



Fundamentos do Bootcamp

Desenvolvedor Python

Aula 1.4.1. Tipos de Dados em Python (Parte 1)

Prof. Túlio Philipe Ferreira e Vieira

Nesta aula



1. Atribuição de valores.
2. Garbage Collector.
3. Valores numéricos.

Conclusão



1. Atribuição de valores em Python.
2. Como a “limpeza de memória” funciona em Python.
3. Tipos de dados numéricos.

Próxima aula



1. Tipos de dados em Python (Parte 2).



Fundamentos do Bootcamp

Desenvolvedor Python

Aula 1.4.2. Tipos de Dados em Python (Parte 2)

Prof. Túlio Philipe Ferreira e Vieira

Nesta aula



1. Tipo de dados String (objeto String).

Conclusão



1. Objeto string e alguns métodos.

Próxima aula



1. Tipos de dados em Python (Parte 3)



Fundamentos do Bootcamp

Desenvolvedor Python

Aula 1.4.3. Tipos de Dados em Python (Parte 3)

Prof. Túlio Philipe Ferreira e Vieira

Nesta aula



1. Tipo de dados “Lista”.

Conclusão



1. Características do tipo de dados “Lista”.

Próxima aula



1. Tipos de dados em Python (Parte 4).



Fundamentos do Bootcamp

Desenvolvedor Python

Aula 1.4.4. Tipos de Dados em Python (Parte 4)

Prof. Túlio Philipe Ferreira e Vieira

Nesta aula



1. Tipo de dados “Tuplas”.
2. Tipos de dados “booleanos”.

Conclusão



1. Características do tipo de dados “Tuplas”.
2. Características do tipo de dados “booleanos”.

Próxima aula



1. Tipos de dados em Python (Parte 5).



Fundamentos do Bootcamp

Desenvolvedor Python

Aula 1.4.5. Tipos de Dados em Python (Parte 5)

Prof. Túlio Philipe Ferreira e Vieira

Nesta aula



1. Tipo de dados “Set”.

Conclusão



1. Características do tipo de dados “Set”.

Próxima aula



1. Tipos de dados em Python (Parte 6).



Fundamentos do Bootcamp

Desenvolvedor Python

Aula 1.4.6. Tipos de Dados em Python (Parte 6)

Prof. Túlio Philipe Ferreira e Vieira

Nesta aula



1. Tipo de dados “Dicionários”.
2. Entrada de dados.

Conclusão



1. Características do tipo de dados “Dicionários”.
2. Processo de entrada de dados.

Próxima aula



1. Decisões com Python.

Fundamentos do Bootcamp

Desenvolvedor Python

Capítulo 2. Decisões com Python

Prof. Túlio Philipe Ferreira e Vieira

Fundamentos do Bootcamp

Desenvolvedor Python

Aula 2.1. O condicional “if”

Prof. Túlio Philipe Ferreira e Vieira

Nesta aula



1. Funcionamento do condicional “if” em Python.

Conclusão



1. Como funciona o condicional “if” em Python.

Próxima aula



1. “If” aninhados.



Fundamentos do Bootcamp

Desenvolvedor Python

Aula 2.2. Trabalhando com “If” aninhados

Prof. Túlio Philipe Ferreira e Vieira

Nesta aula



1. Condições mais elaboradas com “if”.
2. “If” aninhados.

Conclusão



1. Decisões mais elaboradas.
2. “If” aninhados.

Próxima aula



1. “If”-elif-else.

Fundamentos do Bootcamp

Desenvolvedor Python

Aula 2.3. Trabalhando com if-elif-else

Prof. Túlio Philipe Ferreira e Vieira

Nesta aula



1. Como utilizamos o if-elif-else.

Conclusão



1. Quais as características do if-elif-else.

Próxima aula



1. Trabalhando com exceções.

Fundamentos do Bootcamp

Desenvolvedor Python

Aula 2.4. Trabalhando exceções

Prof. Túlio Philipe Ferreira e Vieira

Nesta aula



1. O que são e para que utilizar exceções.

Conclusão



1. Como utilizar exceções em Python.

Próxima aula



1. Estruturas de repetição.