

Introdução à Programação com Python

Agenda

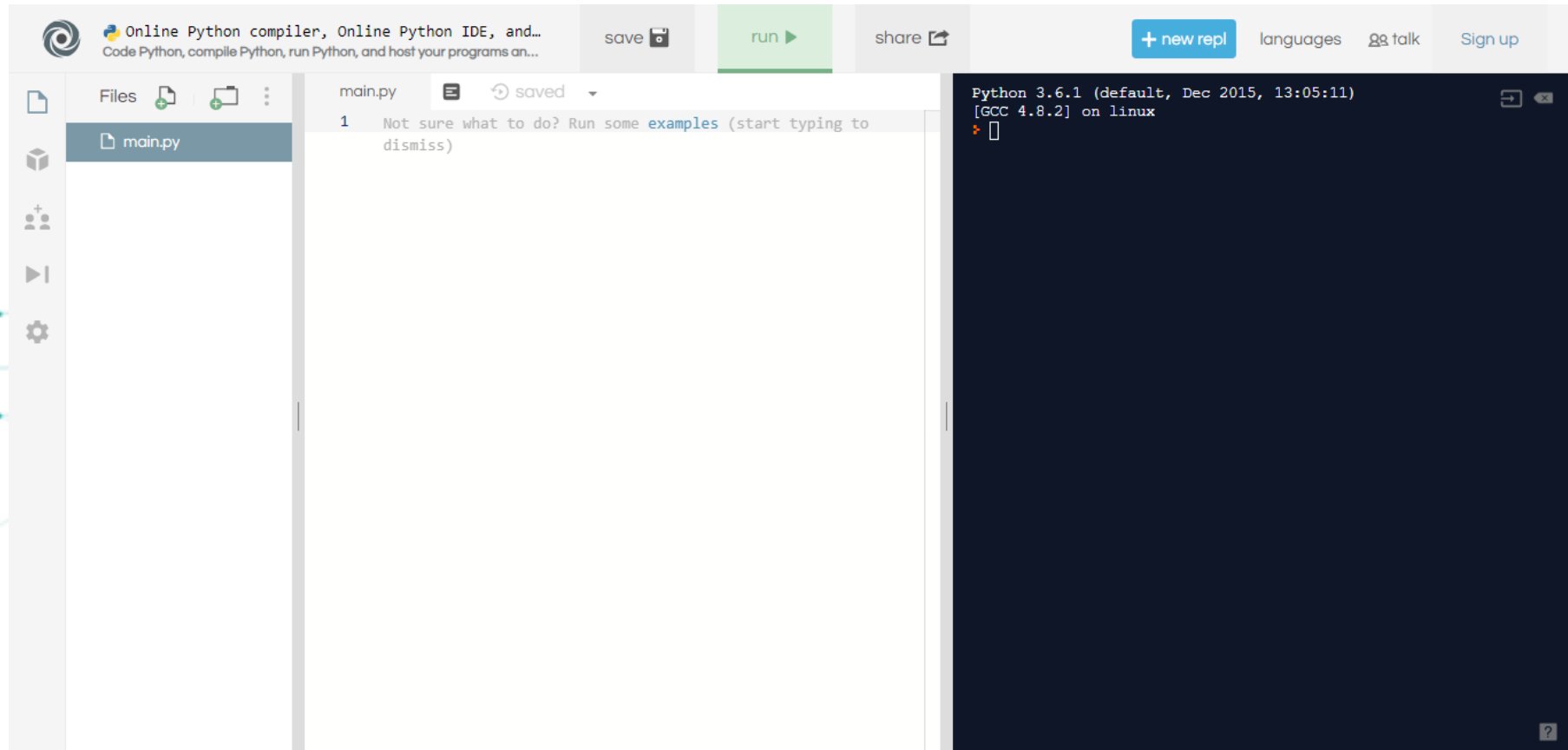
Tópicos:

- **Como Começar?**
- **Operadores Aritméticos e Tipos Numéricos**
- **Comentários**
- **Variáveis**
- **Textos**
- **Saída de Dados na Tela**
- **Entrada de Dados da Tela e Conversão de Tipos**
- **Funções Matemáticas e Módulos Externos**
- **Procurando Ajuda**

Como Começar?

repl.it/languages/python3

Executando na Web:



Como Começar?

www.python.org

Instalando no computador:

The screenshot shows the Python.org homepage. At the top, there's a navigation bar with links to Python, PSF, Docs, PyPI, Jobs, and Community. Below this is a search bar and a 'Donate' button. A secondary navigation bar contains links to About, Downloads, Documentation, Community, Success Stories, News, and Events. The 'Downloads' link is highlighted, and a dropdown menu is open, showing options like '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' page. On this page, the 'Python 3.7.3' button is highlighted with a red box. Below the button, there's a note stating 'Note that Python 3.5+ cannot be used on Windows XP or earlier.' and a link to 'View the full list of downloads.' At the bottom of the page, there's a statement: 'Python is a programming language that lets you work quickly and integrate systems more effectively. >>> [Learn More](#)'.

Python

PSF

Docs

PyPI

Jobs

Community

python™

Donate

Search

GO

Socialize

About

Downloads

Documentation

Community

Success Stories

News

Events

All releases

Source code

Windows

Mac OS X

Other Platforms

License

Alternative Implementations

Download for Windows

Python 3.7.3

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

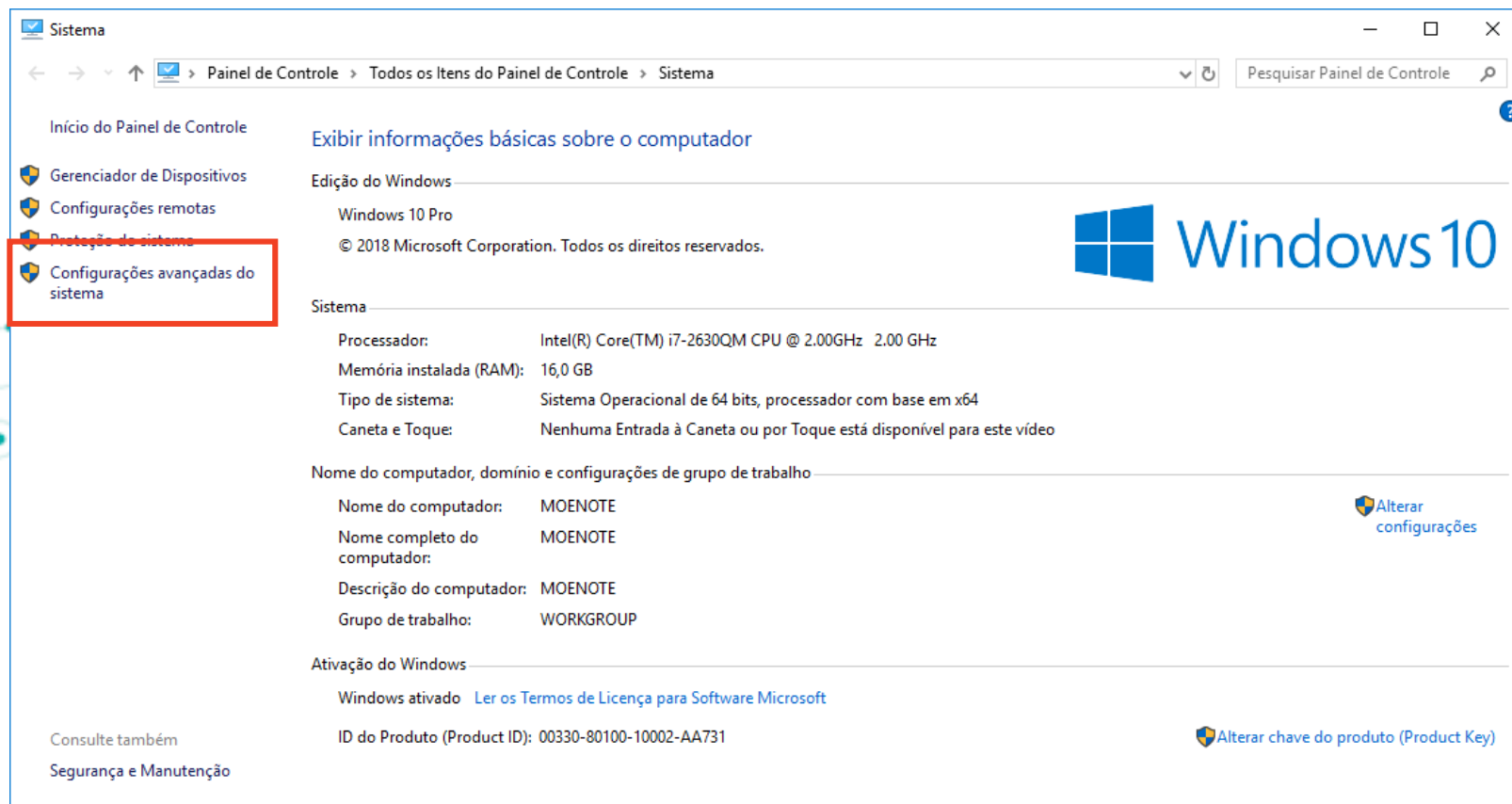
Not the OS you are looking for? Python can be used on many operating systems and environments.

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

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

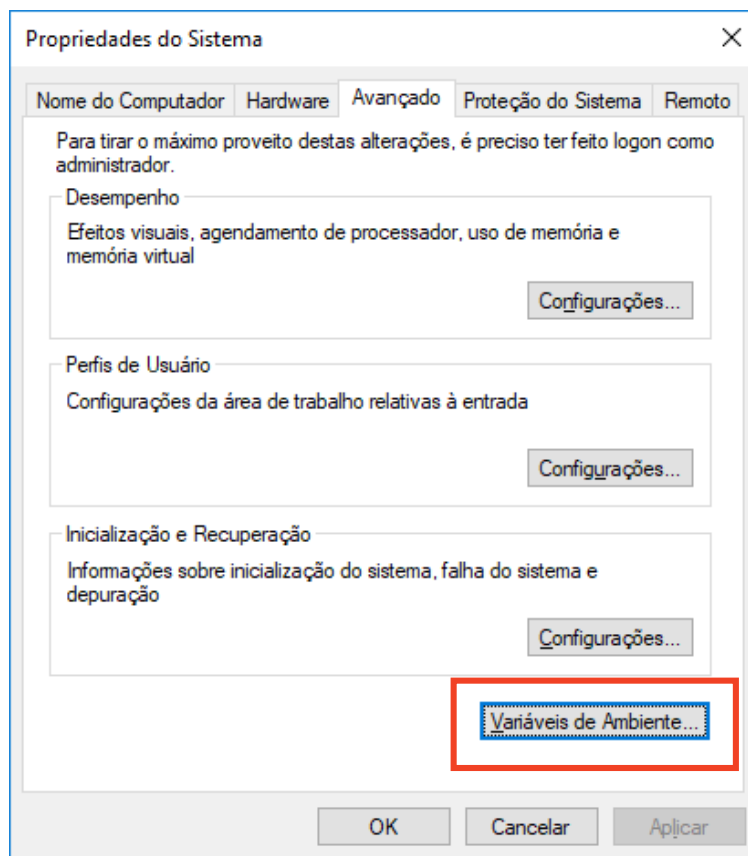
Como Começar?

Instalando no computador:



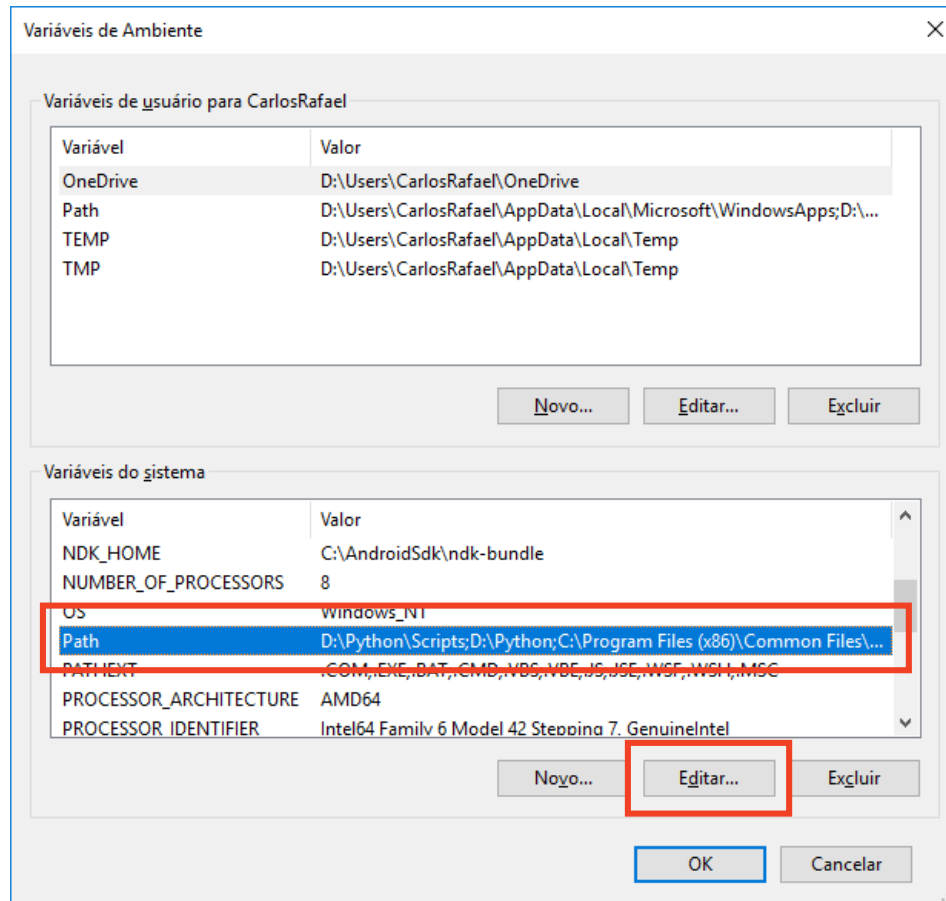
Como Começar?

Instalando no computador:



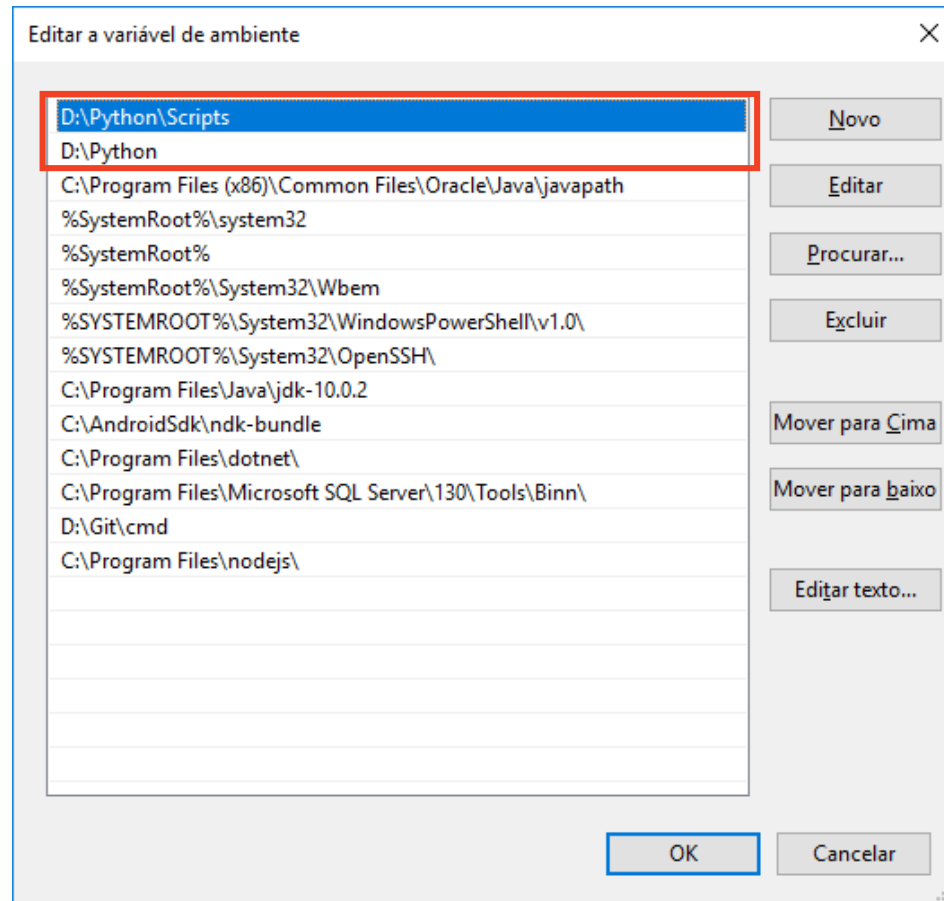
Como Começar?

Instalando no computador:



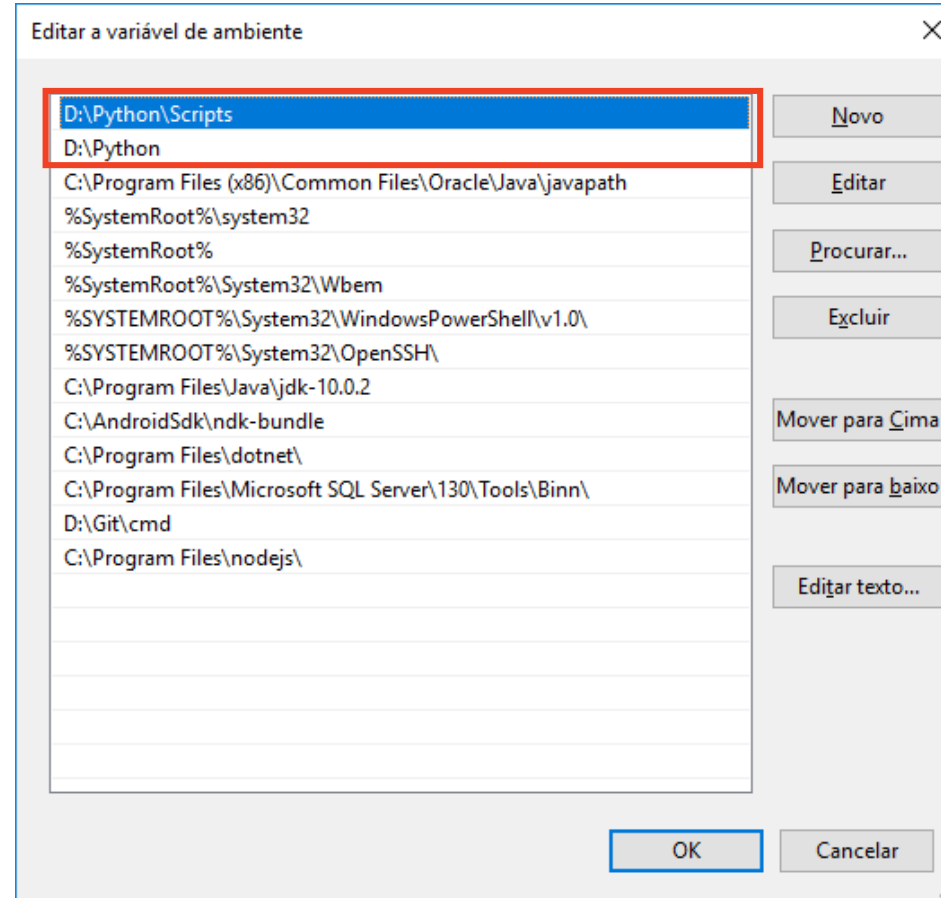
Como Começar?

Instalando no computador:



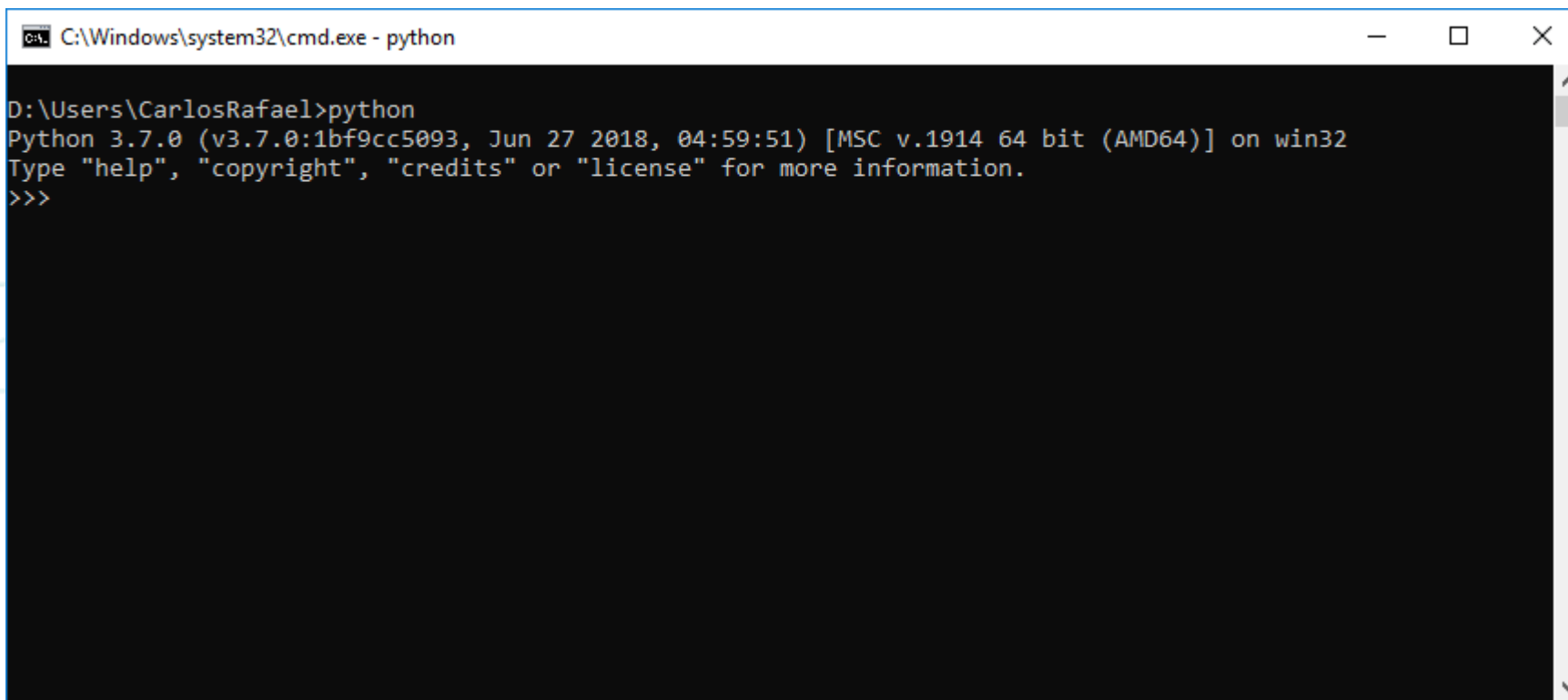
Como Começar?

Instalando no computador:



Como Começar?

Executando via linha de comando:



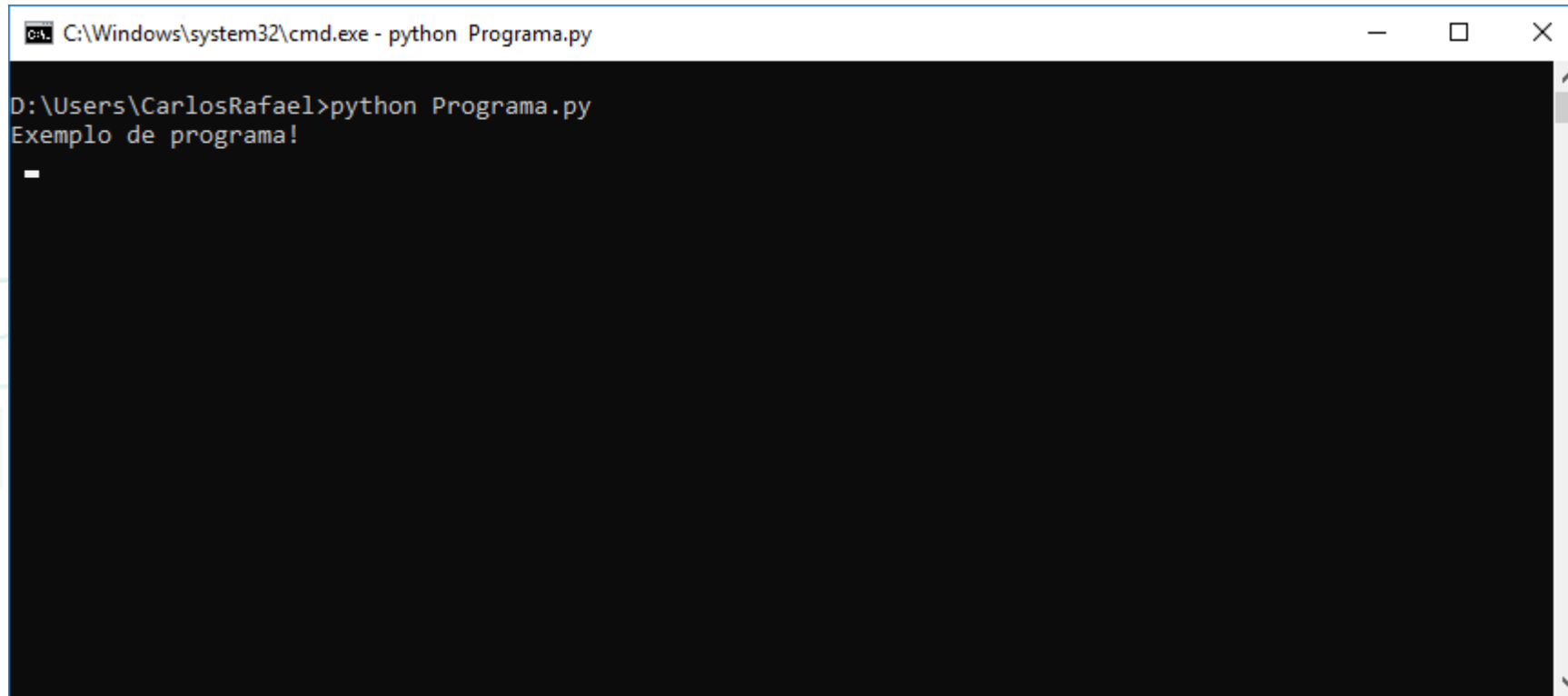
```
C:\Windows\system32\cmd.exe - python

D:\Users\CarlosRafael>python
Python 3.7.0 (v3.7.0:1bf9cc5093, Jun 27 2018, 04:59:51) [MSC v.1914 64 bit (AMD64)] on win32
Type "help", "copyright", "credits" or "license" for more information.
>>>
```

Digite exit() para sair

Como Começar?

Executando via linha de comando (Código existente):



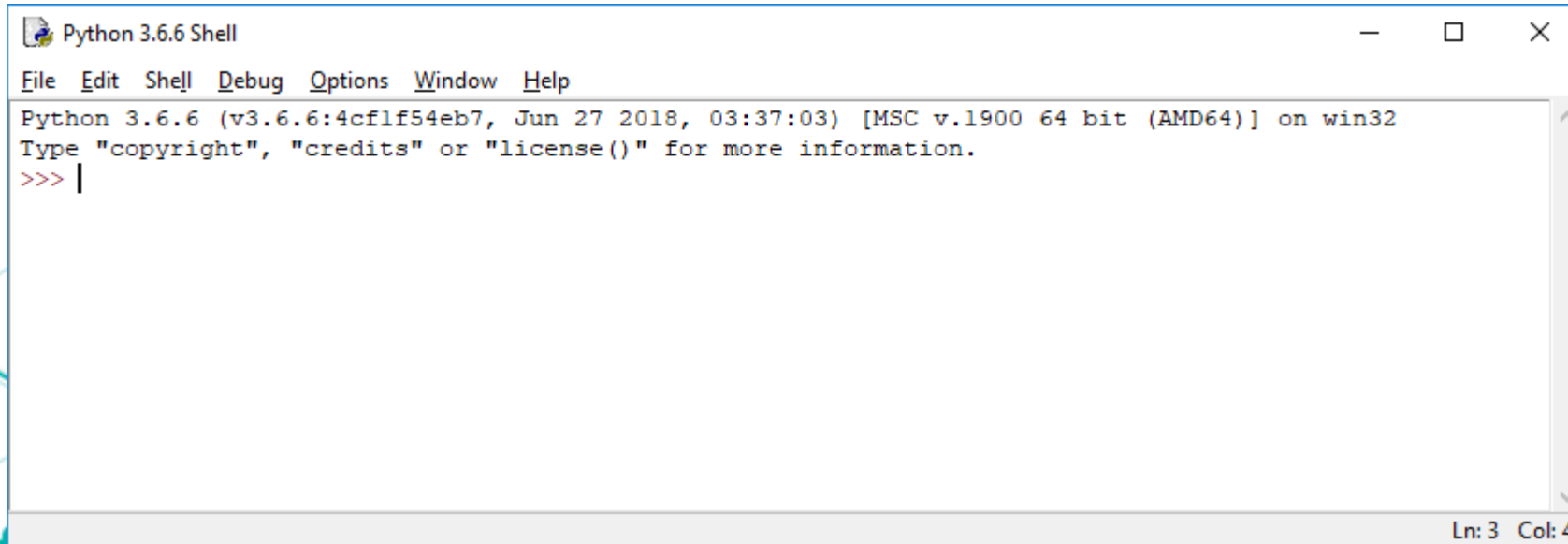
A screenshot of a Windows command prompt window. The title bar reads "C:\Windows\system32\cmd.exe - python Programa.py". The command prompt shows the user "D:\Users\CarlosRafael" typing the command "python Programa.py". The output of the script is "Exemplo de programa!" followed by a blank line. The window has standard Windows window controls (minimize, maximize, close) in the top right corner.

```
C:\Windows\system32\cmd.exe - python Programa.py

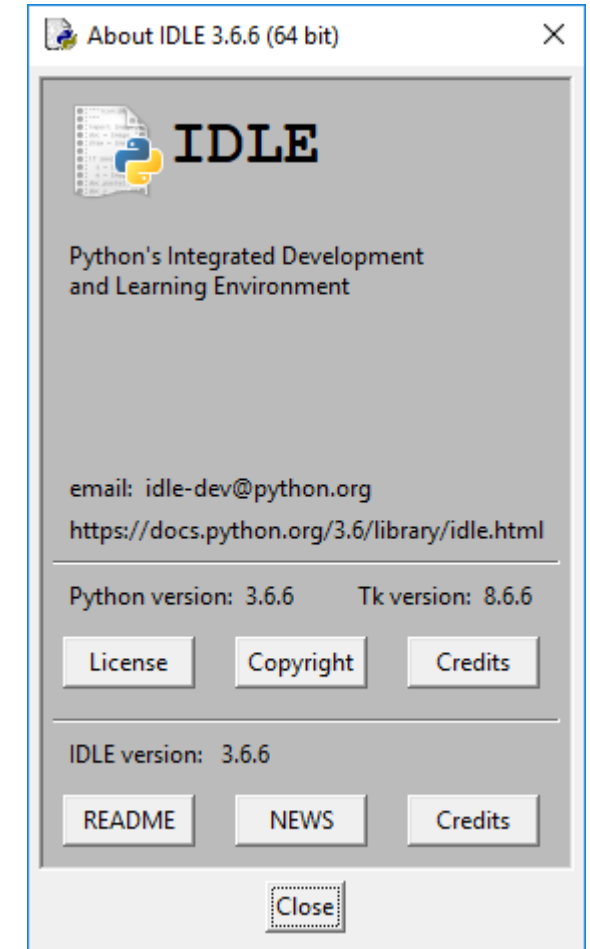
D:\Users\CarlosRafael>python Programa.py
Exemplo de programa!
_
```

Como Começar?

Utilizando a IDE IDLE para escrever e executar códigos:




A screenshot of the Python 3.6.6 Shell window. The title bar reads "Python 3.6.6 Shell". The menu bar includes "File", "Edit", "Shell", "Debug", "Options", "Window", and "Help". The main text area displays the following information: "Python 3.6.6 (v3.6.6:4cflf54eb7, Jun 27 2018, 03:37:03) [MSC v.1900 64 bit (AMD64)] on win32", "Type 'copyright', 'credits' or 'license()' for more information.", and a prompt ">>> |". The status bar at the bottom right shows "Ln: 3 Col: 4".




Como Começar?

visualstudio.microsoft.com/downloads

Instalando e utilizando a IDE Visual Studio Community:



Downloads



Visual Studio 2019
Ambiente de desenvolvimento integrado (IDE) completo para Android, iOS, Windows, Web e nuvem

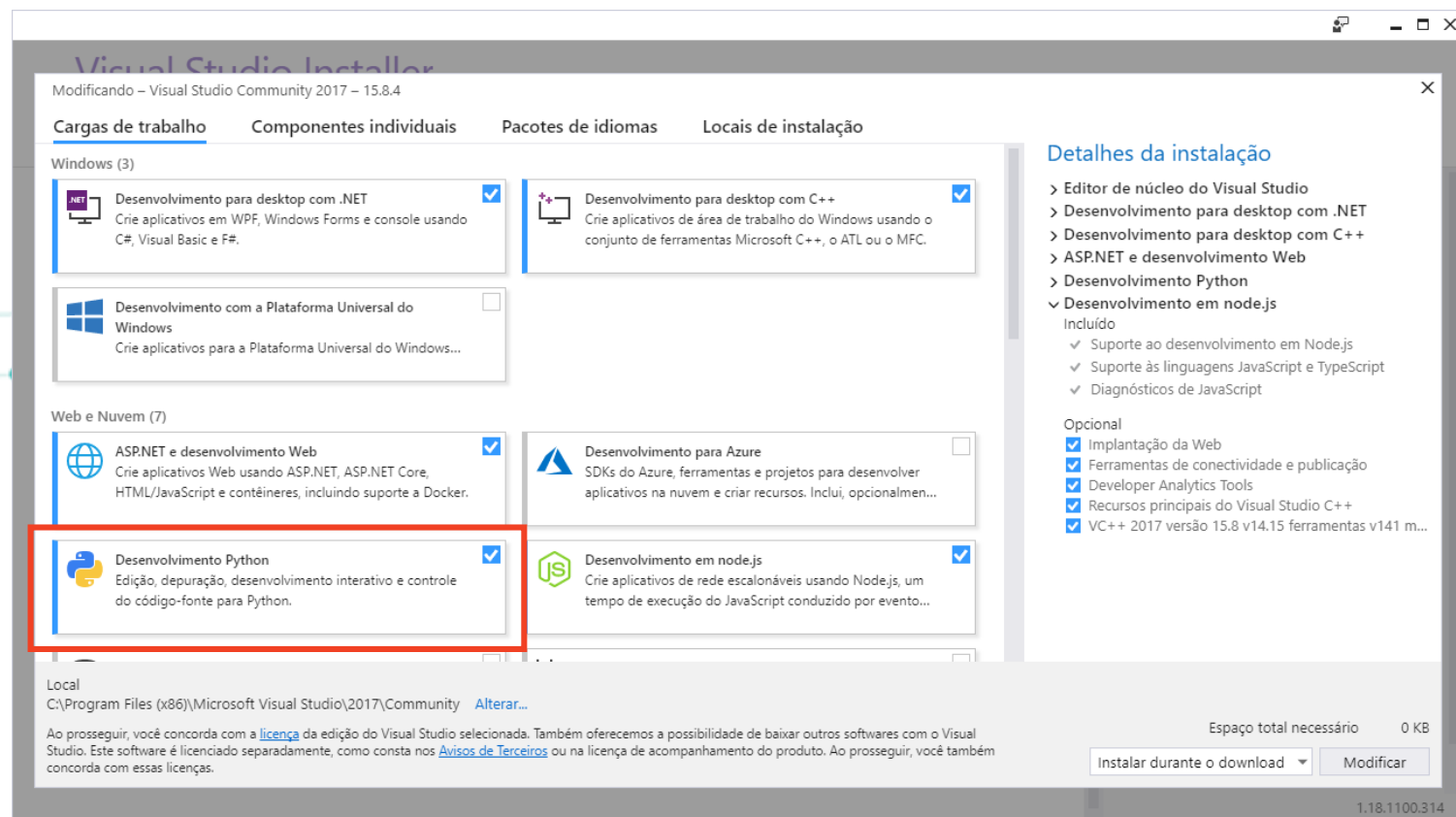
Versão: 16.
[Notas de versão](#)

[Compare as edições](#)
[Como instalar offline](#)

| Community | Professional | Enterprise |
|--|--|--|
| IDE avançado, gratuito para estudantes, colaboradores de software livre e indivíduos | IDE profissional mais indicado para equipes pequenas | Solução ponta a ponta escalonável para equipes de qualquer tamanho |
| Download gratuito ↓ | Avaliação gratuita ↓ | Avaliação gratuita ↓ |
| Baixar versão prévia > | Baixar versão prévia > | Baixar versão prévia > |

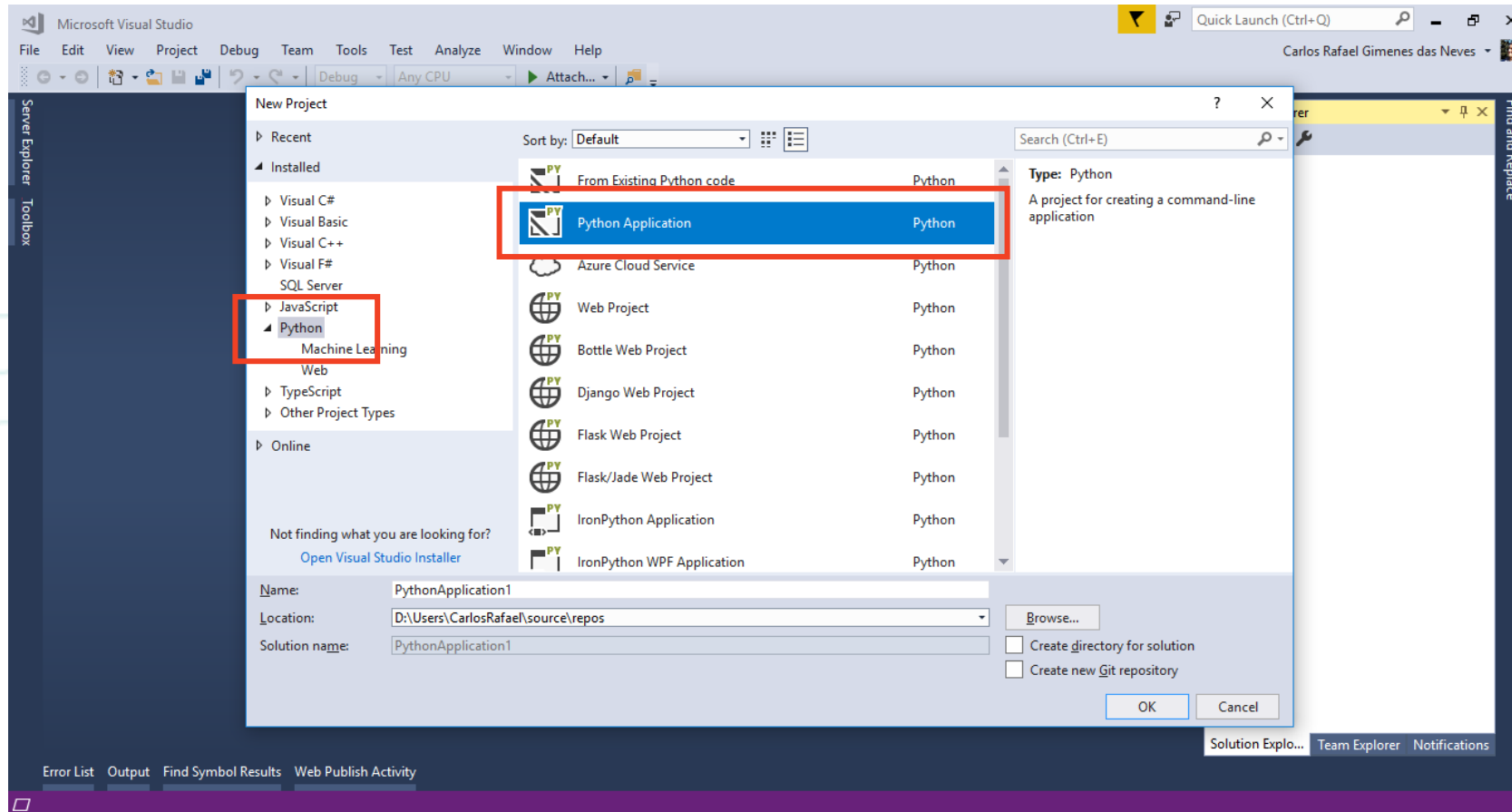
Como Começar?

Instalando e utilizando a IDE Visual Studio Community:



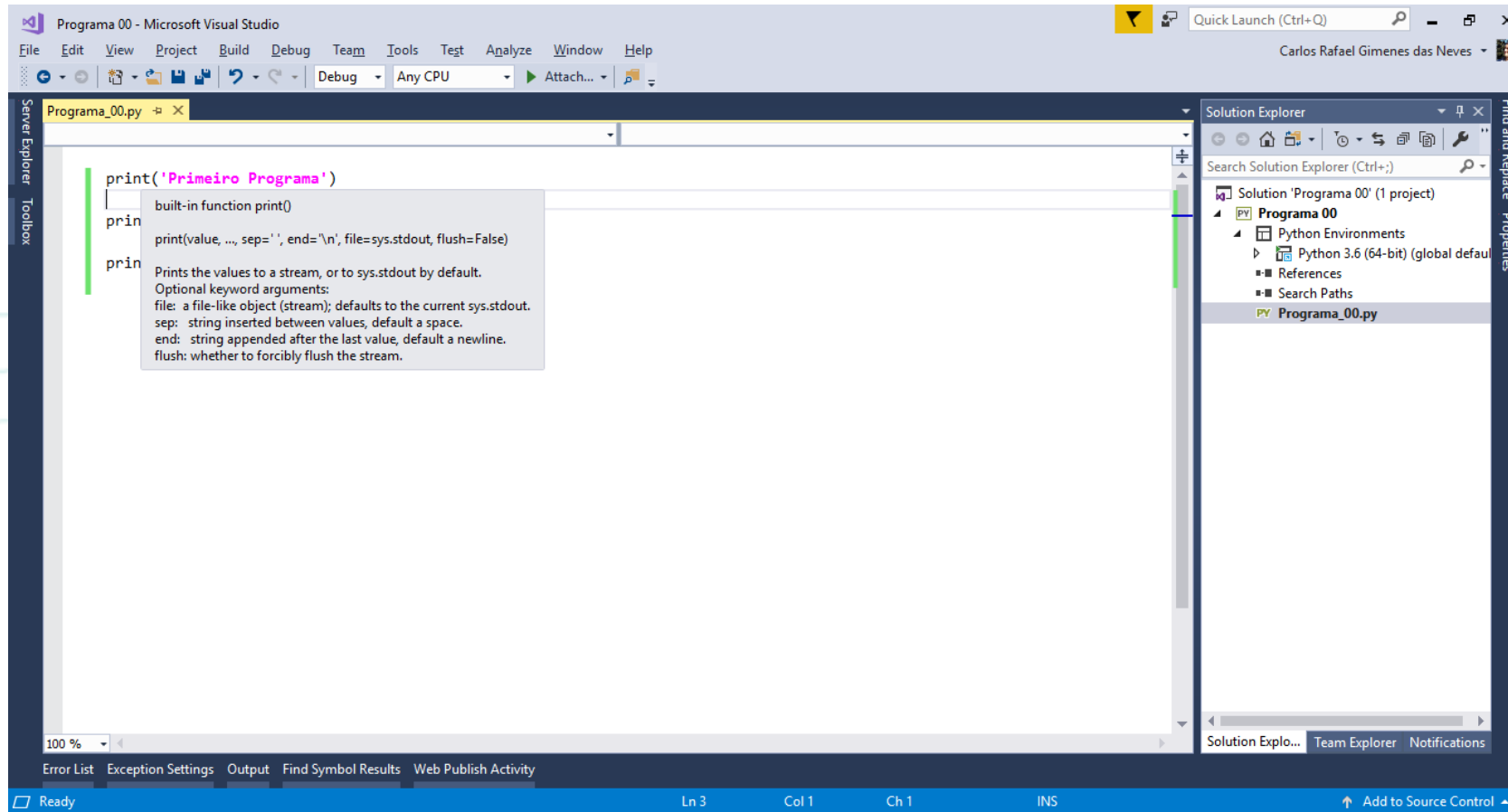
Como Começar?

Instalando e utilizando a IDE Visual Studio Community:



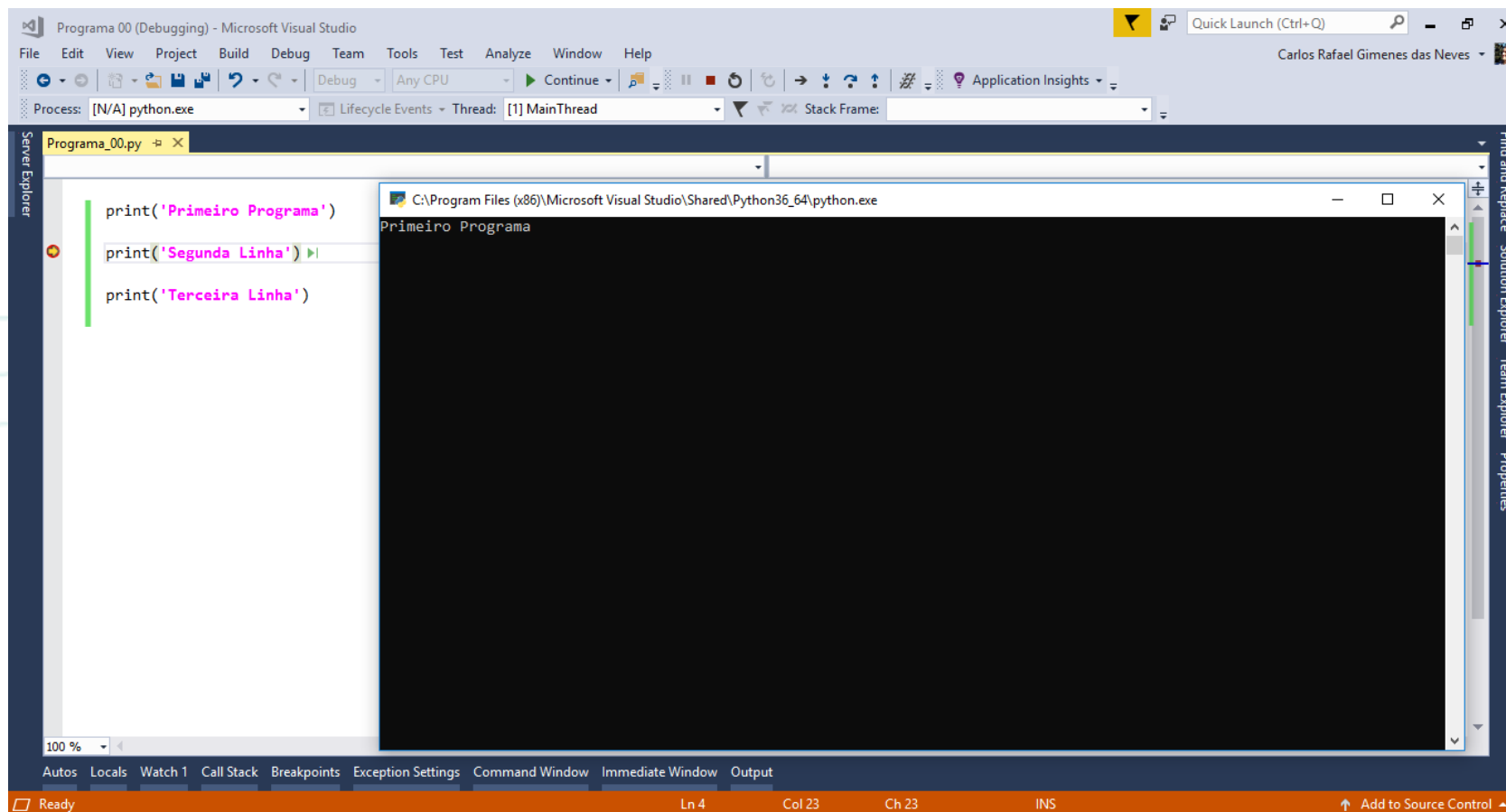
Como Começar?

Instalando e utilizando a IDE Visual Studio Community:



Como Começar?

Instalando e utilizando a IDE Visual Studio Community:



**Tecle F5
para
executar, e
F9 para
pedir para
parar em
uma linha**

Operadores Aritméticos e Tipos Numéricos

Soma, Subtração, Multiplicação, Divisão e Potenciação

+ - * / **

Precedência (igual ao ensinado na escola)

```
>>> 2 * 5 ** 2
```

```
>>> 9 + 1 * 5
```

Parênteses (igual ao ensinado na escola)

```
>>> (9 + 1) * 5
```

Operadores Aritméticos e Tipos Numéricos

```
>>> type(1)  
<class 'int'>
```

Inteiro

```
>>> type(1.5)  
<class 'float'>
```

Real

Operadores Aritméticos e Tipos Numéricos

```
>>> type(1 + 2)  
<class 'int'>
```

Inteiro

```
>>> type(1.5 + 2)  
<class 'float'>
```

Real

Operadores Aritméticos e Tipos Numéricos

Divisão é um caso especial, um pouco diferente de outras linguagens...

```
>>> 999 / 100
```

```
9.99
```

```
>>> type(999 / 100)
```

```
<class 'float'>
```

Real

Operadores Aritméticos e Tipos Numéricos

Divisão é um caso especial, um pouco diferente de outras linguagens...

```
>>> 999 // 100
```

```
9
```

```
>>> type(999 // 100)
```

```
<class 'int'>
```

Inteiro

Operadores Aritméticos e Tipos Numéricos

Divisão é um caso especial, um pouco diferente de outras linguagens...

```
>>> 999.0 // 100
```

```
9.0
```

```
>>> type(999.0 // 100)
```

```
<class 'float'>
```

Real

Operadores Aritméticos e Tipos Numéricos

Divisão é um caso especial, um pouco diferente de outras linguagens...

```
>>> 999.0 // 100
```

```
9.0
```

```
>>> type(999.0 // 100)
```

```
<class 'float'>
```

Real



Operadores Aritméticos e Tipos Numéricos

Divisão é um caso especial, um pouco diferente de outras linguagens...

```
>>> 999.0 // 100
```

```
9.0
```

```
>>> type(999.0 // 100)
```

```
<class 'float'>
```

Real



Operadores Aritméticos e Tipos Numéricos

Por falar em divisão, existe um operador conhecido como “Resto de divisão” (modulo em inglês)



```
>>> 14 % 5
```

```
4
```

```
>>> 16 % 3
```

```
1
```

```
>>> 15 % 3
```

```
0
```

$$\begin{array}{r|l} 16 & 3 \\ \hline 1 & 5 \end{array}$$

Operadores Aritméticos e Tipos Numéricos

Por falar em divisão, existe um operador conhecido como “Resto de divisão” (modulo em inglês)

```
>>> 14 % 5
```

```
4
```

```
>>> 16 % 3
```

```
1
```

```
>>> 15 % 3
```

```
0
```

$$\begin{array}{r} 16 \overline{) 3} \\ \underline{15} \\ 1 \end{array}$$



Operadores Aritméticos e Tipos Numéricos

Para aqueles mais “matematicamente inclinados”, o Python oferece suporte a números complexos...

docs.python.org/3/library/cmath.html

```
>>> type(4 + 2j)
<class 'complex'>
>>> abs(4 + 2j)
4.47213595499958
```



Comentários

Textos que não são interpretados (servem para explicar o programa)

```
# Qualquer coisa depois de um # será  
# ignorada! O texto aqui é livre!
```

Para quem já programa: Python não tem comentários em bloco... (como /* ... */)

... Mas existe uma “gambiarrinha” para simular isso (como veremos depois...)

Variáveis



Variáveis

```
>>> a = 9
```

```
>>> a
```

```
9
```

```
>>> type(a)
```

```
<class 'int'>
```

```
>>> a + 1
```

```
10
```

```
>>> a
```

```
>>> ???
```

Variáveis

```
>>> a = 9
>>> a
9
>>> type(a)
<class 'int'>
>>> a + 1
10
>>> a
9
>>> 9
```

**a + 1 não altera
o valor de a!**

Variáveis

```
>>> a = 9
```

```
>>> a
```

```
9
```

```
>>> a = a + 1
```

```
>>> a
```

```
10
```

Variáveis

```
>>> a = b + 1
```

```
Traceback (most recent call last):
```

```
  File "<stdin>", line 1, in <module>
```

```
NameError: name 'b' is not defined
```

Variáveis

```
>>> b = 2  
>>> a = b + 1  
>>> a  
3
```

Variáveis

Quando for nomear uma variável, é proibido utilizar os nomes da lista de nomes reservados da linguagem:

docs.python.org/3/reference/lexical_analysis.html#keywords



Textos

```
>>> a = 'ESPM'
```

```
>>> a
```

```
'ESPM'
```

```
>>> a = "ESPM"
```

```
>>> a
```

```
'ESPM'
```

Textos

```
>>> a = 'ESPM'
```

```
>>> a
```

```
'ESPM'
```

```
>>> a = "ESPM"
```

```
>>> a
```

```
'ESPM'
```

Abaixo de 30
anos de idade

30 anos de idade
ou mais



Textos

```
>>> a = 'TECH'
>>> b = 'ESPM'
>>> c = a + b
>>> c
'TECHESPM'
>>> len(a)
4
>>> len(c)
8
```

Textos

```
>>> a = 'A variável b vale '  
>>> b = 8  
>>> c = a + str(b)  
>>> c  
'A variável b vale 8'
```


Textos

```
>>> a = 'A variável b vale '  
>>> b = 8  
>>> c = a + str(b) + ', e eu gosto disso!'  
>>> c  
'A variável b vale 8, e eu gosto disso!'
```

Textos

```
>>> b = 8  
>>> c = f'A variável b vale {b}, e eu gosto disso!'  
>>> c  
'A variável b vale 8, e eu gosto disso!'
```



Textos

Ainda outras formas de fazer a mesma coisa...
(Não precisa se preocupar!)

```
>>> b = 8  
>>> c = 'A variável b vale {}'.format(b)  
>>> c  
'A variável b vale 8'
```



Textos

Ainda outras formas de fazer a mesma coisa...
(Não precisa se preocupar!)

```
>>> b = 8  
>>> c = 'A variável b vale {b}'.format(b=b)  
>>> c  
'A variável b vale 8'
```



Textos

Ainda outras formas de fazer a mesma coisa...
(Não precisa se preocupar!)

```
>>> b = 8  
>>> c = 'A variável b vale %i' % b  
>>> c  
'A variável b vale 8'
```



Textos

```
>>> b = 8.1
>>> c = f'A variável b vale {b:.3f}'
>>> c
'A variável b vale 8.100'
```

Textos

```
>>> b = 8.29
>>> c = f'A variável b vale {b:.1f}'
>>> c
'A variável b vale 8.3'
```

Para mais exemplos de formatação:
docs.python.org/3/library/string.html#format-examples

Textos

Para quebras de linha:

```
>>> a = 'Uma linha\nOutra linha'
```

```
>>> a
```

```
'Uma linha\nOutra linha'
```

```
>>> print(a)
```

```
Uma linha
```

```
Outra linha
```


Textos

Para quebras de linha:

```
>>> a = '''Uma linha  
Outra linha'''  
>>> a  
'Uma linha\nOutra linha'  
>>> print(a)  
Uma linha  
Outra linha
```

Textos

Para mais informações sobre textos e caracteres de escape, como o `\n`:

docs.python.org/3/reference/lexical_analysis.html#literals

Textos

E, conforme o criador do Python, essa é a “gambiarrinha” para fazer comentários com várias linhas...

```
'''a = 5  
b = 18  
c = a + b'''
```



twitter.com/gvanrossum/status/112670605505077248

Saída de Dados na Tela

```
>>> print('Olá, classe!')  
Olá, classe!
```



Saída de Dados na Tela

```
>>> b = 8  
>>> print(b)  
8
```



Saída de Dados na Tela

```
>>> b = 8  
>>> print(f'A variável b vale {b}')
```

A variável b vale 8



Saída de Dados na Tela

```
>>> b = 8  
>>> c = f'A variável b vale {b}'  
>>> print(c)  
A variável b vale 8
```

Entrada de Dados da Tela e Conversão de Tipos

```
>>> a = input('Digite um valor para a: ')
```

```
Digite um valor para a: 10
```

```
>>> a
```

```
'10'
```

```
>>> type(a)
```

```
<class 'str'>
```

**Perceba que a é
um texto!**

Entrada de Dados da Tela e Conversão de Tipos

```
>>> a = input('Digite um valor para a: ')
Digite um valor para a: 10
>>> a = int(a)
>>> a
10
>>> type(a)
<class 'int'>
```

**Agora a é um
número inteiro!**

Entrada de Dados da Tela e Conversão de Tipos

```
>>> a = input('Digite um valor para a: ')
```

```
Digite um valor para a: 10.5
```

```
>>> a = int(a)
```

```
Traceback (most recent call last):
```

```
  File "<stdin>", line 1, in <module>
```

```
ValueError: invalid literal for int() with base 10: '10.5'
```

**Cuidado com
números reais!**

Entrada de Dados da Tela e Conversão de Tipos

```
>>> a = input('Digite um valor para a: ')
Digite um valor para a: 10.5
>>> a = float(a)
>>> a
10.5
>>> type(a)
<class 'float'>
```

**Agora a é um
número real!**

Entrada de Dados da Tela e Conversão de Tipos

```
>>> a = input()
```

```
10
```

```
>>> a
```

```
'10'
```

**Você também pode utilizar input() sem uma mensagem...
Mas fica meio sem graça!**





Será que vai ficar só nisso...?!



Não vai ficar só nisso...

Vamos praticar!!!

Funções Matemáticas e Módulos Externos

Um dos grandes trunfos do Python é sua coleção de bibliotecas e módulos já prontos


Para utilizar um módulo ou uma função, basta importar:

```
>>> import math  
>>> math.sqrt(10)  
3.1622776601683795
```

Funções Matemáticas e Módulos Externos

A importação pode ocorrer de diversas formas...

Diretamente (como já visto):



```
import math  
a = math.sqrt(10)
```


Funções Matemáticas e Módulos Externos

A importação pode ocorrer de diversas formas...


Através de um apelido, ou alias (útil para encurtar nomes muito longos):

```
import math as mat  
a = mat.sqrt(10)
```

Funções Matemáticas e Módulos Externos

A importação pode ocorrer de diversas formas...

Importando apenas as funções que serão utilizadas:



```
from math import sqrt  
a = sqrt(10)
```

Procurando Ajuda

[**www.python.org/doc**](http://www.python.org/doc)

[**docs.python.org/3/reference**](http://docs.python.org/3/reference)

[**docs.python.org/release/3.7.3/reference**](http://docs.python.org/release/3.7.3/reference)

[**docs.python.org/3/library**](http://docs.python.org/3/library)

[**docs.python.org/release/3.7.3/library**](http://docs.python.org/release/3.7.3/library)

[**docs.python.org/3/tutorial/introduction.html**](http://docs.python.org/3/tutorial/introduction.html)

[**stackoverflow.com**](http://stackoverflow.com)

[**pt.stackoverflow.com**](http://pt.stackoverflow.com)



Ficamos por aqui!