

```
>>> print('Olá, mundo!')
```

```
>>> print('Olá, mundo!')
```

```
|  
|_ Olá, mundo!
```

```
>>> input('Nome: ')
```

```
>>> input('Nome: ')
```

```
|  
|_ 'Jackson Osvaldo'
```

Introdução à Linguagem de Programação Python

Jackson Osvaldo da Silva Braga
Graduando em EAER (UFRA)

Capanema-PA
2018

Sobre Mim

- Graduando em Engenharia Ambiental e Energias Renováveis;
- Entusiasta do Software Livre;
- Apaixonado por Programação;
- Autor do PyEst;
- Computação Aplicada, Redes Neurais Artificiais, IA e *Machine Learning*.

Tópicos do Minicurso

- 1 - Histórico da Linguagem;
- 2 - Instalação da Linguagem;
 - A) Em Ambiente Windows;
 - B) Em Ambiente Linux;
- 3 - Operações Básicas
- 4 - Variáveis e Entradas de Dados
- 4 - Tipos de Dados;

Tópicos do Minicurso

5 - *if, else e elif*;

6 - *for, while*;

7 - Arquivos;

8 - Estatística;

A) *Pandas*;

B) *Numpy*;

C) *Matplotlib*;

D) *Scipy*.

9 - Aplicações Python: Web e Desktop;

10 - Considerações Finais.

Histórico da Linguagem

Histórico da Linguagem

- A Linguagem Python foi concebida no fim dos anos 80
- **Guido Van Rossum** (<3), CWI e ABC;
- Algol 68, Tunembau, *C* e *Shell Script*;
- "[...] havia necessidade de uma linguagem que 'preencheria o vazio entre *C* e o *shell*'. Por um tempo longo, esse foi o principal objetivo do Python" (Guido Van Rossum).

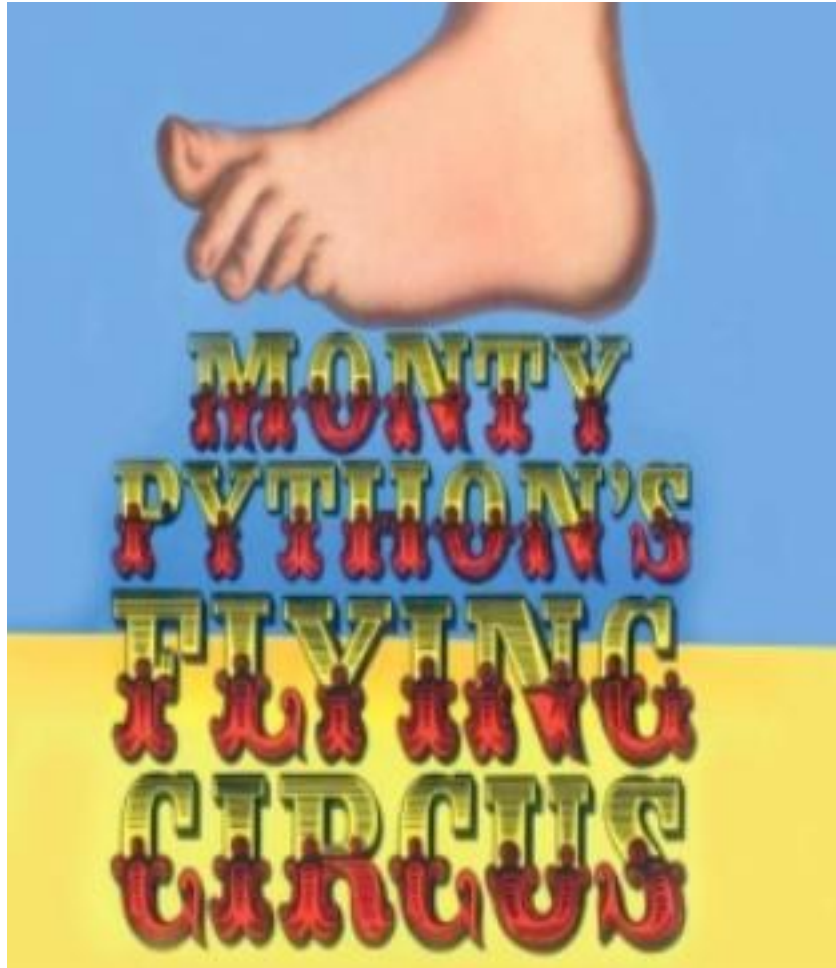
Histórico da Linguagem

- *Em 1989 o desenvolvimento do Python realmente teve início, nos primeiros meses de 1990 o autor já possuía uma versão mínima e operacional, pelo fim do ano de 1990 Python já era mais utilizada no CWI que a própria linguagem ABC.*
- *"ABC me deu a inspiração crucial para Python, o grupo Amoeba a motivação imediata e o grupo de multimídia fomentou seu crescimento" (Guido Van Rossum).*

Histórico da Linguagem

- *"No dia **20 de Fevereiro** [de 1991], foi lançada a primeira versão do Python, então denominada de **v0.9.0**. O anúncio foi feito no grupo de discussão **(newsgroup) alt.sources** (<http://www.faqs.org/faqs/alt-sources-intro/>)".*
- Licença MIT;

Histórico da Linguagem



- No início de seu projeto, Guido sabia que não queria siglas ou um nome fraco, como ABC;
- Foi então que Guido usou a primeira coisa que veio a sua cabeça: Monty Python's Flying Circus;
- O'Reilly e o primeiro livro da linguagem, o "Programming Python".

Histórico da Linguagem



Histórico da Linguagem

- Open Source, Free Software Foundation, Python Software Foundation;
- Comunidade Python;
- Comunidade Python Brasil;
- Comunidade Python Norte;
- PyCon.

Histórico da Linguagem

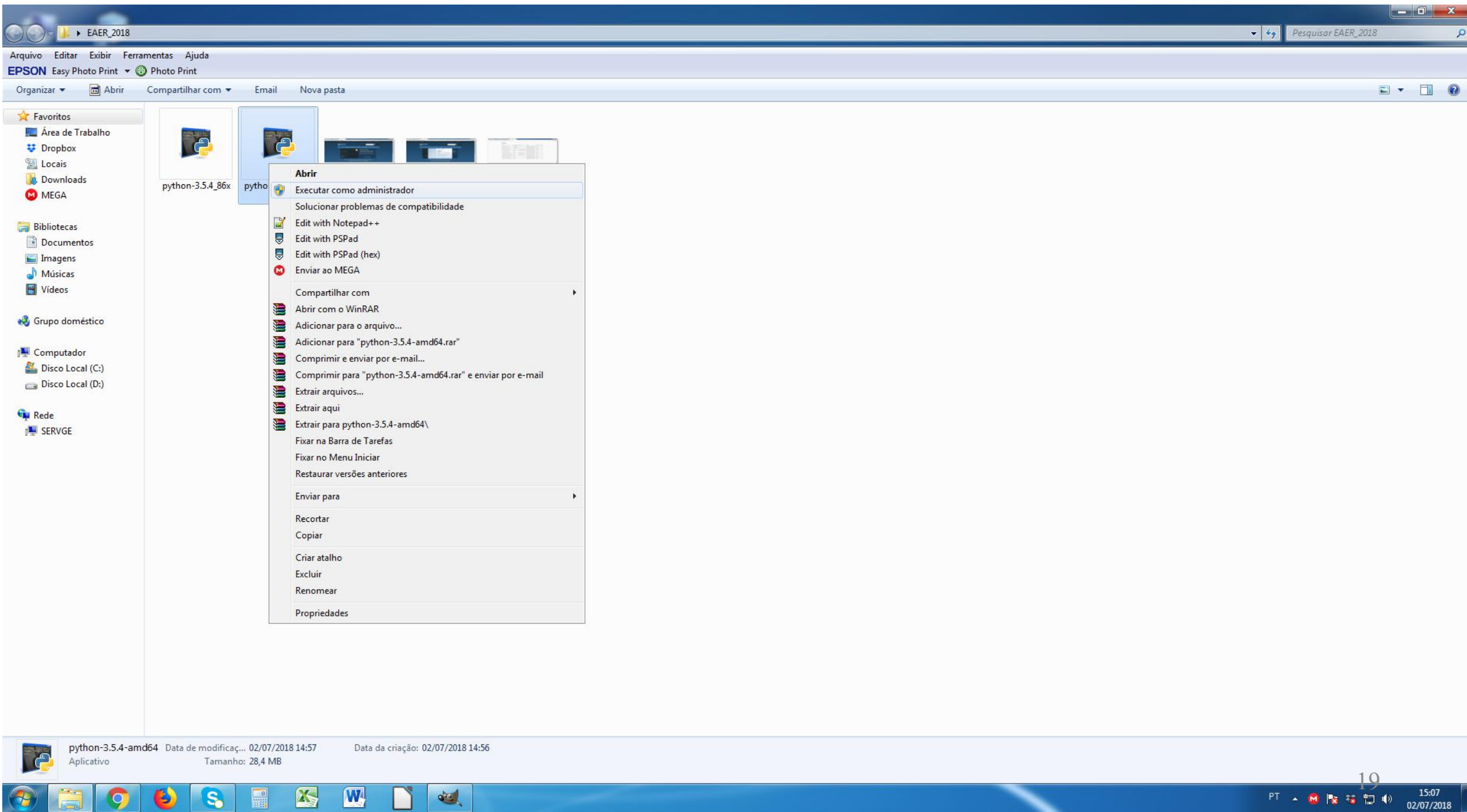
- ActiveState;
- Advanced Simulation Technology Inc. (ASTi);
- Array BioPharma, Inc.;
- Beslist.nl;
- BizRate.com (<http://BizRate.com>);
- Canonical;
- CCP Games.;
- cPacket Networks;
- Edgestream Partners, L.P.;
- Enthought, Inc.;
- Globo;
- Google;
- Hood Media GmbH;
- KNMP ;
- Lincoln Loop;
- Lucasfilm;
- Microsoft;
- Online Degree Reviews;
- OpenEye Scientific Software;
- O'Reilly Media, Inc.; Red Hat;
- SEO Moves;
- Uniblue Systems Ltd.;
- Wargaming.com (<http://Wargaming.com>);
- ZeOmega, LLC.

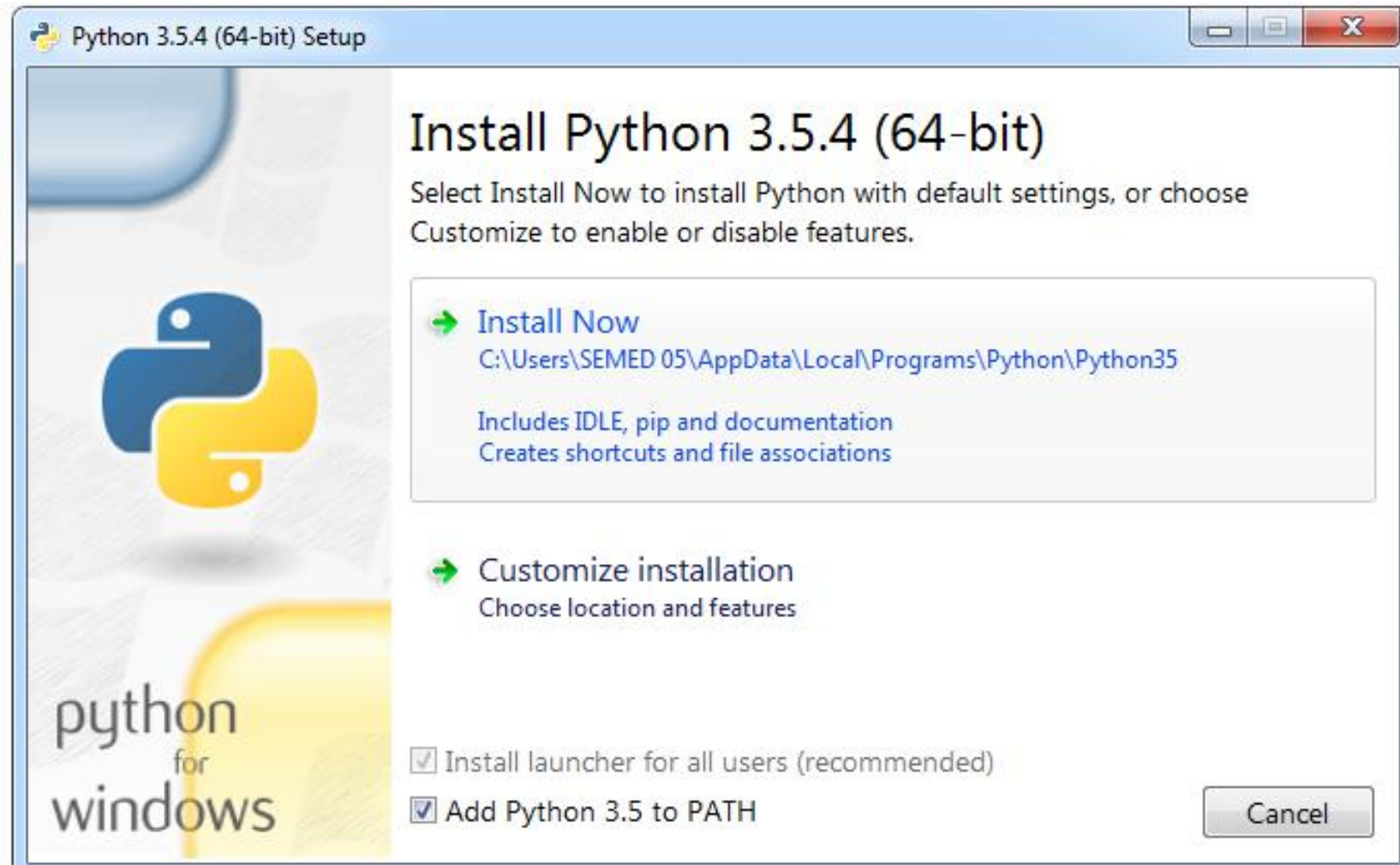


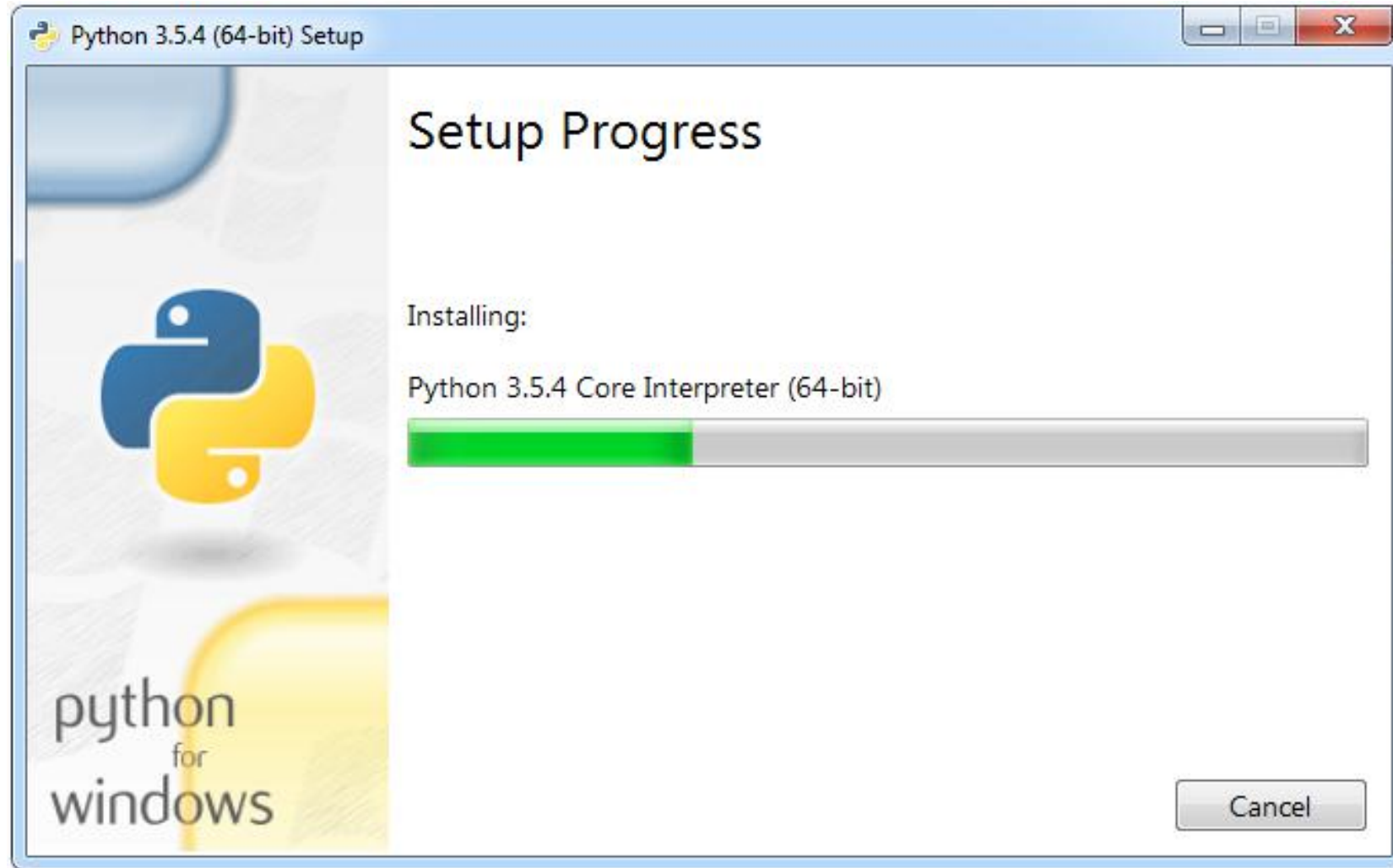
Instalação da Linguagem em Ambiente Windows

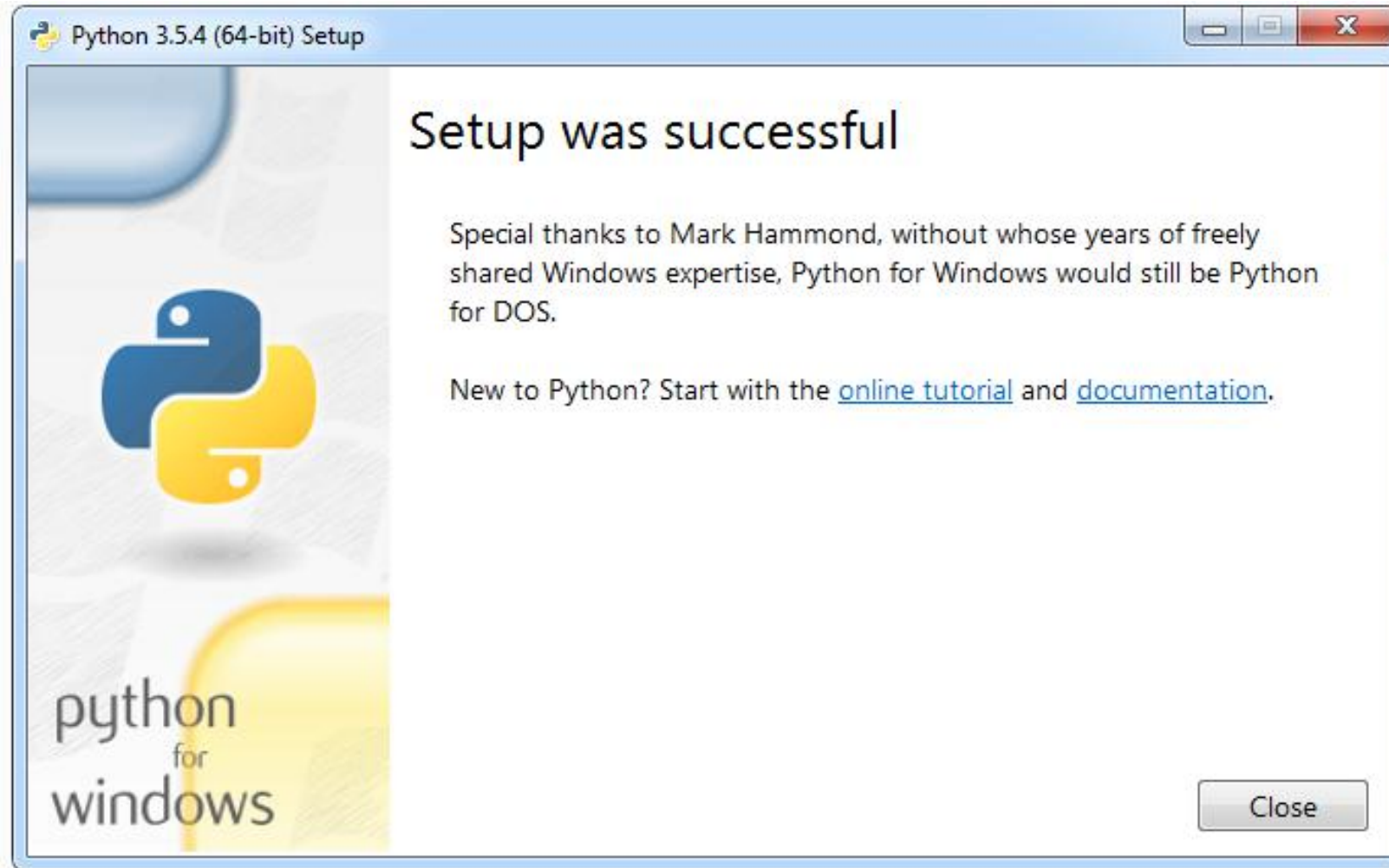
Instalação da Linguagem - Win

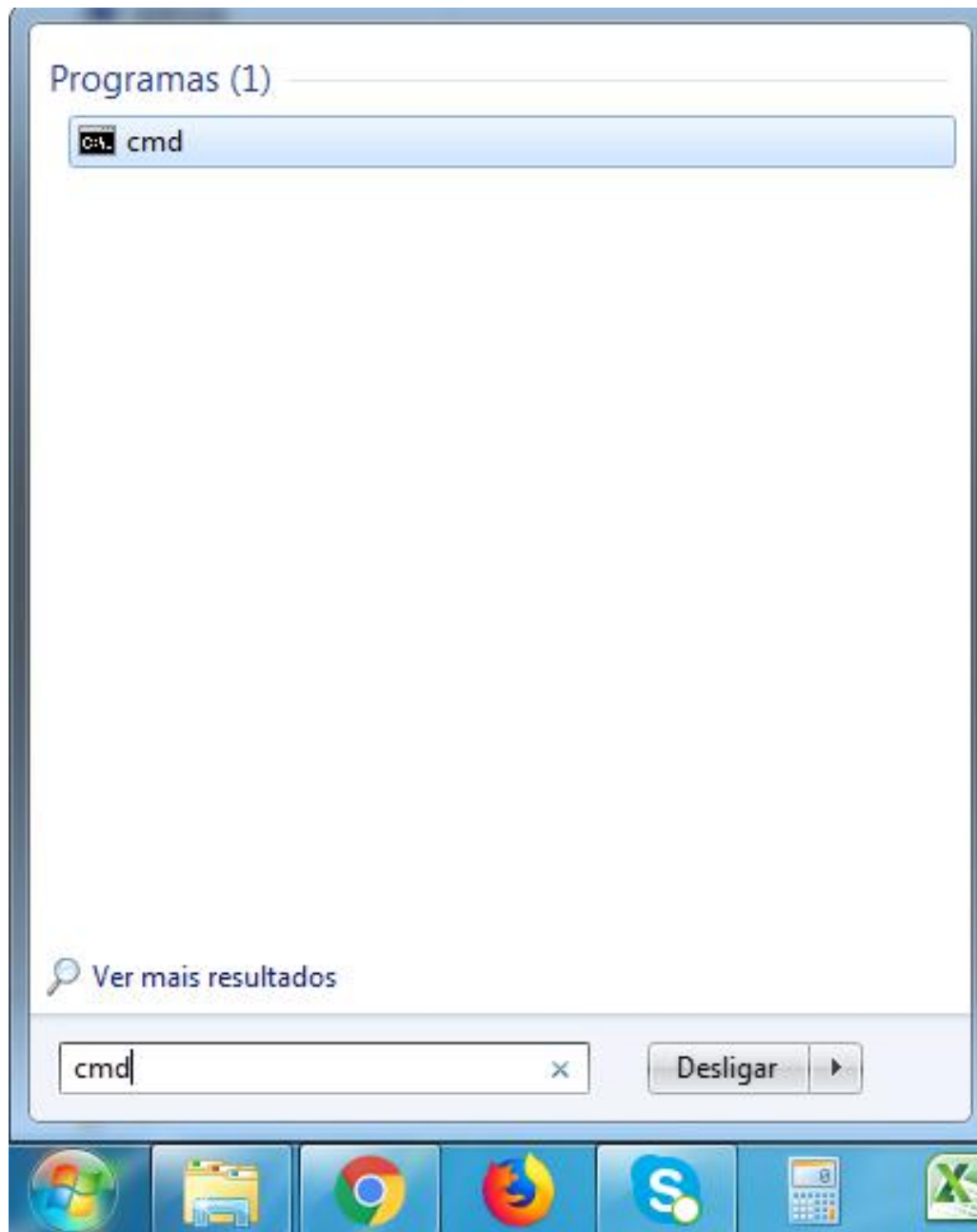
- Python 3.5;
- Verificar o "tipo" do sistema, se 32 bits ou 64 bits;
- Executar o .EXE (32 bits ou 64) como administrador (caso apareça alguma solicitação de permissão, clique em sim ou *yes*);
- Feito isso, preferir pela instalação padrão, não esquecendo de marcar as opções, e clicar em "*Install Now*".

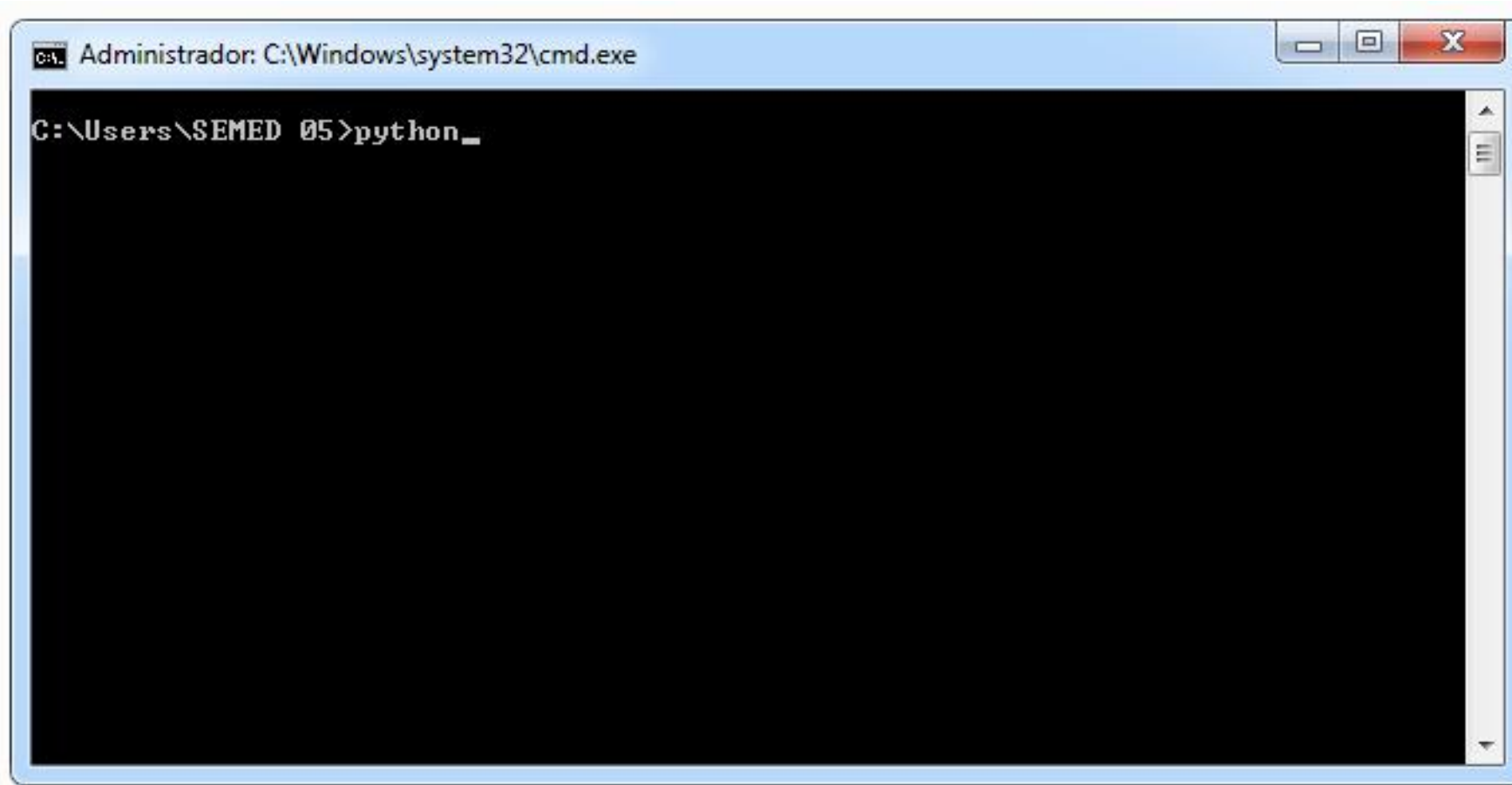




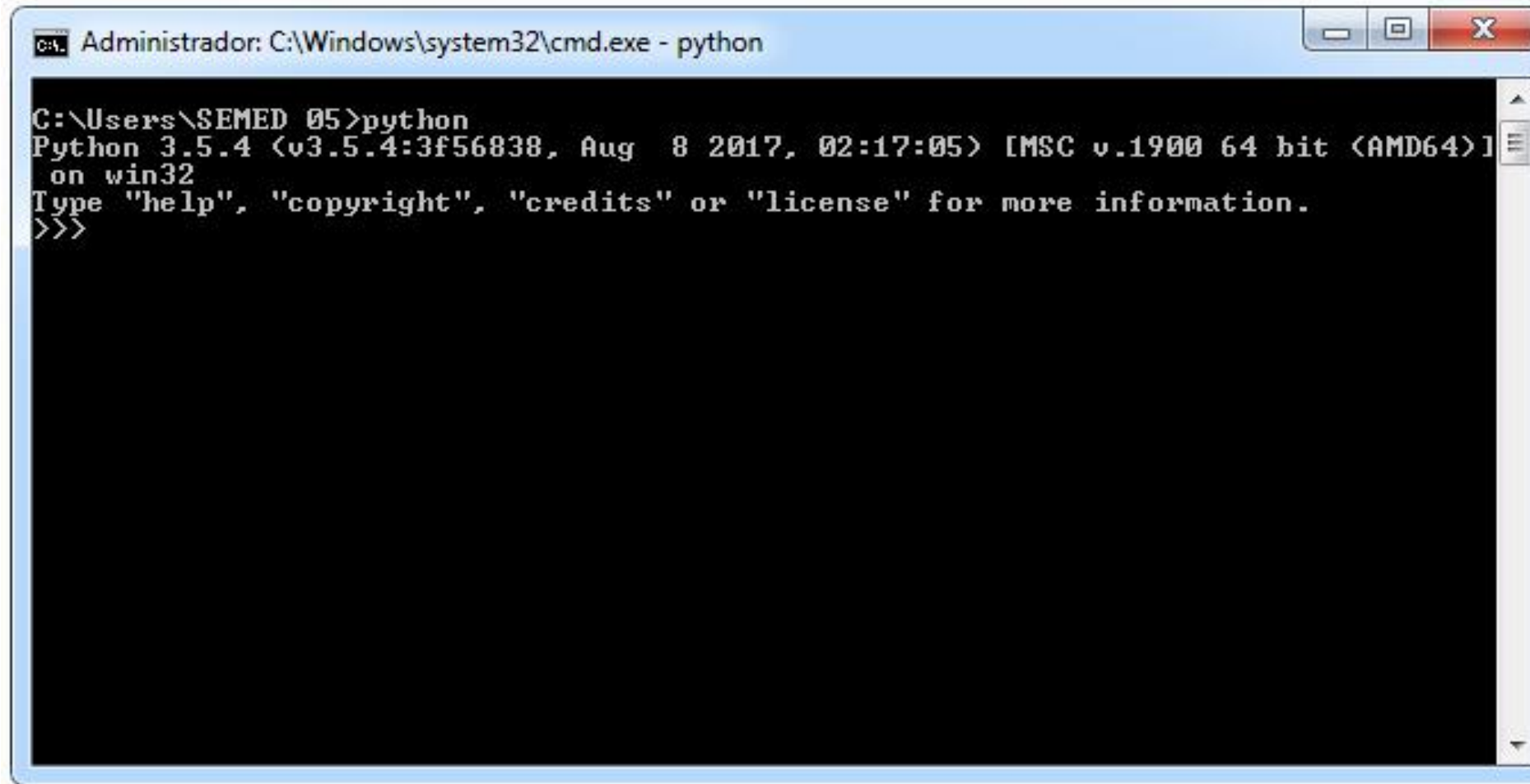






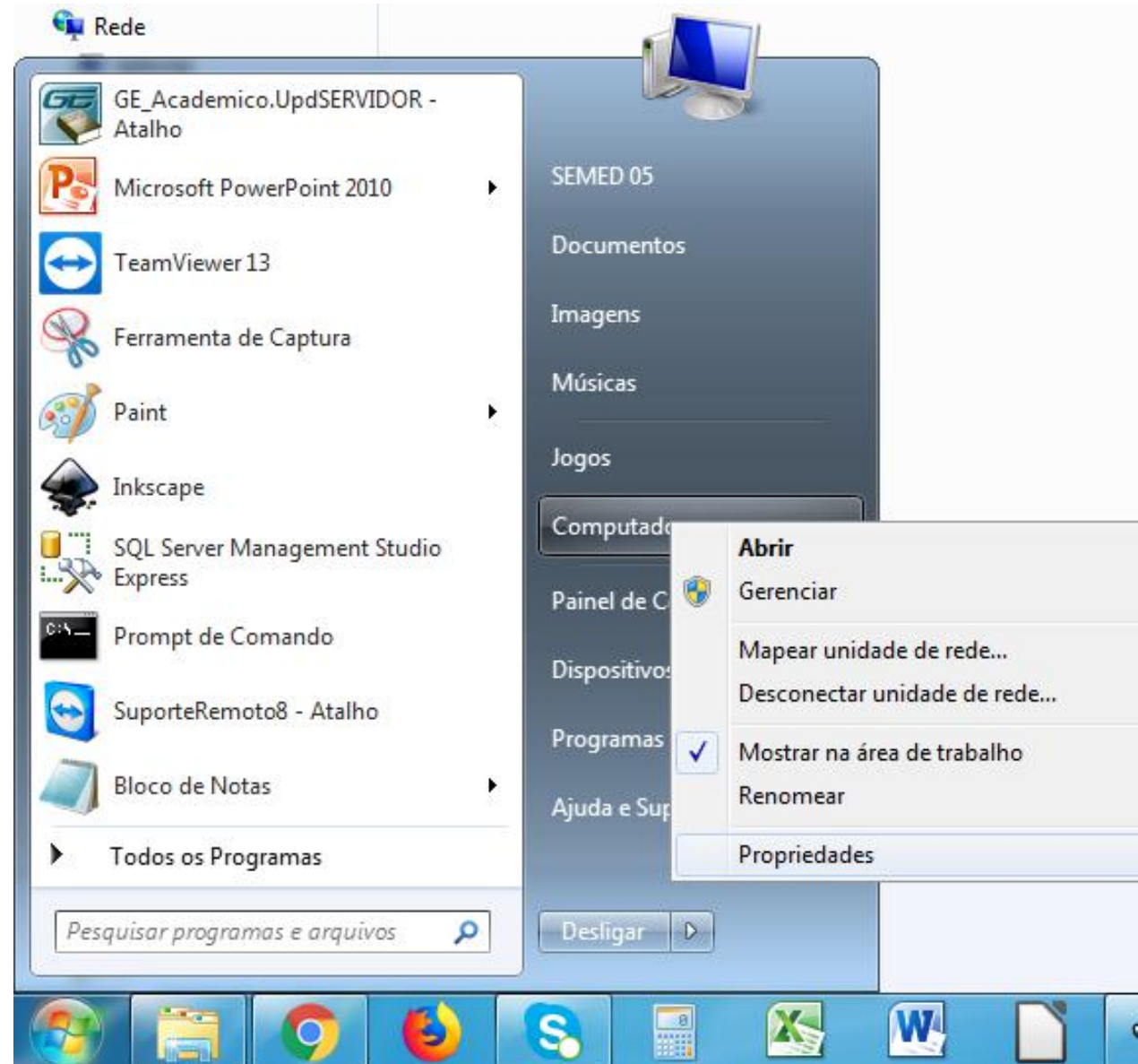


```
Administrador: C:\Windows\system32\cmd.exe  
C:\Users\SEMED 05>python_
```

A screenshot of a Windows command prompt window. The title bar reads "Administrador: C:\Windows\system32\cmd.exe - python". The command prompt shows the user typing "python" at the "C:\Users\SEMED 05" prompt. The output displays the Python version and build information: "Python 3.5.4 (v3.5.4:3f56838, Aug 8 2017, 02:17:05) [MSC v.1900 64 bit (AMD64)] on win32". It then prompts the user with "Type 'help', 'copyright', 'credits' or 'license' for more information." and shows the interactive prompt ">>>".

```
C:\Users\SEMED 05>python
Python 3.5.4 (v3.5.4:3f56838, Aug 8 2017, 02:17:05) [MSC v.1900 64 bit (AMD64)]
on win32
Type "help", "copyright", "credits" or "license" for more information.
>>>
```



Exibir informações básicas sobre o computador

Windows Edition

Windows 7 Ultimate

Copyright © 2009 Microsoft Corporation. Todos os direitos reservados.

Service Pack 1



Sistema

Classificação:  Índice de Experiência do Windows

Processador: Intel(R) Pentium(R) CPU N3530 @ 2.16GHz 2.17 GHz

Memória instalada (RAM): 4,00 GB (utilizável: 3,88 GB)

Tipo de sistema: Sistema Operacional de 64 Bits

Caneta e Toque: Nenhuma Entrada à Caneta ou por Toque está disponível para este vídeo

Nome do computador, domínio e configurações de grupo de trabalho

Nome do computador: ServGe

Nome completo do computador: ServGe

Descrição do computador: ServGe

Grupo de trabalho: SEMED1



Ativação do Windows

Windows ativado

ID do Produto (Product ID): 00426-OEM-8992662-00006



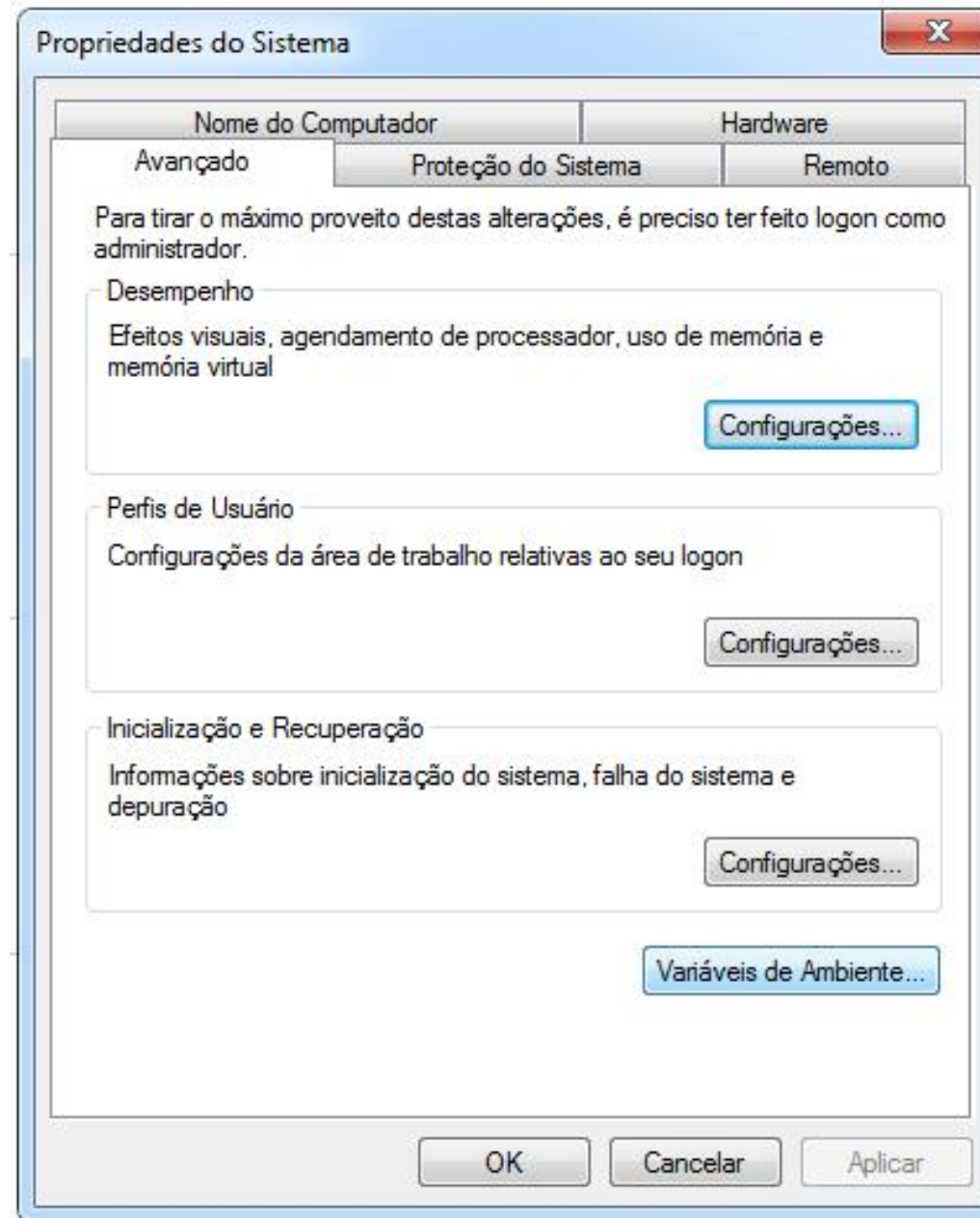
Obter mais informações online...

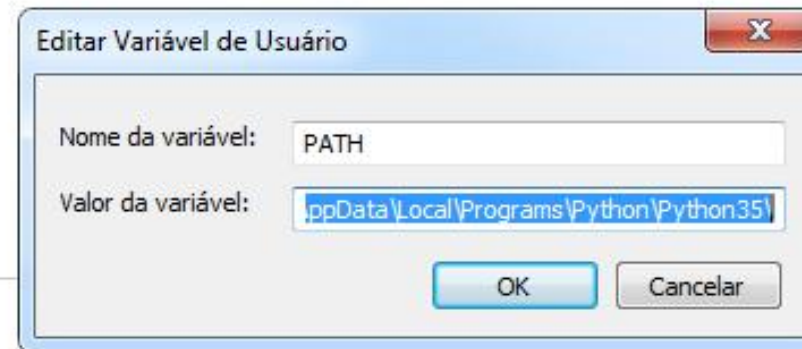
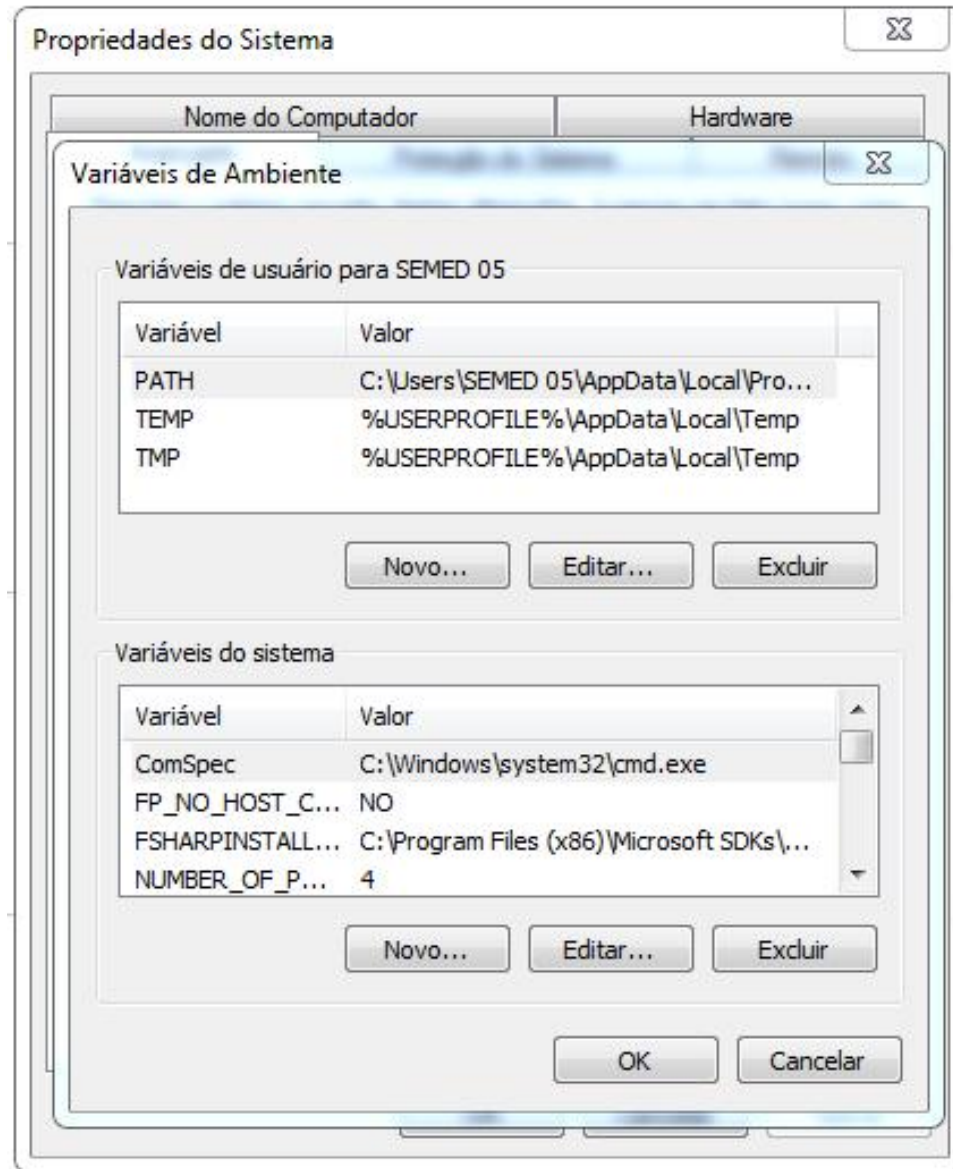
Consulte também

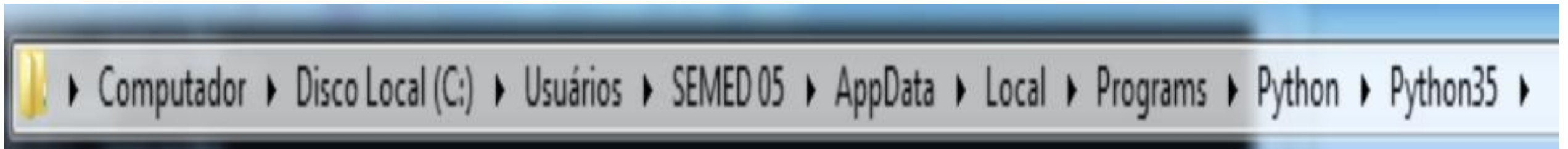
Central de Ações

Windows Update

Informações e Ferramentas de Desempenho







Instalação da Linguagem em Ambiente GNU/Linux

Instalação da Linguagem - GNU

- Na geral, as distros GNU/Linux já vem com Python instalado;
- Verificar a versão da linguagem (Python 3);
 - `python3 --version`
- Caso a versão 3.5 não esteja instalada
 - `sudo apt-get install python3.5 idle-python3.5`

Operações Básicas

Operações Básicas

OPERAÇÃO	NOTAÇÃO EM PYTHON
SOMA	+
SUBTRAÇÃO	-
MULTIPLICAÇÃO	*
DIVISÃO	/
EXPONENCIAÇÃO	**
RESTO DA DIVISÃO INTEIRA	%

Operações Básicas

- 1) Converta as seguintes expressões matemáticas para que possam ser calculadas usando o interpretador Python (MENEZES, 2016).

$$10 + 20 \times 30$$

```
|__ 10 + 20 * 30
```

$$(9^4 + 2) \times 6 - 1$$

```
|__ ((9**4)+2)*(6-1)
```

$$4^2 \div 30$$

```
|__ (4**2)/30
```

Variáveis e Entradas de Dados

Variáveis e Entradas de Dados

Nome	Válido	Comentários
a1	sim	Embora contenha um número, o nome a1 inicia com letra.
velocidade	sim	Nome formado por letras.
velocidade90	sim	Nome formado por letras e números, mas iniciado por letra.
salário_médio	sim	O símbolo sublinha () é permitido e facilita a leitura de nomes grandes.
salário médio	não	Nomes de variáveis não podem conter espaços em branco.
_b	sim	O sublinha () é aceito em nomes de variáveis, mesmo no início.
1a	não	Nomes da variáveis não podem começar com número.

Fonte: MENEZES, 2016.

Variáveis e Entradas de Dados

- Para entrada de dados usamos a função *input*. Esta função pode estar atribuída (=) a uma variável.

- Exemplo:

```
>>> nome = input('Seu nome: ')
```

```
>>> nome
```

```
['__ 'Jackson Osvaldo']
```

Variáveis e Entradas de Dados

- Você pode usar a função *print* para exibir uma mensagem no terminal, bem como para mostrar o valor de uma dada variável ou o valor de uma operação matemática.
- Exemplo:
 - >>> print ('Olá, mundo!')
 - >>> print (nome)
 - >>> print (2**3+2)

Tipos de Dados em Python

Tipos de Dados em Python

- **str** (string);
- **True** ou **False** (Lógicos);
- **int** (inteiro);
- **float** (ponto flutuante frações);

Tipos de Dados em Python - **str**

- O tipo **str** (String);

|__ Textos ou dados 'brutos' provenientes de uma determinada entrada, seja via terminal ou não.

Exemplo: O programador pede para você digitar o seu nome ou um número. Os dados que você inseriu são, inicialmente, 'textos'.

Tipos de Dados em Python - **str**

```
>>> help(str)
```

__ Pedindo Ajuda

```
>>> dir(str)
```

__ Mostrando as "ações"

Python 3.5.2 (default, Nov 23 2017, 16:37:01)

[GCC 5.4.0 20160609] on linux

Type "copyright", "credits" or "license()" for more information.

>>> help(str)

Help on class str in module builtins:

class str(object)

str(object='') -> str

str(bytes or buffer[, encoding[, errors]]) -> str

Create a new string object from the given object. If encoding or errors is specified, then the object must expose a data buffer that will be decoded using the given encoding and error handler. Otherwise, returns the result of object. str () (if defined) or repr(object).

encoding defaults to sys.getdefaultencoding().

errors defaults to 'strict'.

Methods defined here:

add (self, value, /)
Return self+value.

contains (self, key, /)
Return key in self.

eq (self, value, /)
Return self==value.

format (...)

```
>>> dir(str)
[' add ', ' class ', ' contains ', ' delattr ', ' dir ', ' doc ', ' eq ', ' format ', ' ge ', ' getatt
ribute ', ' getitem ', ' getnewargs ', ' gt ', ' hash ', ' init ', ' iter ', ' le ', ' len ', ' lt ',
' mod ', ' mul ', ' ne ', ' new ', ' reduce ', ' reduce ex ', ' repr ', ' rmod ', ' rmul ', ' setattr
', ' sizeof ', ' str ', ' subclasshook ', 'capitalize', 'casefold', 'center', 'count', 'encode', 'endswith', 'expa
ndtabs', 'find', 'format', 'format map', 'index', 'isalnum', 'isalpha', 'isdecimal', 'isdigit', 'isidentifier', 'islower'
, 'isnumeric', 'isprintable', 'isspace', 'istitle', 'isupper', 'join', 'ljust', 'lower', 'lstrip', 'maketrans', 'partitio
n', 'replace', 'rfind', 'rindex', 'rjust', 'rpartition', 'rsplit', 'rstrip', 'split', 'splitlines', 'startswith', 'strip'
, 'swapcase', 'title', 'translate', 'upper', 'zfill']
>>>
```

Ln: 394 Col: 0

Tipos de Dados em Python - **str**

```
>>> nome = 'JAckson OsvAldo'
```

```
>>> nome.replace('A','a')
```

```
|__ 'Jackson Osvaldo'
```

```
>>> nome
```

```
|__ 'JAckson OsvAldo'
```

Tipos de Dados em Python - **bool**

- O tipo lógico (**True** ou **False**);
|__ São tipos de dados booleanos.

Exemplo:

```
>>> resultado = True
```

```
>>> aprovado = False
```

Tipos de Dados em Python - **bool**

- Operadores Relacionais

|__ Para realizarmos operações lógicas, utilizamos os operadores relacionais

Operador	Operação	Símbolo matemático
==	igualdade	=
>	maior que	>
<	menor que	<
!=	diferente	≠
>=	maior ou igual	≥
<=	menor ou igual	≤

Fonte: MENEZES, 2016.

Tipos de Dados em Python - **bool**

- Operadores Lógicos

|__ Para agrupar operações com lógica booleana, utilizaremos operadores lógicos.

Operador Python	Operação
not	não (negação)
and	e (conjunção)
or	ou (disjunção)

Fonte: MENEZES, 2016.

Tipos de Dados em Python - **bool**

```
>>> número = 201
```

```
>>> número == 201
```

```
>>> número != 201
```

```
>>> número = False
```

Tipos de Dados em Python - **int**

- O tipo inteiro (**int**);
|__ São dados do tipo inteiro, ou seja não possuem fração.

Exemplo:

```
>>> num1 = 2018  
>>> num2 = 1997  
>>> print ('Você tem', num1-num2, 'anos.')
```

Tipos de Dados em Python - **float**

- O tipo flutuante (**float**);
|__ São dados que contém alguma fração, ou seja, ponto flutuante.

Exemplo:

```
>>> num1 = 1000.798
```

```
>>> num2 = 190.82
```

```
>>> print ('Você gastou R$',num2,'e tem R$', num1-num2, 'restantes.')
```

Exercitando

Exercitando

2) Escreva um programa que calcule a temperatura em graus F. Lembre-se, o usuário forneceu a temperatura em graus C.

$$F = ((9 \times C) \div 5) + 32$$

Exercitando

3) Um aluno deseja saber se foi aprovado na disciplina de Programação I. Para isso, ele precisa tirar a média aritmética de suas duas notas, sendo que o resultado tem de ser maior ou igual a 6. Do contrário, o aluno está reprovado.

$$\text{MediaAritimética} = (\text{nota1} + \text{nota2})/2$$

Exercitando

4) Escreva um programa que pergunte a quantidade de km percorridos por um carro alugado pelo usuário, assim como a quantidade de dias pelos quais o carro foi alugado. Calcule o preço a pagar, sabendo que o carro custa R\$ 60 por dia e R\$ 0,15 por km rodado.

Exercitando

5) Escreva um programa que calcule o tempo de uma viagem de carro. Pergunte a distância a percorrer e a velocidade média esperada para a viagem.

$$V_m = S/t$$

Até Amanhã, pessoal. :)