

Python

Universidade Federal Rural do Rio de Janeiro
PET-Sistemas de Informação

Lucas Nunes Paim
lucaspaim@pet-si.ufrrj.br

PET-SI

Aula 1

1. Introdução

O que é Python?

Por que usar Python?

2. Instalação

3. Variáveis

4. Primeiros Trabalhos

Python

Python é uma linguagem de programação relativamente **simples** que foi criada por **Guido van Rossum** em 1991, ela é de **alto nível**, **interpretada** e de **alta produtividade**

- **Simples**

- Elegante - *Menos linhas de código comparando como Java, C, C++*
- Documentação Gratuita e de fácil acesso

- **Alto nível**

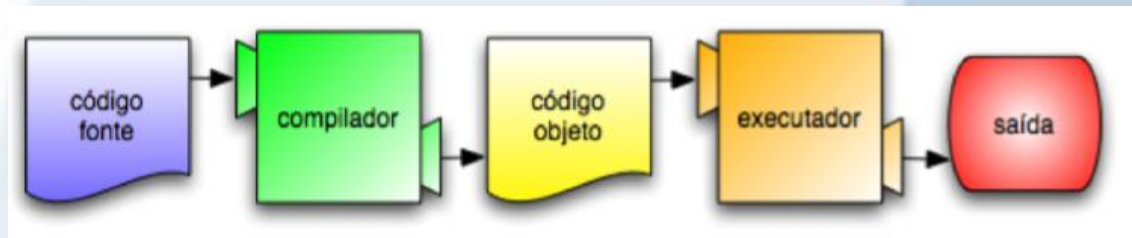
- Abstração elevada
- Longe do código de máquina
- Próximo à linguagem humana – *É como escrever uma carta*

Python

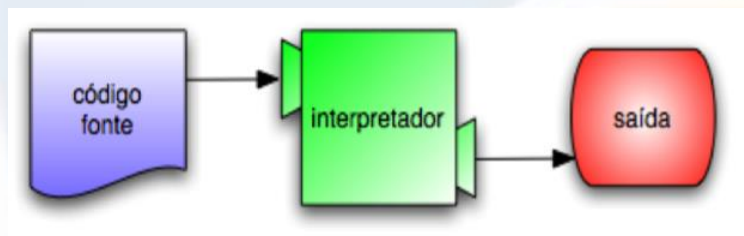
Python é uma linguagem de programação relativamente **simples** que foi criada por **Guido van Rossum** em 1991, ela é de **alto nível**, **interpretada** e de **alta produtividade**

- **Interpretada**

O código fonte é executado por um programa de computador, evita “codifica-compila-roda”



Compilada

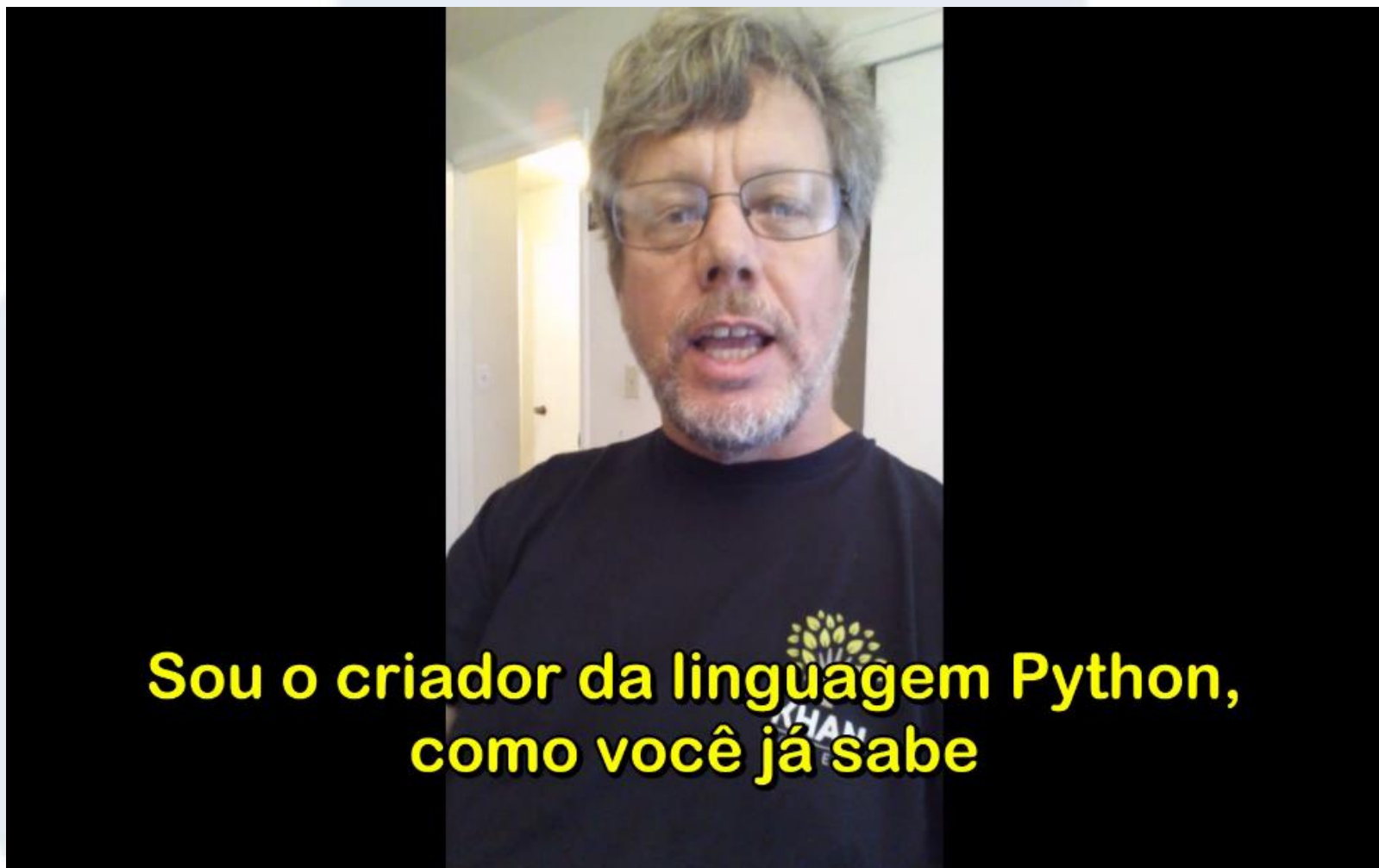


Interpretada

- **Alta Produtividade**

- Imperativa
- Orientada a objetos
- Funcional

Por que Guido criou o Python?

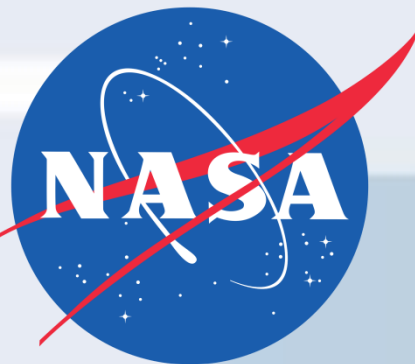


Agradecimentos especiais a [Ana Paula Teixeira](#) – Ciência da Computação UFRRJ/IM

PET-SI

Quem usa Python?

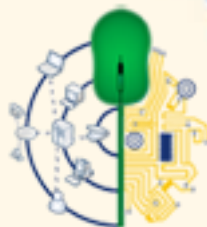
Google



globo.com



Massachusetts
Institute of
Technology



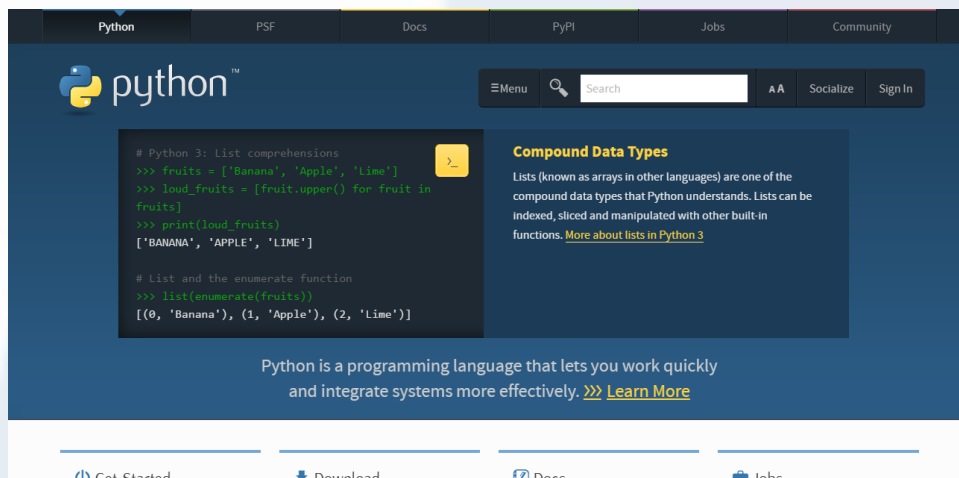
PET-SI

Stanford
University

PET-SI

Instalando o Python

Entrar no site do desenvolvedor do Python: www.python.org



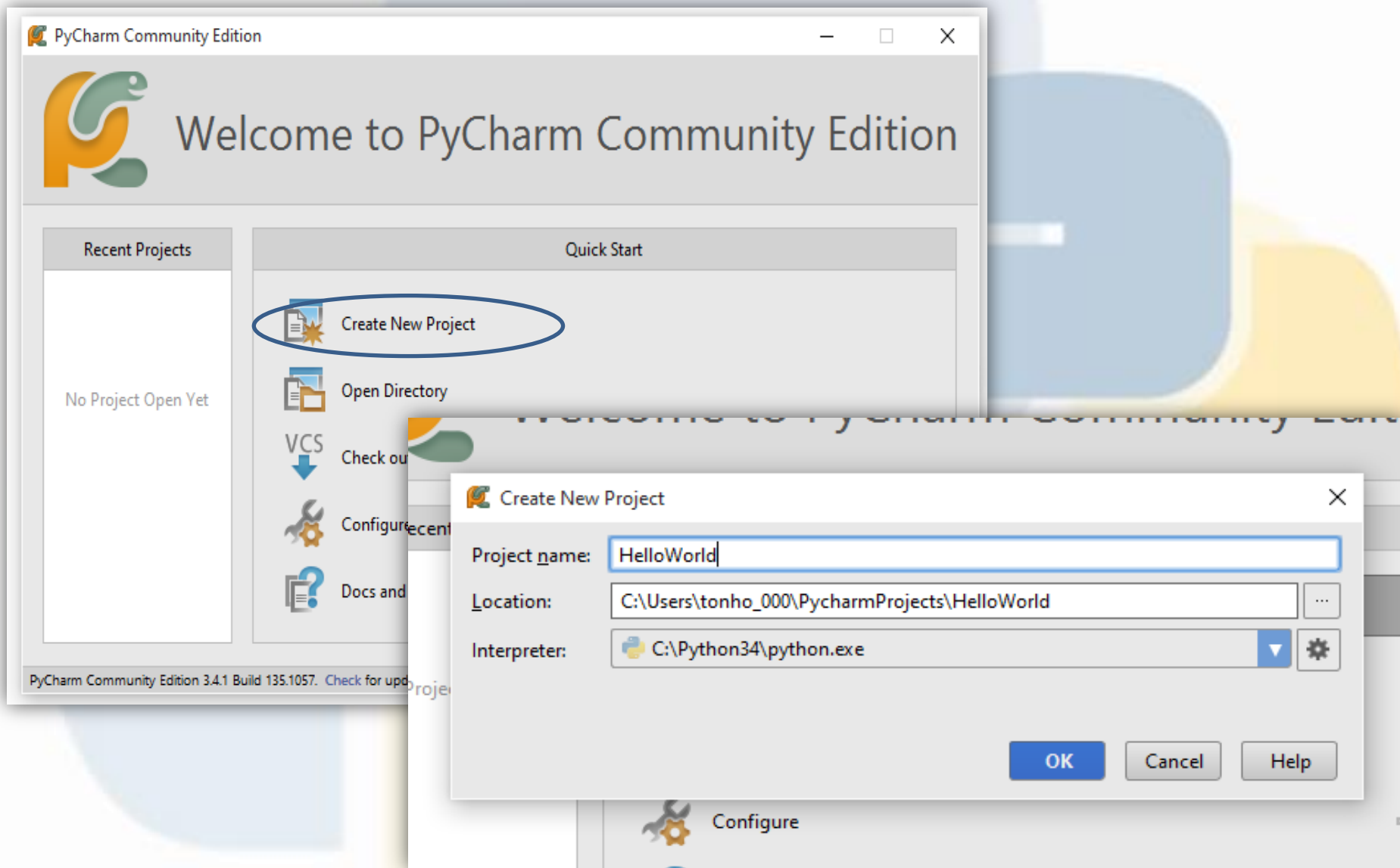
No rodapé do site clicar em **Download**

Escolher a versão do Python desejada



PET-SI

Executando o PyCharm e criando um novo projeto



Tela “**Did You Know?**”: Pode ser fechada

- Não há **declaração de tipos de variáveis**
- Não há **Begin** e **End**, **{ }** ou **;**
- Comentários são feitos com **#**
- Comentários de mais de uma linha **““““**
- Identação é **OBRIGATÓRIA PRA CARAMBA**
- Organização é fundamental
- Nessa versão do **PyCharm** precisamos usar **()** para imprimir
- É case-sensitive

Variáveis String

São variáveis do tipo texto, o texto fica entre aspas “ ”

CÓDIGO

```
a = "Olá mundo"
b = "Hello World"
c = "Olá PET-SI"
d = "Olá UFRRJ"
e = "Curso"
f = "Python"
```

```
print(a)
print(b)
print(c)
print(d)
print(e+" de "+f)
```

SAÍDA

```
Olá mundo
Hello World
Olá PET-SI
Olá UFRRJ
Curso de Python
```

CÓDIGO

```
a = "PET-SI"
b = " está oferecendo um
minicurso de Python"
c = " para alunos da"
d = " UFRRJ"
```

```
print("O "+a+b+c+d)
```

SAÍDA

```
O PET-SI está oferecendo um
minicurso de Python para
alunos da UFRRJ
```

Variáveis String – Leitura

CÓDIGO

```
nome = input('Entre com o seu nome: ')\nprint(nome)
```

```
Entre com o seu nome: fulano de tal\nfulano de tal
```

CÓDIGO

```
nome = input('Digite seu nome: ')\ncurso = input('Digite seu curso: ')\n\nprint('Olá '+nome+', ficamos felizes em conhecer você, gostamos\nmuito do curso de '+curso)
```

```
Digite seu nome: Lucas\nDigite seu curso: Sistemas de Informação\nOlá Lucas, ficamos felizes em conhecer você, gostamos muito do curso de Sistemas de Informação
```

CÓDIGO

```
a = "abcdefghijkl"  
print(a[9])  
print(a[0])  
  
print(a[3])  
print(a[3:])
```

SAÍDA

```
j  
a  
d  
defghijkl
```

CÓDIGO

```
nome = input("Digite seu primeiro nome: ")  
print("A primeira letra do seu nome é: "+nome[0])
```

```
Digite seu primeiro nome: Lucas  
A primeira letra do seu nome é: L
```

Exercício 1:

Faça um programa que mostre o tradicional “Hello World!” na tela

Exercício 2:

Faça um programa que peça um número e então mostre a mensagem: *O número informado foi [número]*.

Exercício 3: (Sem estruturas de repetição)

Faça um programa que peça 5 itens e suas respectivas quantidades e mostre na tela a lista de itens com a quantidade,

Item 1 – Quantidade: V

Item 2 – Quantidade: W

Item 3 – Quantidade: Y

Item 4 – Quantidade: X

Item 5 – Quantidade: Z

Exercício 1

```
print("Hello World!")
```

Exercício 2

```
numero = input("Digite um número: ")  
print("O número informado foi "+numero)
```

Exercício 3

```
item1 = input("Digite o nome do primeiro item: ")  
...  
item5 = input("Digite o nome do quinto item: ")  
qt1 = input("Digite o nome do primeiro item: ")  
...  
qt5 = input("Digite o nome do quinto item: ")  
print(item1+" - Quantidade: "+qt1)  
...  
print(item5+" - Quantidade: "+qt5)
```

Variáveis Numéricas

O Python possui alguns tipos numéricos pré-definidos:

- **Inteiros** (*int*)
- Ponto flutuante (*float*)
- Booleanos (*bool*)
- Complexos (*complex*)

Elas suportam as operações matemáticas básicas

```
>>> a, b = 1, 2.5      # atribui 1 a "a" e 2.5 a "b"
>>>                    # um inteiro e um ponto flutuante
>>> c = True           # booleano
>>> z = 3 + 4j          # complexo
>>>
>>> a + b              # resultado em ponto flutuante
3.5
>>> int(a + b)         # resultado inteiro
3
>>> b * z              # resultado complexo
(7.5+10j)
>>> type(z)            # mostra a tipagem da variável
<type 'complex'>
```

Operadores matemáticos

SOMA (+)	SUBTRAÇÃO (-)	MULTIPLICAÇÃO (*)	DIVISÃO (/)
>>> 8+1 9 >>> 8+1.5 9.5 >>> 8+0 8 >>> 8+ -1 7	>>> 5 – 2 3 >>> 2 – 10 -8 >>> 15 – 1.5 13.5	>>> 25 * 2 50 >>> 10 * 1.5 15.0 >>> 250 * 12 3000	>>> 10/12 0.8333333333333334 >>> 3.5/8.1 0.4320987654320988 >>> 5/2 2.5 >>> 10/2 5.0 >>> 20/0 ZeroDivisionError

Criar uma lista de compra com as seguintes regras:

- É necessário um total de 5 frutas;
- A **primeira** fruta deve **custar 1,00**;
- A **segunda** fruta deve **custar o dobro do valor da primeira**;
- A **terceira** fruta deve **custar metade do valor da primeira**;
- A **quarta** fruta deve **custar 3 vezes o valor da terceira fruta**;
- A **quinta** fruta deve **custar metade do valor da quarta**;
- Cada fruta deve possuir uma variável;
- Usar a menor quantidade possível de variáveis;
- Todas as frutas e seus valores devem ser impressos no seguinte formato:

“A fruta _____ custa _____”

```
a = "Maçã"
b = "Pera"
c = "Uva"
d = "Limão"
e = "Laranja"
preco = 1
print("A fruta", a,"custa", preco)
print("A fruta", b,"custa", preco*2)
preco = preco/2
print("A fruta", c,"custa", preco)
preco = preco*3
print("A fruta", d,"custa", preco)
print("A fruta", e,"custa", preco/2)
```

(Dicionários)

- Dicionários são coleções de elementos onde é possível utilizar um índice de qualquer tipo **imutável**.
- Os dicionários implementam mapeamentos que são coleções de associações entre pares de valores
O primeiro elemento é a **chave**
O segundo elemento é o **conteúdo/valor**

```
DICIONARIO = {"ALAN":'001',"AMARILDO":'002',"ANA":'003',"ARISTIDES":'004'}
```

- As chaves dos dicionários são armazenadas por tabelas de espalhamento (Hash Tables)
- Diferente de listas, não existe uma ordem específica de armazenamento no dicionário

Criação do Dicionário

```
dic = {"Nome": 'Larissa', "Sobrenome": 'Maria'}
```

```
dic = {"Alan": '001', "Amarildo": '002', "Ana": '003', "Aristides": '004'}
```

Operações com Dicionário

`print(dic["Nome"])` - Imprime o conteúdo da chave Nome

`print(dic["Sobrenome"])` – Imprime o conteúdo da chave Sobrenome

`print(dic.keys())` – Imprime apenas as chaves

`print(dic.values())` – Imprime apenas os conteúdos

`print(dic.items())` – Imprime as chaves e conteúdos

Inserindo um novo item no dicionário

```
dic["Idade"] = '18'
```

Alterando o valor das chaves

```
dic["Nome"] = 'Rose'
```

Função GET: retorna o valor da chave e NONE caso não exista

```
print(dic.get('Larissa'))  
print(dic.get('Rose'))
```

Função DEL: Apaga determinado item do dicionário

```
del dic["Nome"]
```

Função CLEAR: Apaga todo o dicionário

```
dic.clear()
```

Função COPY: Copia o conteúdo de um dicionário para outro

```
dic2 = dic.copy()
```

CÓDIGO

```
listatel = {"ana":210012,"bianca":210045,"camila":210019}  
  
print(listatel["ana"])  
print(listatel["bianca"])  
print(listatel["camila"])  
  
print(listatel.keys())  
print(listatel.values())
```

SAÍDA

```
210012  
210045  
210019  
dict_keys(['camila', 'bianca', 'ana'])  
dict_values([210019, 210045, 210012])
```

Exercício:

Faça um dicionário que contenha os dados de uma pessoa, são os seguintes dados: (Preencha os dados iniciais como preferir)

- Nome
- Ultimo Nome
- Idade
- Curso
- Endereço

- Imprima o dicionário completo
- Imprima cada um dos 5 itens separadamente
- Exclua a chave **CURSO** do dicionário
- Altere o **ULTIMO NOME** para Lopes
- Imprima novamente o dicionário completo
- Imprima apenas o endereço
- Crie uma cópia do dicionário e altere **Nome** e **Idade**
- Imprima o segundo dicionário completo

```
dic={"nome":"Edna","ultimonome":"Feliz","idade":"20","curso":"direi  
to","endereco":"Rua Lopes Souza Nº 2"}  
print(dic.items()) #Resposta A  
print(dic["nome"]) #Resposta B  
print(dic["ultimonome"]) #Resposta B  
print(dic["idade"]) #Resposta B  
print(dic["curso"]) #Resposta B  
print(dic["endereco"]) #Resposta B  
del dic["curso"] #Resposta C  
dic["ultimonome"] = "Lopes" #Resposta D  
print(dic.items()) #Resposta E  
print(dic["endereco"]) #Resposta F  
dic2 = dic.copy() #Resposta G  
dic2["nome"] = "Olivia" #Resposta G  
dic2["idade"] = "17" #Resposta G  
print(dic2.items()) #Resposta H
```


Material Usado

- Site Oficial do Python: www.python.org
- WorkShop Python – I SASI UFRRJ (Pfª Soraia Pacheco Felício)
- Livro *Use a cabeça - Programando usando a linguagem Python*
- Python.org.br