

# Introdução a linguagem de programação



python



MINISTRANTE: ADILMAR COELHO DANTAS  
MSC: CIÊNCIA DA COMPUTAÇÃO

# Introdução

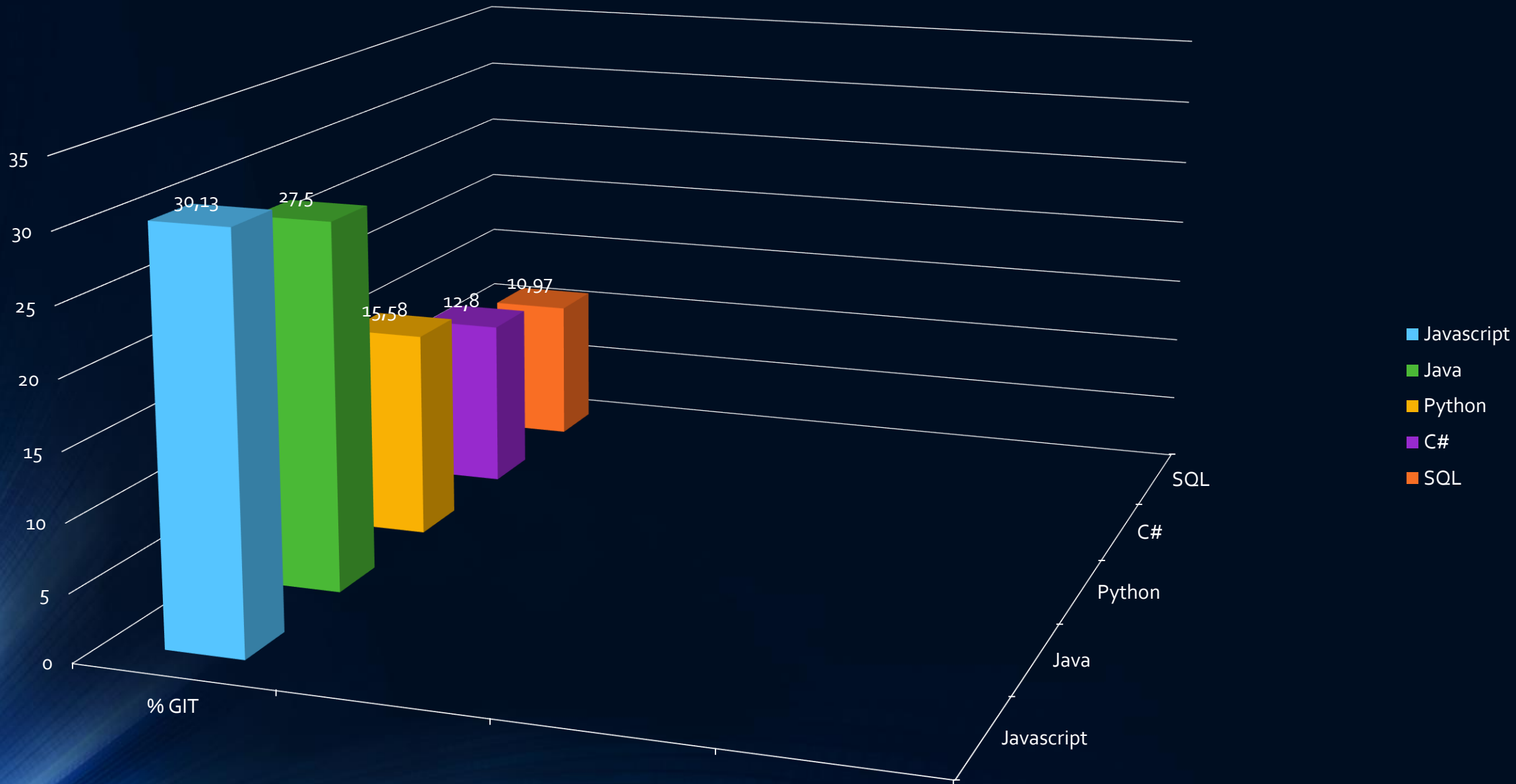
Python é uma linguagem de programação de alto nível interpretada, interativa, orientada a objetos e de alto nível. Foi criado por Guido van Rossum durante 1985-1990.

- Python é interpretado: o Python é processado no tempo de execução pelo intérprete. Você não precisa compilar seu programa antes de executá-lo. Isso é semelhante ao PERL e ao PHP.
- Python é interativo: você pode programar no prompt de Python e interagir com o intérprete diretamente para escrever seus programas.

# Introdução

- Python é orientado a objetos: o Python suporta estilo orientado a objetos ou técnica de programação que encapsula código dentro de objetos.
- Python é um idioma para iniciantes: o Python é uma ótima linguagem para programadores de nível iniciante e oferece suporte ao desenvolvimento de uma ampla gama de aplicativos, desde o processamento de texto simples até navegadores WWW para jogos.

## Linguagens de programação no cenário atual



# Porque aprender Python?

- Fácil de aprender: Python tem poucas palavras-chave, estrutura simples e uma sintaxe claramente definida. Isso permite que o aluno retire o idioma rapidamente.
- Fácil leitura: A exigência da formatação facilita a interpretação no momento da leitura do código fonte.
- Fácil de manter: Por exigir uma boa estrutura facilita na hora de realizar manutenções no fonte.
- Bibliotecas amplas: Possui diversas bibliotecas para as principais plataformas: Windows, MAC e UNIX.



# Porque aprender Python?

- Bancos de dados: o Python fornece interfaces para todos os principais bancos de dados comerciais.
- Programação GUI: o Python suporta aplicações GUI que podem ser criadas e portadas de maneira simples.
- Escalável: Permite trabalhar com aplicações de grande porte.

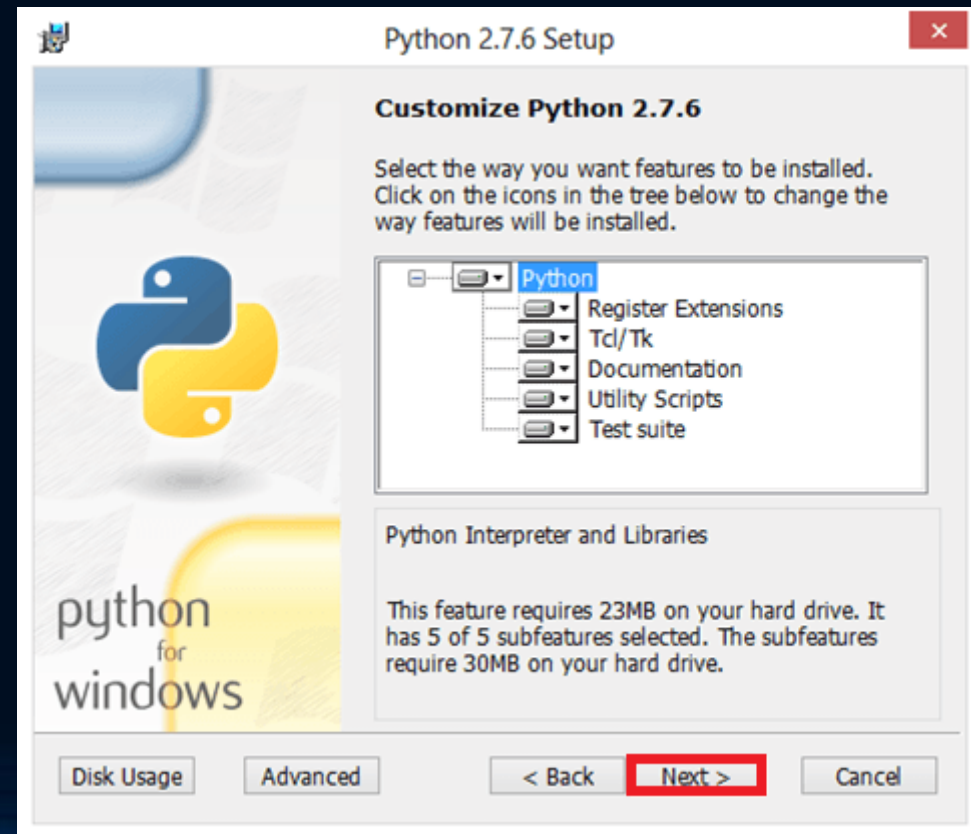
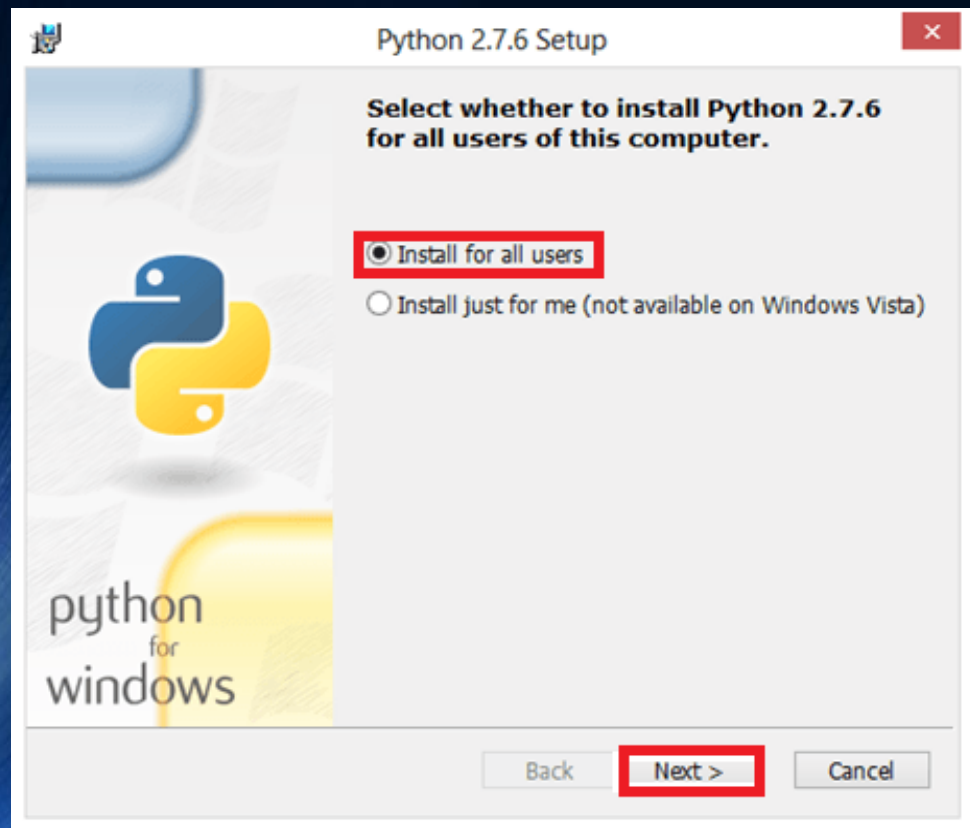


# Instalando o Python

INSTALAÇÃO EM AMBIENTE WINDOWS

# Instalando o Python 2.7

- Acesse: <https://www.python.org/downloads/release/python-2713/>





# Meu primeiro programa em 3 passos!



# Programa 1 – “Hello Word ”

PYTHON – EXERo1.PY

Execute

main.py

```
1  #!/usr/bin/python
2
3  print "Hello, Python!"
```

☒ Result

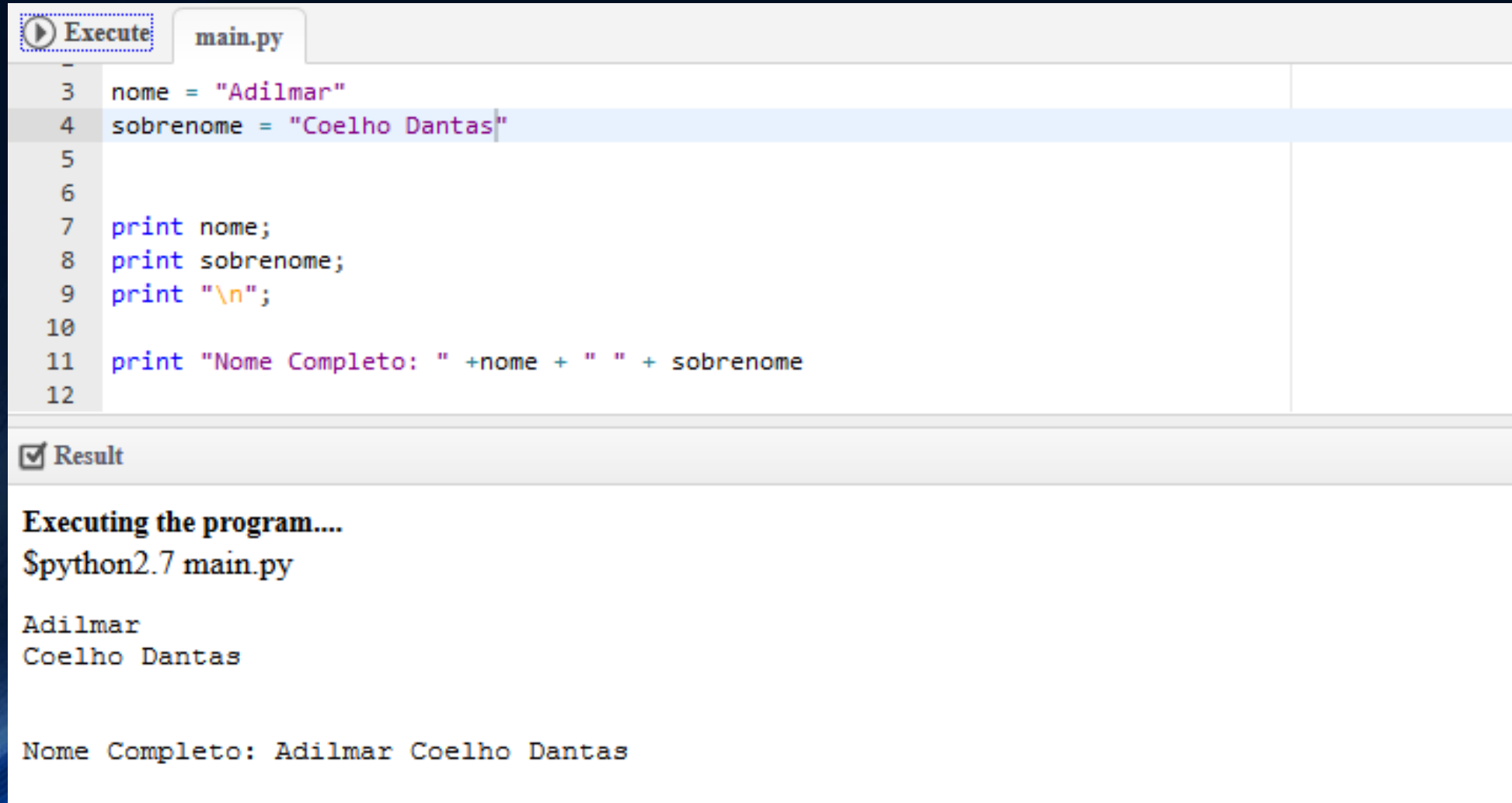
**Executing the program....**

\$python2.7 main.py

Hello, Python!

# Programa 2 – Variáveis

PYTHON – EXER02.PY



The screenshot shows a Python IDE interface. At the top, there is a tab labeled 'main.py' and a button with a play icon and the text 'Execute'. Below this is a code editor with the following Python code:

```
3 nome = "Adilmar"
4 sobrenome = "Coelho Dantas"
5
6
7 print nome;
8 print sobrenome;
9 print "\n";
10
11 print "Nome Completo: " + nome + " " + sobrenome
12
```

Below the code editor is a section titled 'Result' with a checked checkbox. It contains the following text:

**Executing the program....**  
\$python2.7 main.py

Adilmar  
Coelho Dantas

Nome Completo: Adilmar Coelho Dantas

# Programa 3 – Tipos Variáveis

PYTHON – EXER03.PY

	Execute	main.py
3		quantidade = 100 # valor inteiro
4		valor = 1000.0 # valor float
5		nome = "Adilmar" # uma String
6		resultado = True #boolean
7		
8		
9		print nome
10		print quantidade
11		print valor
12		print resultado

☒ Result

**Executing the program....**  
\$python2.7 main.py

Adilmar  
100  
1000.0  
True

# Programa 4 – Operadores aritméticos

PYTHON – EXERo4.PY

```
Execute main.py
2
3 a = 5
4 b = 2
5
6 multiplicacao = a * b
7 soma         = a + b
8 subtracao    = a - b
9 divisao      = a / b
10 restdiv      = a % b
11
12 print "Resultados\n"
13 print "Soma:", soma
14 print "Subtracao:", subtracao
15 print "Dicisao:", divisao
16 print "Resto:", restdiv
```

☒ Result

**Executing the program....**  
\$python2.7 main.py

Resultados

Soma: 7  
Subtracao: 3  
Dicisao: 2  
Resto: 1



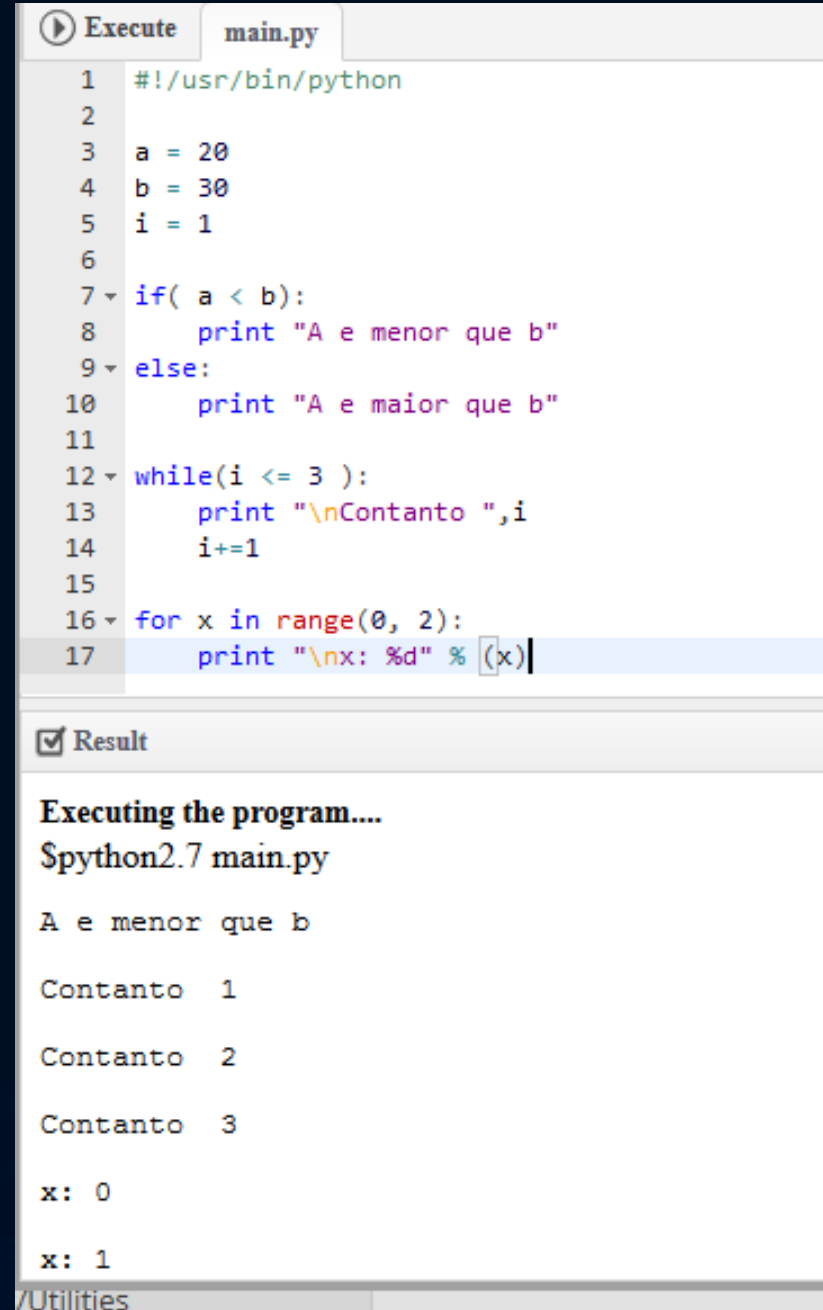
# Programa 5 – Estruturas condicionais

- Chamamos de estrutura condicional as instruções para testar se uma condição é verdadeira ou não.
- Estas estruturas condicionais podem ser associados com as estruturas que se repetem, após o cumprimento da condição, chamamos essas estruturas de repetição.

( if, while, for )

# Programa 5 – Operadores aritméticos e estruturas de repetição

PYTHON – EXER05.PY

A screenshot of a Python IDE window titled 'main.py'. The code is as follows:

```
1  #!/usr/bin/python
2
3  a = 20
4  b = 30
5  i = 1
6
7  if( a < b):
8      print "A e menor que b"
9  else:
10     print "A e maior que b"
11
12  while(i <= 3 ):
13     print "\nContanto ",i
14     i+=1
15
16  for x in range(0, 2):
17     print "\nx: %d" % (x)
```

The 'Execute' button is highlighted. Below the code editor, the 'Result' tab is active, showing the output of the program:

```
Executing the program....
Spython2.7 main.py

A e menor que b

Contanto  1

Contanto  2

Contanto  3

x: 0

x: 1
```

The status bar at the bottom indicates the file is located at '/Utilities'.

# Programa 6 – INPUT de parâmetros

## PYTHON – EXERo6.PY

Python Shell - C:\Users\Adilmar Dantas\Desktop\input.py

File Edit Format Run Options Windows Help

```
# -*- coding: cp1252 -*-
nome = raw_input("Digite seu nome: ")
print "\nOiii,",nome," vamos conversar..."
ano = raw_input("\nEm que ano você nasceu ? ")
print "\nEntão, você tem ",2017-int(ano), "anos hoje."
```

Python Shell

File Edit Shell Debug Options Windows Help

Python 2.7 (r27:82525, Jul 4 2010, 07:32)

Type "copyright", "credits" or "license()"

>>> ===== RESTART: Interactive Shell =====

>>>

Digite seu nome: adilmar

Oiii, adilmar vamos conversar...

Em que ano você nasceu ? 1992

Então, você tem 25 anos hoje.

>>> |

Vamos praticar ....

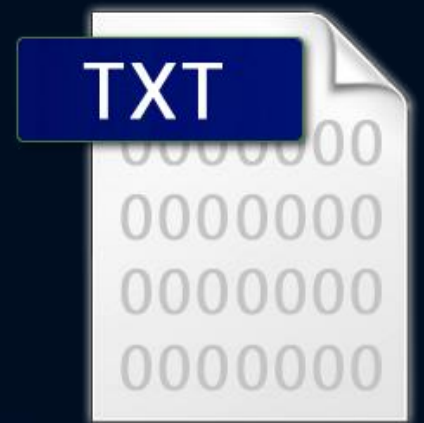
Python – exero7.py

Desenvolva uma calculadora com as operações básicas ( + , - , \* , / , <sup>2</sup> )

\*Dica: use as estruturas ou a função input

# Programa 8 – Manipulando arquivos .txt

- Já imaginou ter que comunicar um equipamento eletrônico e salvar suas saídas, por exemplo com Arduino.
- Por se tratar de uma tarefa relativamente simples podemos utilizar um arquivo “.txt” ou qualquer outra extensão para gravar estes dados, vejamos os passos.
- Definir o arquivo: `fp = open(dados.txt')`
- Criar uma constante para ler as linhas do file: `lines = fp.read().split("\n")`
- Fechar o arquivo: `fp.close()`





# Programa 8 – Lendo arquivos

## PYTHON – EXERo8.PY

7% arquivos.py - C:\Users\Adilmar Dantas\Desktop\arquivos.py

File Edit Format Run Options Windows Help

```
fp = open('C://dados.txt') # abre o arquivo
lines = fp.read().split("\n") # cria uma lista
print(lines)
fp.close() # close file
```

```
print "\nFazendo de outra maneira\n"
```

```
with open('C://dados.txt') as fp:
    for line in fp:
        print line
```

7% Python Shell

File Edit Shell Debug Options Window

Python 2.7 (r27:82525, Jul 4 2008) [AMD64]  
32

Type "copyright", "credits" or

>>> =====

>>>

['adilmar 25', 'pedro 33']

Fazendo de outra maneira

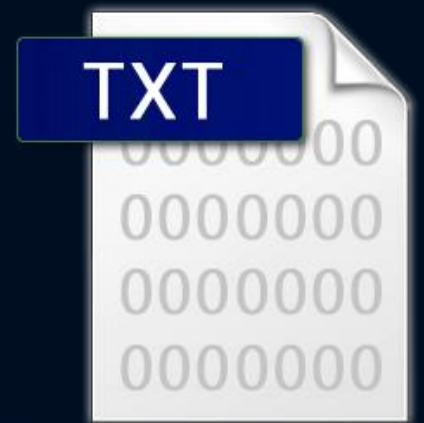
adilmar 25

pedro 33

>>> |

# Programa 9 – Manipulando arquivos .txt - escrita

- Assim como fizemos para ler o arquivo podemos criar um arquivo e escrever nele nossas informações, veja.
- Definir o arquivo: `file = open("testfile.txt","w")`
- Escrever no file: `file.write("Hello World")`



# Programa 9 – Escrevendo arquivos

PYTHON – EXER09.PY

```
7% exer9.py - C:/Users/Adilmar Dantas/Desktop/exer9.py
File Edit Format Run Options Windows Help
file = open("dados.txt", "w")

file.write("Nome: Adilmar\n")
file.write("Telefone: 99309-0377\n")
file.write("Aniversário: 02/02\n")
file.write("Profissão: Analista Sênior\n")
print("Salvo")

file.close()
```

```
7% Python Shell
File Edit Shell Debug Options Wind
Python 2.7 (r27:82525, Jul 4
32
Type "copyright", "credits" or
>>> =====
>>>
Salvo
>>> |
```

# Programa 10 – Criando uma agenda

- Agora vamos criar uma agenda de contatos com os seguintes métodos (inserir e listar) e os parâmetros :
- Nome
- Celular
- Fixo
- Email
- Idade (deve ser calculado de maneira automática a partir do ano de nascimento)
- Delimitador (------)
- Obs: criar um arquivo com o nome agenda.txt para gravar os dados



# Programa 10 – Agenda eletrônica

PYTHON – EXER10.PY

```
7% agenda.py - D:\Dropbox\Doutorado\POSMEC\codes\agenda.py
File Edit Format Run Options Windows Help

op='0'

print ("::~::~:Agenda Eletronica::~:~::~:")
print ("\nLISTA DE OPÇÕES\n1-Inserir\n2-Listar")

while (op=='0'):
    opcao = raw_input("\nDigite uma opção: ")

    if (opcao=='1'):
        file = open("agenda.txt", "a")
        op='1'
        print("\nOPÇÃO SELECIONADA: INSERIR CONTATO")
        nome = raw_input("\nInforme o nome: ")
        celular = raw_input("Informe o celular: ")
        fixo = raw_input("Informe o Fixo: ")
        email = raw_input("Informe o email: ")
        ano = raw_input("Informe o ano de nascimento: ")
        idade= 2017 - int(ano);

        file.write("Nome: "      +nome      +"\n")
        file.write("Celular: "   +celular   +"\n")
        file.write("Fixo: "      +fixo      +"\n")
        file.write("email: "     +email     +"\n")
        file.write("Idade: "     +str(idade) +"\n")
        file.write("-----\n")
        file.close()
        op='0'

    if (opcao=='2'):
        file = open("agenda.txt", "a")
        op='1'
        print("\nOPÇÃO SELECIONADA: LISTAR CONTATOS")
        with open("agenda.txt") as agenda:
            for contato in agenda:
                print contato
        file.close()
```

7% Python Shell

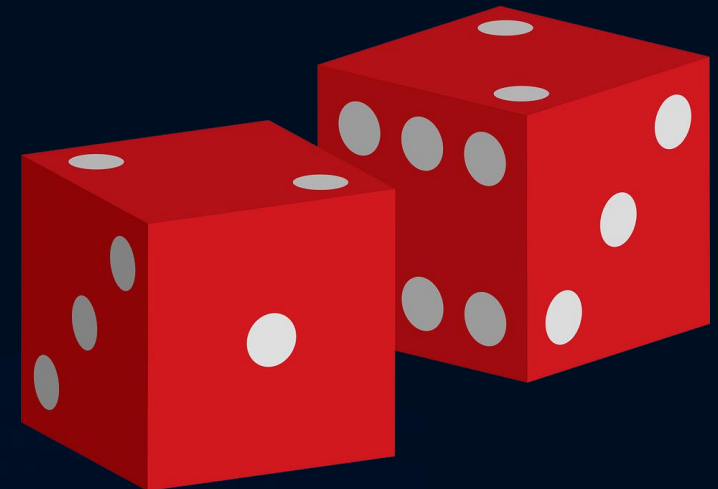
File Edit Shell Debug Options Windows Help

Python 2.7 (r27:82525, Jul 4 2010, 07:43:08) [MSC v.1500 64 32  
Type "copyright", "credits" or "license()" for more informat  
>>> ===== RESTART =====  
>>>  
::~::~:Agenda Eletronica::~:~::~:  
  
LISTA DE OPÇÕES  
1-Inserir  
2-Listar  
  
Digite uma opção: 1  
  
OPÇÃO SELECIONADA: INSERIR CONTATO  
  
Informe o nome: José  
Informe o celular: 9999999  
Informe o Fixo: 333333  
Informe o email: jose@bol.com.br  
Informe o ano de nascimento: 1968  
  
Digite uma opção: 2  
  
OPÇÃO SELECIONADA: LISTAR CONTATOS  
Nome: adilmair  
  
Celular: 1  
  
Fixo: 2  
  
email: 3  
  
Idade: 25  
  
-----  
  
Nome: pedro  
  
Celular: 2



# Desafio, jogo simples !

- Agora que você já conhece a maioria dos conceitos de programação em Python que tal desenvolvermos um jogo de adivinhar o número sorteado pelo computador.
- Para isso vamos conhecer uma função reservada para fazer este sorteio, com as seguintes instruções.
- Precisa ser um número inteiro
- Compreendido de 1 a 100
- Para isso : `sorteado = randint(1, 100)`



# Contatos



Site: [www.adilmar.com.br](http://www.adilmar.com.br)

Email: [akanehar@gmail.com](mailto:akanehar@gmail.com)

Lattes: <http://lattes.cnpq.br/2462384793631673>

