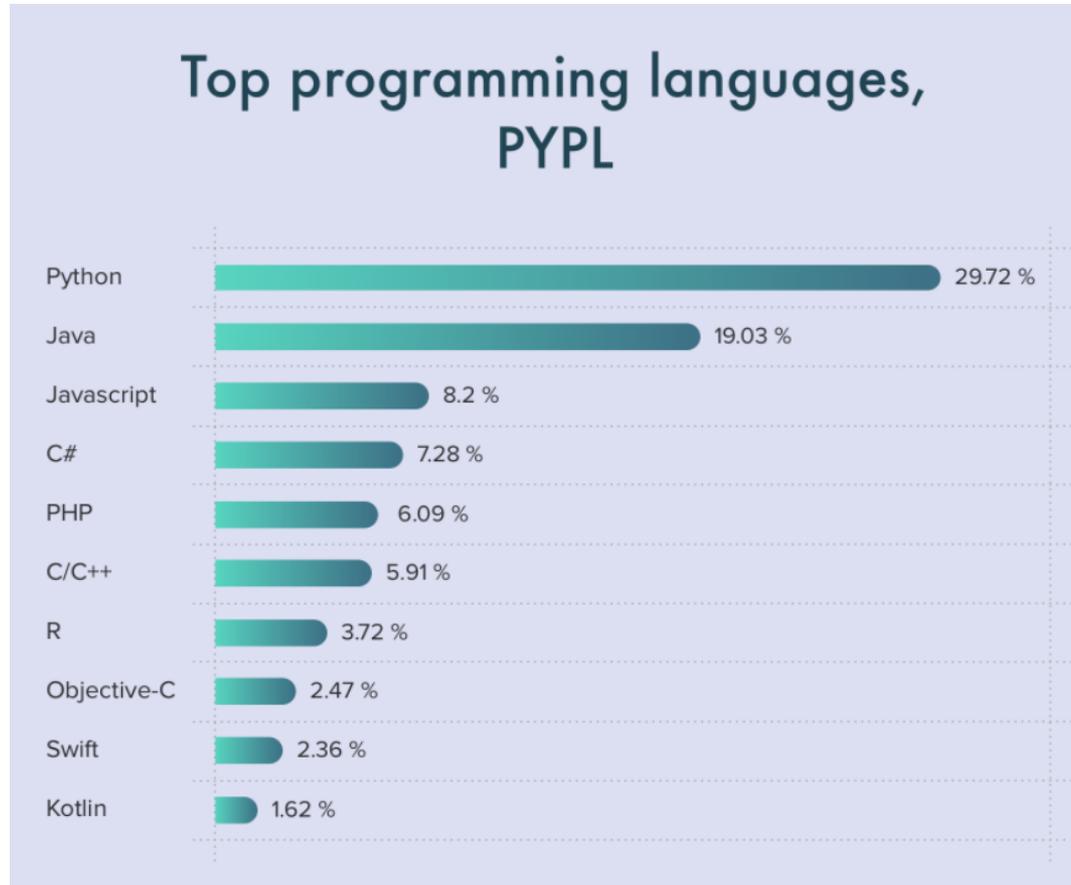




python™

Introdução à Programação em Python

Porquê Python?



Popularity of Programming Language (01-2021):

<https://www.cleveroad.com/blog/programming-languages-ranking>

Documentação

WWW!

w3schools.com

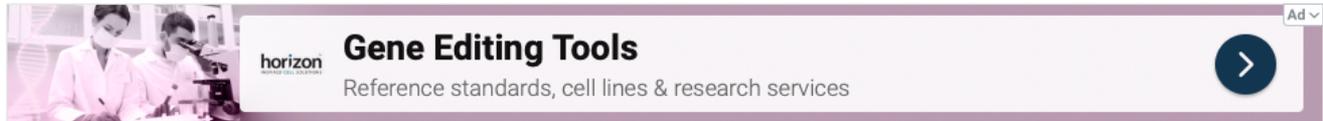
THE WORLD'S LARGEST WEB I

HTML CSS JAVASCRIPT SQL PYTHON PHP BOOTSTRAP HOW TO MORE REFERENCES EXERCISE

Python Tutorial

Python HOME

- Python Intro
- Python Get Started
- Python Syntax
- Python Comments
- Python Variables
- Python Data Types
- Python Numbers
- Python Casting
- Python Strings
- Python Booleans
- Python Operators
- Python Lists
- Python Tuples
- Python Sets
- Python Dictionaries
- Python If...Else
- Python While Loops
- Python For Loops
- Python Functions
- Python Lambda



horizon **Gene Editing Tools**
Reference standards, cell lines & research services

Python Tutorial

< Home

Next >

Python is a programming language.

Python can be used on a server to create web applications.

Start learning Python now »

Learning by Examples

With our "Try it Yourself" editor, you can edit the code and view the result.

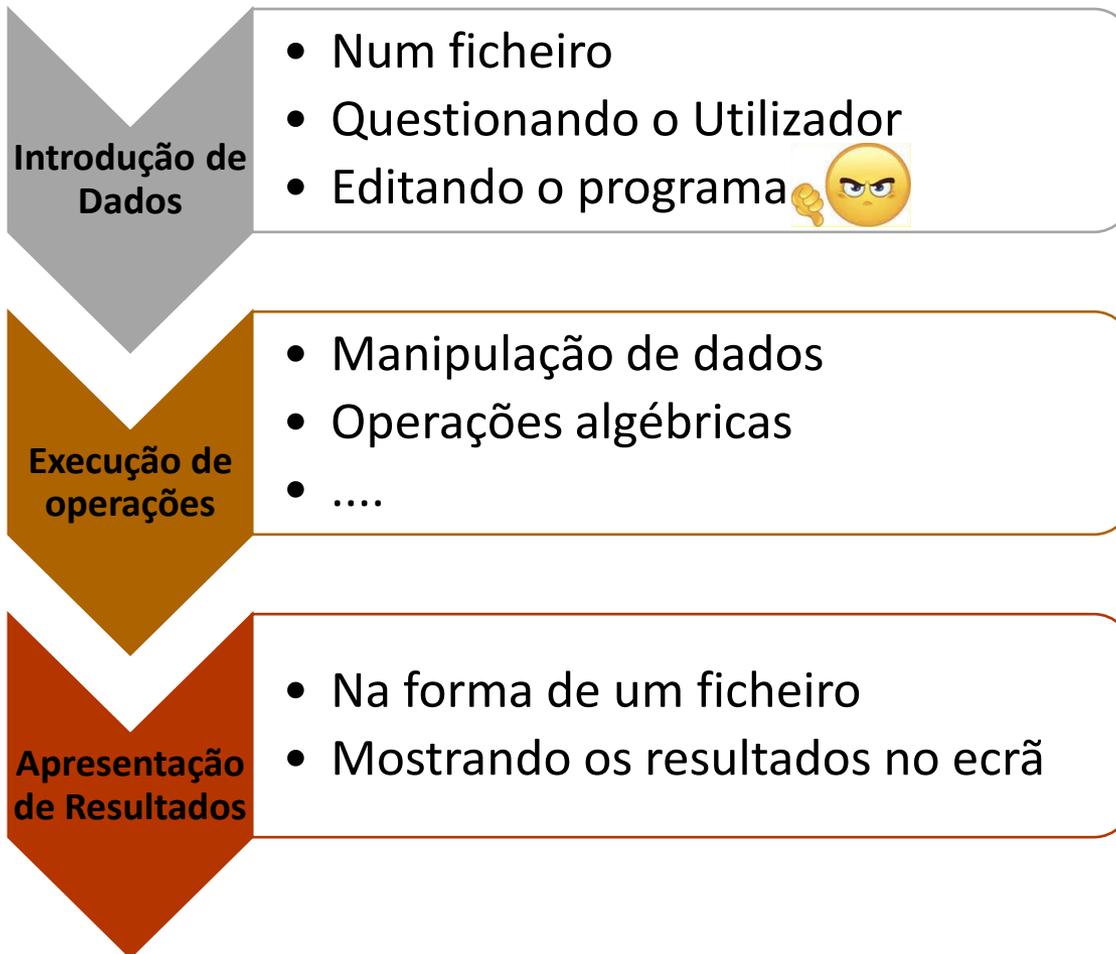
<https://www.w3schools.com/python/default.asp>

O que é um Programa?

É uma sequencia de instruções que realizam uma série de operações, devolvendo no final um resultado ao utilizador.

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

Estrutura de um Programa:



Variáveis

Em programação, uma variável é um objeto (uma posição, frequentemente localizada na memória) capaz de reter e representar um valor ou expressão.

Em Python, estas são iniciadas na primeira vez que são utilizadas, podendo conter texto, números inteiros, reais, ou ser booleanas.

Variáveis

`x = 5` (inteiro)

`y = "Joao"` (texto)

`Z = 0.5` (número de virgula flutuante)

`Z1 = Z2 = 6`

`Flag = True` (booleano)

Para pedir um valor para uma variável, `v`, ao utilizador pode usar-se:

`v = input()`

Operadores Aritméticos

Operador	Função	Exemplo
+	Adição	$x + y$
-	Subtração	$x - y$
*	Multiplicação	$x * y$
/	Divisão	x / y
**	Expoente	$x ** y$
%	Resto	$x \% y$

Ex: Determinar se um ano é bissexto: 2016, 2020, 2024, ...

Múltiplos de 4

Logo se o resto de $\text{ano}/4 \Rightarrow 0$, é bissexto.

Operadores de Atribuição

Operador	Exemplo	Igual a:
=	$x = 5$	$x = 5$
+=	$x += 3$	$x = x + 3$
-=	$x -= 3$	$x = x - 3$
*=	$x *= 3$	$x = x * 3$
/=	$x /= 3$	$x = x / 3$
%=	$x \% = 3$	$x = x \% 3$
**=	$x ** = 3$	$x = x ** 3$

Testes Lógicos

Função **IF**:

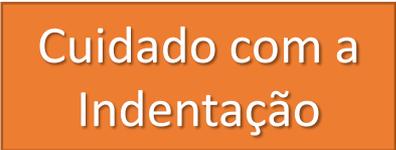
if $V > 0$:

↔ # Faz algo se a condição é verdadeira

else:

↔ # Faz algo se a condição é falsa

Cuidado com a
Indentação



Operadores de Comparação

Operador	Nome	Exemplo
==	Igual	$x == y$
!=	Diferente	$x != y$
>	Maior que	$x > y$
<	Menor que	$x < y$
>=	Maior ou igual	$x >= y$
<=	Menor ou igual	$x <= y$

Outros Operadores

Operador	Descrição	Exemplo
and	Devolve “true” se duas condições são verdadeiras	$x > 5$ and $x < 10$
or	Devolve “true” se uma das condições for verdadeiras	$x < 5$ or $x > 10$
not	Reverso do resultado, i.e., “False” se a condição é verdadeira	not($x < 5$ and $x > 10$)
is	Devolve “true” se os valores são iguais	x is y
is not	Devolve “true” se os valores não são iguais	x is not y

Arrays

Arrays são utilizados para guardar múltiplos valores numa única variável.

```
Numero[0] = 5.6
```

```
carro = ["Ford", "Volvo", "BMW"]
```

```
Numero = [1.0, 2.0, 3.1]
```

Arrays

Quando necessários, os valores são “chamados” indicando a sua posição no array.

```
carro = ["Ford", "Volvo", "BMW"]
```

```
print(carro[1]) → Devolve "Volvo"
```

Atenção: os índices dos valores num array são numerados entre 0 e N-1, onde N é o tamanho do array.

Para saber o tamanho de um array usa-se:

```
len(carro)
```

Arrays Multidimensionais

Array's NxM:

$$A = \begin{matrix} \overbrace{[1, 2]}^{N=0} & \overbrace{[3, 4]}^{N=1} \\ \uparrow & \uparrow & \uparrow & \uparrow \\ M = 0 & 1 & 0 & 1 \end{matrix}$$

Neste caso os valores são utilizados como:

`print("A11 = ", A[1][1])` → Devolve "4"

Arrays

No entanto, na maioria das vezes não se sabe quantos valores serão colocados no array antes de iniciar o programa.

Nestes casos, utiliza-se:

`n = 2`

`D = [[0] for i in range(n)]` → Array com N entradas

`C = [[0] * n for i in range(n)]` → Array N x N

Ciclos

Ciclos **for**:

```
for i in range(5):  
    Faz algo!!!
```

Ciclos **while**:

```
xi = 0.0 ; xf = 3.0 ; step = 0.05
```

```
while xi <= xf:  
    Faz algo !!!  
    xi += step
```

Cuidado com a Indentação!!!

Este código:

```
xi = 0.0 ; xf = 3.0 ; step = 0.05
while xi <= xf:
    print (xi)
    xi += step
```

Não é igual a:

```
xi = 0.0 ; xf = 3.0 ; step = 0.05
while xi <= xf:
print (xi)
xi += step
```

Gravar Ficheiros

Gravar dados num ficheiro:

w – escreve o ficheiro
r – lê o ficheiro
a – adiciona dados a
ficheiro já existente

1) Abrir o ficheiro:

```
f = open("testfile.txt", "w")
```

Nome do ficheiro

2) Gravar linha de texto:

```
f.write(str(xi) + "\t" + str(valor) + "\n")
```

tab

“Fecha” a linha

3) Fechar o ficheiro:

```
f.close()
```

Converte número para texto.
A função inversa é **float**

Funções Matemáticas

É necessário importar bibliotecas de matemática. Para o efeito, no início do programa adicionar:

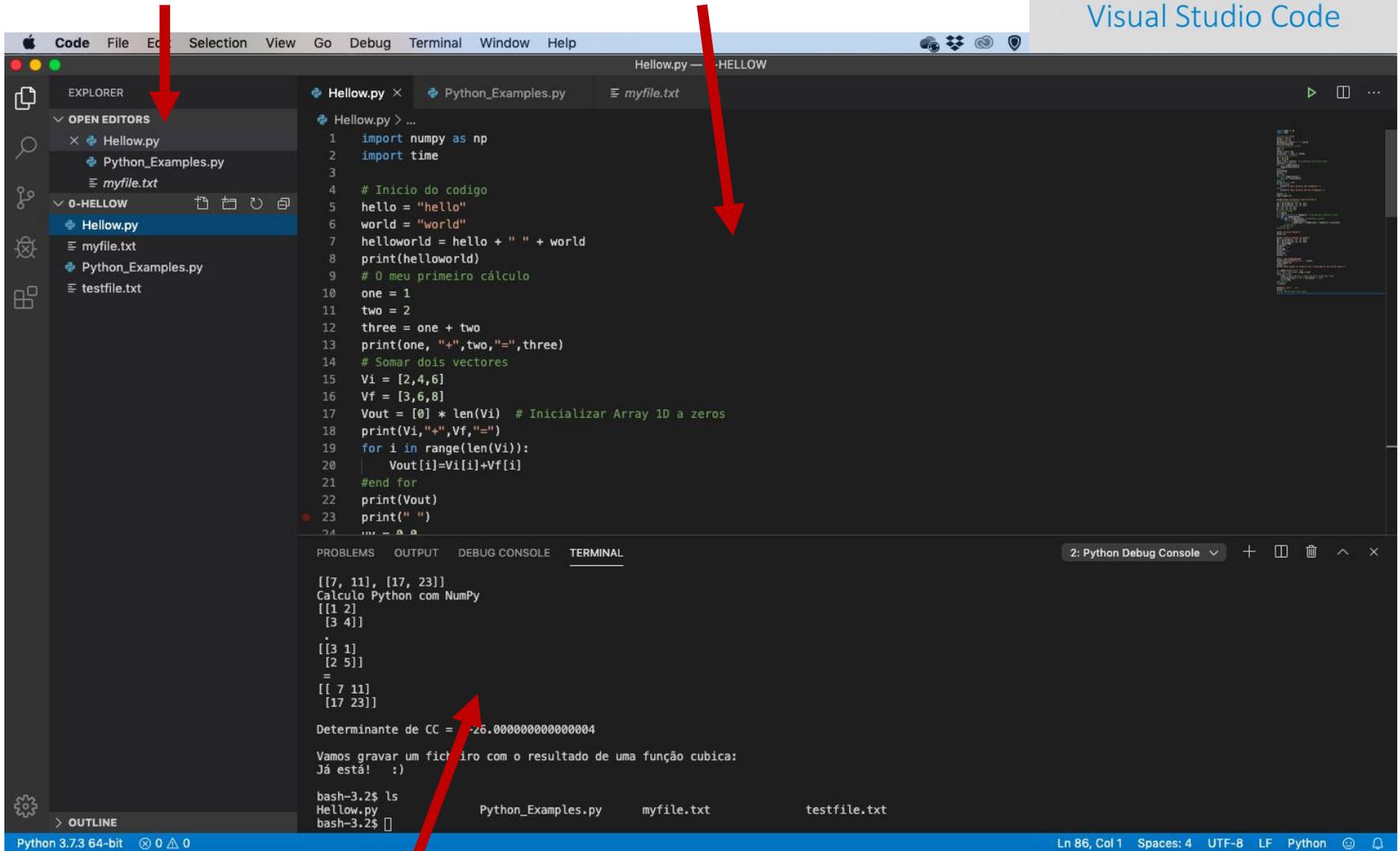
```
import math
```

Por exemplo, utilizar as funções como:

<code>math.cos(x)</code>	Cosseno de x , em radianos
<code>math.sin(x)</code>	Seno de x , em radianos
<code>math.acos(x)</code>	Arco cosseno de x , em radianos
<code>math.asin(x)</code>	Arco seno de x , em radianos
<code>math.tan(x)</code>	Tangente de x , em radianos
<code>math.atan(x)</code>	Arco tangente de x , em radianos
<code>math.exp(x)</code>	Exponencial de base e
<code>math.log(x[, base])</code>	Logaritmo de x (para a base e)
<code>math.sqrt(x)</code>	Raiz quadrada de x

Gestor de Ficheiros

Editor de Programas



Terminal e Consola de Erros

<https://code.visualstudio.com/>

Como criar um programa no VS Code

1) Abrir o VS Code e criar um diretório, por exemplo
“PROGRAMAS”

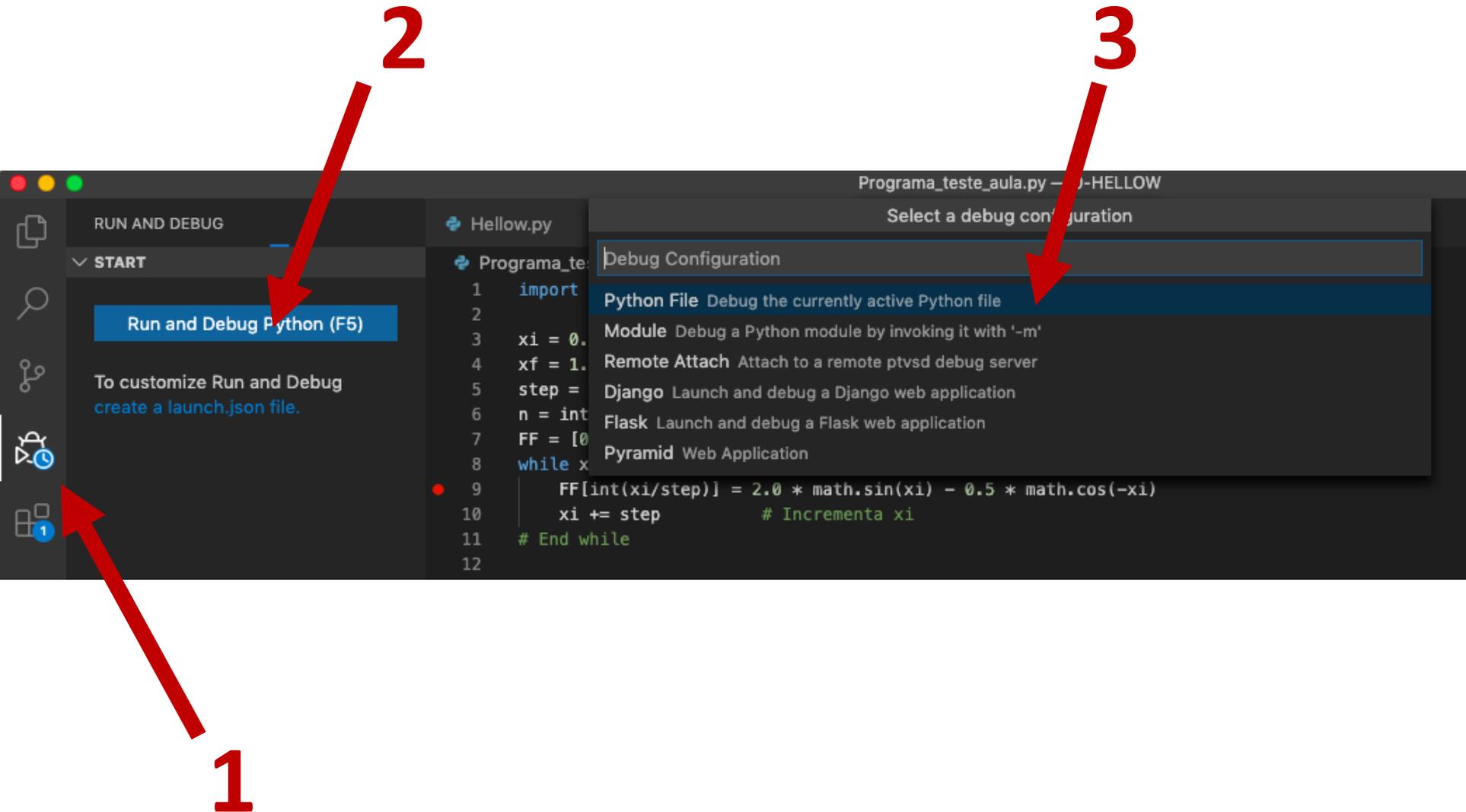
2) Criar um ficheiro “NOME_A_ESCOLHA.py” e escreva
o código:

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

3) Corra o programa.

ATENÇÃO: o nome do ficheiro deve terminar em “.py”

DEBUG

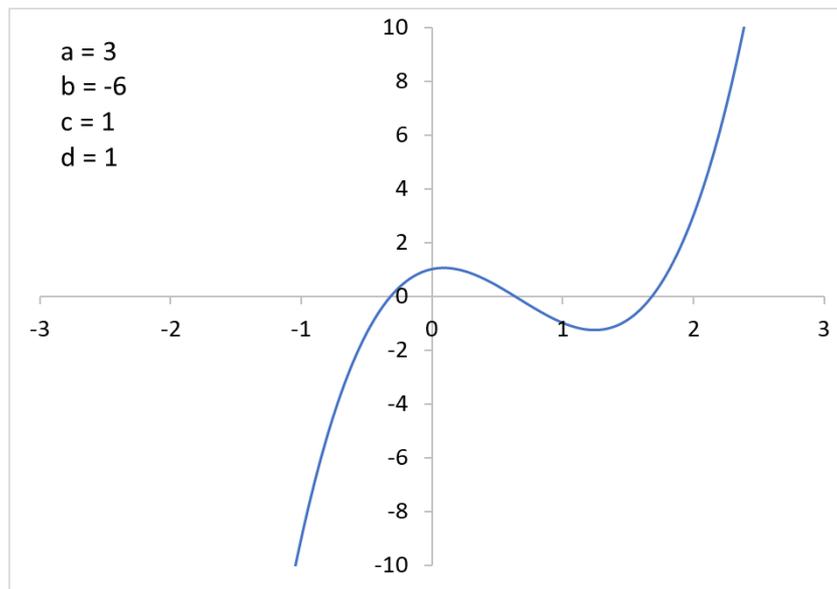


EXEMPLO

Considere a equação:

$$ax^3 + bx^2 + cx + d = 0$$

Quais são os valores das raízes da equação anteriores?



EXEMPLO

Método iterativo:

É um procedimento que gera uma sequência de soluções aproximadas que vão melhorando conforme iterações são executadas.

$$ax^3 + bx^2 + cx + d = 0$$

$$x[ax^2 + bx + c] = -d$$

$$x = \frac{-d}{\underbrace{ax^2 + bx + c}_{(1)}}$$

1. Dá-se uma primeira aproximação para o valor de x .
2. Calcula-se o novo valor de x utilizando a expressão (1).
3. Verifica-se se o valor de x já convergiu. Se sim, termina o cálculo. Caso contrário, retoma-se ao ponto 2.

CÓDIGO

```
# Programa em Python para calcular raízes de uma equação de
# terceiro grau:
#
#          aX^3 + bx^2 + cx + d = 0
#
# Para utilizar o método iterativo, reescrevemos a equação como:
#
#          x = -d / [ ax^2 + bx + c ]

import math # Importa as bibliotecas de matemática

# 1- Definimos as variáveis do polinómio e solicitamos ao utilizador o valor inicial de x.

a, b, c, d = 3.0, -6.0, 1.0, 1.0
print ("Qual o valor inicial de x? ") # Pergunta pelo valor inicial de x
xi = float(input())

# 2- Estabeleço o erro para a minha determinação e o número máximo de iterações.

Erro = 0.0000001 # Erro
Max_iter = 1000 # número maximo de iterações
Contador = 0 # Variável que sair de ciclos infinitos

# 3- Faz-se a primeira avaliação de x

x = -d / ( a * xi**2 + b * xi + c )

# 3a- Coloca os valores no ecrã

print(' Iter      x      Diff')
print('{0:5d} {1:5.8f} {2:5.8f}'.format(Contador, x, abs(x-xi)))

# 4- Determina a raiz

# 4a- Gravar dados num array
x_val = [0] * Max_iter # Aloca a memoria necessária para o array x_val
x_val[Contador] = x
```

```
while abs(xi-x) > Erro:
    xi = x
    x = -d / ( a * xi**2 + b * xi + c )

    Contador += 1
    x_val[Contador] = x

    print('{0:5d} {1:5.8f} {2:5.8f}'.format(Contador, x, abs(x-xi)))

    if Contador > Max_iter: # Verifica a validade do step introduzido
        print("Número máximo iterações excedido!")
        quit() # Termina o programa

# 5. Encontrou a raiz e comunica o resultado
print("...")
txt = "O valor é {0:5.6f}."
print(txt.format(x))

# 6. Grava os dados num ficheiro

f = open("dados.txt", "w") # Abre o ficheiro
for i in range(Contador): # Vai correr todos os valores no array
    # Escreve cada linha com o valor de x e f(x)
    if i == 0:
        f.write(str(i) + "\t" + str(x_val[i]) + "\t" + "\n")
    else:
        f.write(str(i) + "\t" + str(x_val[i]) + "\t" + str(abs(x_val[i]-x_val[i-1])) + "\n")
```

Exercício:

Construa uma calculadora em Python, que pede dois valores, uma operação, e devolve o resultado ao utilizador.