

# Curso completo de Python!





# **Instrutor:Vitor Mazuco**

http://facebook.com/vitormazuco

Email:vitor.mazuco@gmail.com

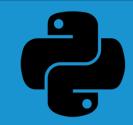
WebSite:

http://vmzsolutions.com.br

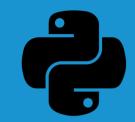


O primeiro exemplo usual para sistemas GUI é mostrar como exibir uma mensagem de "Olá Mundo!" em uma janela usando apenas quatro linhas em Python. Você pode ver que o layout da janela executado em sistemas Windows, Mac e no Linux são diferentes, mas seu funcionamento são iguais.

<sup>\*</sup> Veja mais no arquivo gui1.py



- O gui1.py é um script trivial, mas ilustra as etapas comuns à maioria dos Programas tkinter. Esse código Python faz o seguinte:
- → Carrega uma classe de widget do módulo tkinter;
- → Faz uma instância da classe importada *Label*;
- → Packs (pacotes) o novo Label no seu widget "pai";
- → Chama o mainloop para abrir a janela e iniciar o loop de eventos do tkinter



Como você poderia esperar, há uma variedade de maneiras de codificar o exemplo gui1.py. Por exemplo, se você quiser tornar todas as suas importações tkinter mais explícitas em seu script, pegue o módulo inteiro e prefixe todos os seus nomes com o nome do módulo.

\* Veja mais no arquivo gui1b.py



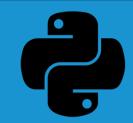
A constante TOP aqui, deixa o aplicativo no topo quando é executado, e também, temos a opção de aumentar o seu tamanho com o uso do *geometry*.

\* Veja mais no arquivo gui1c.py e gui1d.py



Para adicionar botões e callbacks, criamos uma instância da classe tkinter *Button*. É anexado à janela de nível superior padrão. Mas nós definimos uma opção de chamada de comando para a função sys.exit.

<sup>\*</sup> Veja mais no arquivo gui2.py



Nesse nosso script, quando o botão é pressionado, o tkinter chama a função *quit* que exibe uma mensagem no console e depois sai do programa com o uso do comando sys.exit.

\* Veja mais no arquivo gui3.py