

```
[1]: import tkinter as tk
     from tkinter import ttk
```

1. `import tkinter as tk`:

- Esta linha importa o módulo `tkinter` e o renomeia como `tk`. O `tkinter` é a biblioteca padrão do Python para criar interfaces gráficas de usuário (GUIs). Ao usar `as tk`, você pode referenciar os componentes do `tkinter` com o prefixo `tk`, o que torna o código mais conciso.

2. `from tkinter import ttk`:

- Esta linha importa o submódulo `ttk` do `tkinter`. O `ttk` (Themed Tk) fornece um conjunto de widgets com temas que têm uma aparência mais moderna e consistente em diferentes plataformas. Ele inclui widgets como `Combobox`, `Treeview`, `Notebook`, entre outros.

```
def sair():
    root.destroy()
```

Explicação:

1. `def sair():`:

- Esta linha define uma nova função chamada `sair`. Em Python, a palavra-chave `def` é usada para declarar uma função. O nome da função é `sair` e não recebe nenhum parâmetro.

2. `root.destroy()`:

- Dentro da função `sair`, chamamos o método `destroy` do objeto `root`. O `root` é a janela principal da aplicação `tkinter`.
- `root.destroy()`: Este método fecha a janela principal e encerra o loop principal da interface gráfica. Isso efetivamente termina a aplicação.

Contexto:

A função `sair` é usada como um comando para os widgets que precisam fechar a aplicação, como um botão "Sair" ou um item de menu "Sair". Ao associar `command=sair` a esses widgets, você garante que, quando o usuário clicar neles, a função `sair` será chamada e a aplicação será fechada de maneira controlada.

```
[3]: # Menu
     menu = tk.Menu(root)
     root.config(menu=menu)
     submenu = tk.Menu(menu)
     menu.add_cascade(label="Arquivo", menu=submenu)
     submenu.add_command(label="Sair", command=root.quit)
```

Explicação:

1. `menu = tk.Menu(root)` :
 - Esta linha cria um objeto `Menu` associado à janela principal (`root`). O `Menu` é um widget que permite adicionar menus suspensos à janela.
2. `root.config(menu=menu)` :
 - Aqui, o menu criado é configurado como o menu principal da janela (`root`). Isso faz com que o menu apareça na barra de menus da janela.
3. `submenu = tk.Menu(menu)` :
 - Esta linha cria um submenu que será adicionado ao menu principal. O submenu é um menu dentro de outro menu, permitindo a criação de hierarquias de opções.

4. `menu.add_cascade(label="Arquivo", menu=submenu)` :
 - Esta linha adiciona o `submenu` ao `menu` principal com o rótulo "Arquivo". O método `add_cascade` é usado para criar um item de menu que, quando clicado, exibe o submenu associado.
5. `submenu.add_command(label="Sair", command=root.quit)` :
 - Aqui, um comando é adicionado ao `submenu` com o rótulo "Sair". O parâmetro `command=root.quit` especifica que, quando o item "Sair" for clicado, o método `quit` da janela principal (`root`) será chamado, fechando a aplicação.

Resumo:

Este trecho de código cria um menu na janela principal com um item "Arquivo". Dentro deste item, há uma opção "Sair" que, quando clicada, fecha a aplicação. Isso é útil para adicionar funcionalidades de menu padrão, como "Arquivo", "Editar", etc., em uma interface gráfica.

```
•[4]: # Campos de formulário
tk.Label(root, text="Nome:").grid(row=0, column=0, padx=10, pady=10)
entry_nome = tk.Entry(root)
entry_nome.grid(row=0, column=1, padx=10, pady=10)

tk.Label(root, text="Idade:").grid(row=1, column=0, padx=10, pady=10)
entry_idade = tk.Entry(root)
entry_idade.grid(row=1, column=1, padx=10, pady=10)
```

Explicação:

1. `tk.Label(root, text="Nome:").grid(row=0, column=0, padx=10, pady=10)` :
 - `tk.Label(root, text="Nome:")` : Cria um rótulo (label) com o texto "Nome:" na janela principal (`root`).
 - `.grid(row=0, column=0, padx=10, pady=10)` : Posiciona o rótulo na linha 0 e coluna 0 da grade (grid) da janela, com um preenchimento (padding) de 10 pixels nas direções horizontal (`padx`) e vertical (`pady`).
2. `entry_nome = tk.Entry(root)` :
 - Cria um campo de entrada de texto (entry) na janela principal (`root`) onde o usuário pode digitar o nome.

3. `entry_nome.grid(row=0, column=1, padx=10, pady=10) :`
 - Posiciona o campo de entrada de texto na linha 0 e coluna 1 da grade, ao lado do rótulo “Nome:”, com um preenchimento de 10 pixels nas direções horizontal e vertical.
4. `tk.Label(root, text="Idade:").grid(row=1, column=0, padx=10, pady=10) :`
 - Cria um rótulo com o texto “Idade:” na janela principal e o posiciona na linha 1 e coluna 0 da grade, com um preenchimento de 10 pixels nas direções horizontal e vertical.
5. `entry_idade = tk.Entry(root) :`
 - Cria um campo de entrada de texto na janela principal onde o usuário pode digitar a idade.
6. `entry_idade.grid(row=1, column=1, padx=10, pady=10) :`
 - Posiciona o campo de entrada de texto na linha 1 e coluna 1 da grade, ao lado do rótulo “Idade:”, com um preenchimento de 10 pixels nas direções horizontal e vertical.

Resumo:

Este trecho de código cria dois campos de formulário na interface gráfica: um para o nome e outro para a idade. Cada campo é composto por um rótulo (`Label`) e um campo de entrada de texto (`Entry`). Os elementos são organizados em uma grade (grid) para facilitar o layout, com preenchimento para espaçamento.

```
[ ]: # Radio buttons
tk.Label(root, text="Gênero:").grid(row=2, column=0, padx=10, pady=10)
genero_var = tk.StringVar()
tk.Radiobutton(root, text="Masculino", variable=genero_var, value="Masculino").grid(row=2, column=1, padx=10, pady=10)
tk.Radiobutton(root, text="Feminino", variable=genero_var, value="Feminino").grid(row=2, column=2, padx=10, pady=10)
```

Explicação:

1. `tk.Label(root, text="Gênero:").grid(row=2, column=0, padx=10, pady=10) :`
 - `tk.Label(root, text="Gênero:")` : Cria um rótulo (label) com o texto “Gênero:” na janela principal (`root`).
 - `.grid(row=2, column=0, padx=10, pady=10)` : Posiciona o rótulo na linha 2 e coluna 0 da grade (grid) da janela, com um preenchimento (padding) de 10 pixels nas direções horizontal (`padx`) e vertical (`pady`).
2. `genero_var = tk.StringVar() :`
 - Cria uma variável de controle do tipo `StringVar` que será usada para armazenar o valor selecionado pelos botões de rádio. Esta variável é necessária para que os botões de rádio possam compartilhar o mesmo grupo e armazenar o valor selecionado.

3. `tk.Radiobutton(root, text="Masculino", variable=genero_var, value="Masculino").grid(row=2, column=1, padx=10, pady=10)` :
 - o `tk.Radiobutton(root, text="Masculino", variable=genero_var, value="Masculino")` : Cria um botão de rádio com o texto "Masculino". Este botão de rádio está associado à variável `genero_var` e tem o valor "Masculino". Quando o botão é selecionado, `genero_var` será atualizado para "Masculino".
 - o `.grid(row=2, column=1, padx=10, pady=10)` : Posiciona o botão de rádio na linha 2 e coluna 1 da grade, com um preenchimento de 10 pixels nas direções horizontal e vertical.
4. `tk.Radiobutton(root, text="Feminino", variable=genero_var, value="Feminino").grid(row=2, column=2, padx=10, pady=10)` :
 - o `tk.Radiobutton(root, text="Feminino", variable=genero_var, value="Feminino")` : Cria um botão de rádio com o texto "Feminino". Este botão de rádio também está associado à variável `genero_var` e tem o valor "Feminino". Quando o botão é selecionado, `genero_var` será atualizado para "Feminino".
 - o `.grid(row=2, column=2, padx=10, pady=10)` : Posiciona o botão de rádio na linha 2 e coluna 2 da grade, com um preenchimento de 10 pixels nas direções horizontal e vertical.

Resumo:

Este trecho de código cria um grupo de botões de rádio para selecionar o gênero. Os botões de rádio são associados à variável `genero_var`, que armazena o valor selecionado ("Masculino" ou "Feminino"). Os elementos são organizados em uma grade (grid) para facilitar o layout, com preenchimento para espaçamento.

```
[ ]: # Checklists
tk.Label(root, text="Interesses:").grid(row=3, column=0, padx=10, pady=10)
var_python = tk.BooleanVar()
tk.Checkbutton(root, text="Python", variable=var_python).grid(row=3, column=1, padx=10, pady=10)
var_java = tk.BooleanVar()
tk.Checkbutton(root, text="Java", variable=var_java).grid(row=3, column=2, padx=10, pady=10)
var_javascript = tk.BooleanVar()
tk.Checkbutton(root, text="JavaScript", variable=var_javascript).grid(row=3, column=3, padx=10, pady=10)
```

Explicação:

1. `tk.Label(root, text="Interesses:").grid(row=3, column=0, padx=10, pady=10)` :
 - o `tk.Label(root, text="Interesses:")` : Cria um rótulo (label) com o texto "Interesses:" na janela principal (`root`).
 - o `.grid(row=3, column=0, padx=10, pady=10)` : Posiciona o rótulo na linha 3 e coluna 0 da grade (grid) da janela, com um preenchimento (padding) de 10 pixels nas direções horizontal (`padx`) e vertical (`pady`).
2. `var_python = tk.BooleanVar()` :
 - o Cria uma variável de controle do tipo `BooleanVar` que será usada para armazenar o estado (selecionado ou não) da caixa de seleção "Python".

3. `tk.Checkbutton(root, text="Python", variable=var_python).grid(row=3, column=1, padx=10, pady=10)` :
- `tk.Checkbutton(root, text="Python", variable=var_python)` : Cria uma caixa de seleção com o texto "Python". Esta caixa de seleção está associada à variável `var_python`, que armazenará `True` se a caixa estiver marcada e `False` se não estiver.
 - `.grid(row=3, column=1, padx=10, pady=10)` : Posiciona a caixa de seleção na linha 3 e coluna 1 da grade, com um preenchimento de 10 pixels nas direções horizontal e vertical.
4. `var_java = tk.BooleanVar()` :
- Cria uma variável de controle do tipo `BooleanVar` que será usada para armazenar o estado da caixa de seleção "Java".
5. `tk.Checkbutton(root, text="Java", variable=var_java).grid(row=3, column=2, padx=10, pady=10)` :
- `tk.Checkbutton(root, text="Java", variable=var_java)` : Cria uma caixa de seleção com o texto "Java". Esta caixa de seleção está associada à variável `var_java`.

- `.grid(row=3, column=2, padx=10, pady=10)` : Posiciona a caixa de seleção na linha 3 e coluna 2 da grade, com um preenchimento de 10 pixels nas direções horizontal e vertical.
6. `var_javascript = tk.BooleanVar()` :
- Cria uma variável de controle do tipo `BooleanVar` que será usada para armazenar o estado da caixa de seleção "JavaScript".
7. `tk.Checkbutton(root, text="JavaScript", variable=var_javascript).grid(row=3, column=3, padx=10, pady=10)` :
- `tk.Checkbutton(root, text="JavaScript", variable=var_javascript)` : Cria uma caixa de seleção com o texto "JavaScript". Esta caixa de seleção está associada à variável `var_javascript`.
 - `.grid(row=3, column=3, padx=10, pady=10)` : Posiciona a caixa de seleção na linha 3 e coluna 3 da grade, com um preenchimento de 10 pixels nas direções horizontal e vertical.

Resumo:

Este trecho de código cria um conjunto de caixas de seleção para os interesses "Python", "Java" e "JavaScript". Cada caixa de seleção está associada a uma variável `BooleanVar` que armazena o estado da caixa (marcada ou desmarcada). Os elementos são organizados em uma grade (grid) para facilitar o layout, com preenchimento para espaçamento.

```
•[1]: # Caixa de combinação
tk.Label(root, text="País:").grid(row=4, column=0, padx=10, pady=10)
combo_pais = ttk.Combobox(root, values=["Brasil", "Estados Unidos", "Canadá", "Austrália"])
combo_pais.grid(row=4, column=1, padx=10, pady=10)
```

Explicação:

1. `tk.Label(root, text="País:").grid(row=4, column=0, padx=10, pady=10):`
 - o `tk.Label(root, text="País:")`: Cria um rótulo (label) com o texto "País:" na janela principal (`root`).
 - o `.grid(row=4, column=0, padx=10, pady=10)`: Posiciona o rótulo na linha 4 e coluna 0 da grade (grid) da janela, com um preenchimento (padding) de 10 pixels nas direções horizontal (`padx`) e vertical (`pady`).
2. `combo_pais = ttk.Combobox(root, values=["Brasil", "Estados Unidos", "Canadá", "Austrália"]):`
 - o `ttk.Combobox(root, values=["Brasil", "Estados Unidos", "Canadá", "Austrália"])`: Cria uma caixa de combinação (combobox) na janela principal (`root`). A caixa de combinação permite ao usuário selecionar um valor de uma lista suspensa. Os valores disponíveis são "Brasil", "Estados Unidos", "Canadá" e "Austrália".
 - o `combo_pais`: A variável `combo_pais` armazena a referência à caixa de combinação criada.
3. `combo_pais.grid(row=4, column=1, padx=10, pady=10):`

- o Posiciona a caixa de combinação na linha 4 e coluna 1 da grade, ao lado do rótulo "País:", com um preenchimento de 10 pixels nas direções horizontal e vertical.

Resumo:

Este trecho de código cria uma caixa de combinação para selecionar um país. A caixa de combinação é acompanhada por um rótulo "País:". Os elementos são organizados em uma grade (grid) para facilitar o layout, com preenchimento para espaçamento. A caixa de combinação permite ao usuário escolher entre as opções "Brasil", "Estados Unidos", "Canadá" e "Austrália".

```
# Botões de comando
tk.Button(root, text="Sair", command=sair).grid(row=5, column=1, padx=10, pady=10)

# Inicia o loop principal da interface
root.mainloop()
```

Explicação:

1. `tk.Button(root, text="Sair", command=sair).grid(row=5, column=1, padx=10, pady=10):`
 - o `tk.Button(root, text="Sair", command=sair):`
 - `tk.Button`: Cria um botão na janela principal (`root`).
 - `root`: Especifica que o botão será adicionado à janela principal.
 - `text="Sair"`: Define o texto que será exibido no botão, que neste caso é "Sair".
 - `command=sair`: Associa a função `sair` ao botão. Quando o botão for clicado, a função `sair` será chamada, executando `root.destroy()` para fechar a aplicação.
 - o `.grid(row=5, column=1, padx=10, pady=10):`
 - `grid(row=5, column=1)`: Posiciona o botão na linha 5 e coluna 1 da grade (grid) da janela.

- `grid(row=5, column=1)` : Posiciona o botão na linha 5 e coluna 1 da grade (grid) da janela.
- `padx=10, pady=10` : Adiciona um preenchimento (padding) de 10 pixels nas direções horizontal (`padx`) e vertical (`pady`), criando espaço ao redor do botão.

2. `root.mainloop()` :

- `root.mainloop()` : Inicia o loop principal da interface gráfica. Este loop mantém a janela aberta e permite que a aplicação responda a eventos do usuário, como cliques de botão e entradas de teclado.
- Enquanto o loop principal está em execução, a aplicação permanece ativa e interativa. Quando o usuário clica no botão “Sair”, a função `sair` é chamada, o que destrói a janela principal e encerra o loop, fechando a aplicação.

Resumo:

Este trecho de código cria um botão “Sair” que, quando clicado, chama a função `sair` para fechar a aplicação. O botão é posicionado na linha 5 e coluna 1 da grade da janela, com um preenchimento de 10 pixels ao redor. O loop principal (`root.mainloop()`) mantém a janela aberta e interativa até que o botão “Sair” seja clicado.