**🧰 Funções e Métodos do Tkinter Utilizados**

**1. Tk()**

* **Descrição**: Cria a janela principal da aplicação.
* **Uso**: janela = tk.Tk()
* **Detalhes**: Inicializa a aplicação e é necessário para qualquer programa que utilize Tkinter.​

**2. Label**

* **Descrição**: Cria um rótulo (texto estático) na interface.
* **Uso**: tk.Label(janela, text="CPF\*")
* **Detalhes**: Utilizado para identificar campos de entrada ou fornecer informações ao usuário.​

**3. Entry**

* **Descrição**: Cria um campo de entrada de texto de uma linha.
* **Uso**: cpf\_entry = tk.Entry(janela, width=30)
* **Métodos Associados**:
  + .get(): Retorna o conteúdo atual do campo.
  + .delete(0, tk.END): Remove o conteúdo do campo.
  + .insert(0, "texto"): Insere "texto" no início do campo.​

**4. Button**

* **Descrição**: Cria um botão que pode executar uma função quando clicado.
* **Uso**: tk.Button(janela, text="Cadastrar", command=cadastrar)
* **Detalhes**: O parâmetro command associa o botão a uma função que será chamada ao ser clicado.​[Medium](https://medium.com/%40shovitkafle/python-tkinter-module-gui-5df7ba254a15?utm_source=chatgpt.com" \t "_blank)

**5. Text**

* **Descrição**: Cria uma área de texto multilinha.
* **Uso**: texto\_text = tk.Text(texto)
* **Métodos Associados**:
  + .insert(tk.END, "texto"): Insere "texto" ao final da área de texto.
  + .pack(): Adiciona o widget à janela.​[summet.com](https://www.summet.com/dmsi/html/guiProgramming.html?utm_source=chatgpt.com)

**6. Toplevel**

* **Descrição**: Cria uma nova janela secundária.
* **Uso**: texto = tk.Toplevel(janela)
* **Detalhes**: Utilizado para exibir informações adicionais sem interferir na janela principal.​

**7. messagebox**

* **Descrição**: Exibe caixas de diálogo para interações com o usuário.
* **Importação**: from tkinter import messagebox
* **Funções Comuns**:
  + messagebox.showinfo("Título", "Mensagem"): Exibe uma mensagem informativa.
  + messagebox.showwarning("Título", "Mensagem"): Exibe um aviso.
  + messagebox.showerror("Título", "Mensagem"): Exibe uma mensagem de erro.​

**8. grid()**

* **Descrição**: Gerenciador de layout que organiza widgets em uma grade (linhas e colunas).
* **Uso**: widget.grid(row=0, column=0)
* **Parâmetros Comuns**:
  + row: Linha onde o widget será colocado.
  + column: Coluna onde o widget será colocado.
  + padx: Espaçamento horizontal.
  + pady: Espaçamento vertical.
  + sticky: Alinhamento dentro da célula (ex: "w", "e", "n", "s").​

**9. pack()**

* **Descrição**: Gerenciador de layout que empacota widgets em ordem.
* **Uso**: widget.pack()
* **Parâmetros Comuns**:
  + fill: Define se o widget deve expandir para preencher o espaço disponível (ex: tk.BOTH).
  + expand: Permite que o widget expanda para preencher o espaço extra.​[mycodenotein.netlify.app+7Ohidur's Blog+7Medium+7](https://blog.ohidur.com/posts/python/tkinter/?utm_source=chatgpt.com)

**10. mainloop()**

* **Descrição**: Inicia o loop principal da aplicação, mantendo a janela aberta e respondendo a eventos.
* **Uso**: janela.mainloop()
* **Detalhes**: Deve ser a última linha do programa para que a interface permaneça ativa.