

Aula 19

Radiobutton

O Radiobutton é um widget do tkinter que permite ao usuário selecionar apenas uma opção de um conjunto de opções. Ele é semelhante a um botão de opção, mas é usado quando há um conjunto de opções mutuamente exclusivas.

Para criar um Radiobutton, você precisa instanciar a classe `tkinter.Radiobutton` e configurar algumas de suas opções.

Por exemplo, para criar um `Radiobutton` que permita ao usuário escolher entre as opções "gremista" e "colorado" :

```
import tkinter as tk

def mostra_valor():
    lbl_mostra['text'] = escolha.get()

janela = tk.Tk()
escolha = tk.StringVar()
# usamos None como string para os botões estarem desmarcados
escolha.set('None')

lbl_mostra = tk.Label(janela)
btn_inter = tk.Radiobutton(janela)
btn_gremio = tk.Radiobutton(janela, text="gremista", variable=escolha,
                             value="gremista")
btn_verifica = tk.Button(janela, text="verifica", command=mostra_valor)

btn_inter['text'] = "colorado"
btn_inter['variable'] = escolha
btn_inter['value'] = "colorado"

btn_inter.pack()
btn_gremio.pack()
btn_verifica.pack()
lbl_mostra.pack()

janela.mainloop()
```

Aqui, criamos uma instância da classe `tk.Tk()`, que é a janela principal da aplicação. Em seguida, criamos uma instância da classe `tk.StringVar()` e o valor `None` como string, dessa forma, os botões ficarão desmarcados ao inicializarmos o programa.

Em seguida, criamos duas instâncias da classe `tk.Radiobutton`, uma para cada opção. Passamos a janela principal (`janela`) como primeiro argumento, o texto que deve ser exibido perto do botão de opção como segundo argumento, a variável que armazena o valor selecionado como terceiro argumento e o valor que essa opção deve armazenar como quarto argumento. Também criamos um `Button` e um `Label`, para nos ajudar a ver os valores selecionados.

Por último, usamos o método `pack()` para adicionar tudo à janela e o método `mainloop()` para iniciar o loop de eventos da aplicação.

Algumas opções adicionais que você pode configurar em um `Radiobutton` incluem :

- **`indicatoron`**: determina se o indicador circular de seleção do botão de opção deve ser exibido. O valor padrão é 1 (exibir indicador), mas você pode definir como 0 para esconder o indicador;

- selectcolor: define a cor do indicador de seleção quando o botão de opção estiver selecionado. O valor padrão é o padrão do sistema;
- activebackground: define a cor de fundo do botão de opção quando ele estiver ativo (quando o cursor do mouse estiver sobre ele);
- activeforeground: define a cor do texto do botão de opção quando ele estiver ativo;
- command: define uma função que será chamada quando o botão de opção for selecionado. Por exemplo, você pode usar essa opção para executar uma determinada ação quando o usuário seleciona uma opção específica;

Veja abaixo um exemplo usando o `command` para executar uma função sempre que qualquer um dos botões é clicado, mostrando o valor dele no terminal:

```
import tkinter as tk

def mostra_selecionado():
    print("Selecionado: ", escolha.get())

janela = tk.Tk()
escolha = tk.StringVar()
escolha.set('None')

btn_gremio = tk.Radiobutton(janela, text="gremista", variable=escolha,
                             value="gremista", command=mostra_selecionado)
btn_gremio.pack()

btn_inter = tk.Radiobutton(janela, text="colorado", variable=escolha,
                             value="colorado", command=mostra_selecionado)
btn_inter.pack()

janela.mainloop()
```

Algumas outras opções que você pode configurar em um `Radiobutton` incluem :

- font define a fonte usada para exibir o texto do botão de opção;
- width define a largura do botão de opção;
- height define a altura do botão de opção;
- state define o estado do botão de opção. Os valores possíveis são normal (ativo), disabled (desativado) e active (ativo). Quando um botão está desativado, ele não pode ser selecionado pelo usuário;
- image: define uma imagem que deve ser exibida no botão de opção;
- compound: determina como a imagem e o texto devem ser exibidos juntos no botão de opção;

Aqui está um exemplo de como usar algumas dessas opções:

```
import tkinter as tk

janela = tk.Tk()

escolha = tk.StringVar()
escolha.set('None')

btn_n64 = tk.Radiobutton(janela, text='Nintendo 64',
                          variable=escolha, compound="right", value="gremista")
btn_n64.pack()

btn_ps = tk.Radiobutton(
    janela,
    text="playstation",
    variable=escolha,
    value="playstation",
    font=("Arial", 12),
    state="disabled",
    width=20, height=10)
btn_ps.pack()

janela.mainloop()
```

Checkbutton

O Checkbutton é um widget do tkinter que permite ao usuário selecionar várias opções de um conjunto de opções. Ele é semelhante a um radiobutton, mas é usado quando há um conjunto de opções não mutuamente exclusivas.

Para criar um Checkbutton, você precisa instanciar a classe `tkinter.Checkbutton` e configurar algumas de suas opções.

Por exemplo, para criar um Checkbutton que permita ao usuário escolher entre as opções C#, Python e Java:

```
import tkinter as tk

janela = tk.Tk()
var1 = tk.IntVar()
var2 = tk.IntVar()
var3 = tk.IntVar()

btn_c = tk.Checkbutton(janela, text="C#", variable=var1)
btn_c.pack()

btn_python = tk.Checkbutton(janela, text="Python", variable=var2)
btn_python.pack()

btn_java = tk.Checkbutton(janela, text="Java", variable=var3)
btn_java.pack()

janela.mainloop()
```

Aqui, criamos uma instância da classe `tk.Tk()`, que é a janela principal da aplicação. Em seguida, criamos três instâncias da classe `tk.IntVar()` e atribuímos a três variáveis diferentes, `var1`, `var2` e `var3`. Essas variáveis são usadas para armazenar e recuperar o valor selecionado pelo usuário.

Em seguida, criamos três instâncias da classe `tk.Checkbutton`, uma para cada opção. Passamos a janela principal (`janela`) como primeiro argumento, o texto que deve ser exibido perto do botão de opção como segundo argumento e a variável que armazena o valor selecionado como terceiro argumento.

Por último, usamos o método `pack()` para adicionar os botões de opção à janela e o método `mainloop()` para iniciar o loop de eventos da aplicação.

Assim como no Radiobutton, você pode usar o método `grid()` ou `place()` para posicionar os botões de opção em vez de `pack()`.

Ao selecionar o botão, o valor da variável associada é 1, caso contrário é 0.

Algumas opções adicionais que você pode configurar em um Checkbutton incluem:

- onvalue: define o valor que a variável associada deve ter quando o Checkbutton está selecionado. O padrão é 1;
- offvalue: define o valor que a variável associada deve ter quando o Checkbutton não está selecionado. O padrão é 0;
- activebackground: define a cor de fundo do Checkbutton quando ele estiver ativo (quando o cursor do mouse estiver sobre ele);
- activeforeground: define a cor do texto do Checkbutton quando ele estiver ativo;
- selectcolor: define a cor do indicador de seleção quando o Checkbutton estiver selecionado. O valor padrão é o padrão do sistema;

- command: define uma função que será chamada quando o Checkbutton for selecionado ou desmarcado. Por exemplo, você pode usar essa opção para executar uma determinada ação quando o usuário seleciona uma opção específica;

Exemplo:

```
import tkinter as tk

def mostra_selecionado():
    print("C#:", var1.get())
    print("Python:", var2.get())
    print("Java:", var3.get())

janela = tk.Tk()

var1 = tk.IntVar(value=1)
var2 = tk.IntVar()
var3 = tk.IntVar()

btn_c = tk.Checkbutton(janela, text="C#", variable=var1, onvalue=1,
                        offvalue=0, activebackground='blue', command=mostra_selecionado)
btn_c.pack()
btn_python = tk.Checkbutton(janela, text="Python", variable=var2, onvalue=2, offvalue=0,
                             activeforeground='red', selectcolor='green',
                             command=mostra_selecionado)
btn_python.pack()
btn_java = tk.Checkbutton(janela, text="Java", variable=var3, onvalue=3, offvalue=0,
                           activeforeground='green', selectcolor='red',
                           command=mostra_selecionado)
btn_java.pack()

janela.mainloop()
```

Aqui, criamos uma função chamada `mostra_selecionado()` que é chamada quando o Checkbutton é selecionado ou desmarcado. Essa função simplesmente imprime o valor selecionado na tela. Passamos essa função como o valor da opção `command` aos Checkbutton, e quando o Checkbutton é selecionado ou desmarcado, a função é chamada.

Configuramos as opções `onvalue` e `offvalue` para os valores 1 e 0 respectivamente para o botão C#, 2 e 0 para o botão Python e 3 e 0 para o botão Java. Também configuramos as opções `activebackground` e `activeforeground` para definir as cores de fundo e foreground dos botões respectivamente. E a opção `selectcolor` para definir a cor do indicador de seleção.

Exercícios para Praticar

1. Crie uma janela com dois RadioButtons e exiba o valor selecionado quando um deles for escolhido.
2. Crie três RadioButtons com diferentes opções e exiba uma mensagem correspondente à opção selecionada.
3. Crie um RadioButton que desativa um campo de entrada quando selecionado e ativa quando não selecionado.
4. Crie um conjunto de RadioButtons que representam opções de cores e altere o fundo da janela para a cor selecionada.
5. Crie um grupo de RadioButtons que permite ao usuário selecionar seu gênero e exiba uma mensagem adequada com base na escolha.
6. Crie um conjunto de RadioButtons que representam tamanhos de fonte e altere o tamanho do texto em um rótulo quando uma opção for selecionada.
7. Crie um conjunto de RadioButtons que representam diferentes formatos de data e exiba a data atual formatada de acordo com a opção selecionada.
8. Crie um grupo de RadioButtons que permite ao usuário escolher uma opção de animal de estimação e exiba uma mensagem correspondente à opção selecionada.
9. Crie um conjunto de RadioButtons que representam diferentes unidades de medida e converta um valor inserido para a unidade selecionada.
10. Crie três RadioButtons com diferentes opções e exiba uma mensagem informando quantos RadioButtons estão selecionados.
11. Crie um grupo de RadioButtons que permite ao usuário escolher seu estado civil e exiba uma mensagem personalizada com base na escolha.
12. Crie um grupo de RadioButtons que permite ao usuário escolher um idioma e altere o texto em um rótulo para esse idioma.
13. Crie um conjunto de RadioButtons que representam diferentes opções de tema e altere o estilo da janela para o tema selecionado.
14. Crie um grupo de RadioButtons que permite ao usuário selecionar uma opção de classificação e ordene uma lista de nomes com base na escolha.
15. Crie um grupo de RadioButtons que permite ao usuário escolher uma opção de pagamento e exiba uma mensagem com detalhes adicionais com base na escolha.
16. Crie um conjunto de RadioButtons que representam diferentes estilos de cursor e altere o cursor quando uma opção for selecionada.
17. Crie um grupo de RadioButtons que permite ao usuário escolher uma opção de transporte e exiba uma mensagem informando o tempo estimado de viagem com base na escolha.
18. Crie uma janela com um Checkbutton e exiba uma mensagem quando ele for selecionado.
19. Crie três Checkbuttons com diferentes opções e exiba uma mensagem com base nas opções selecionadas.
20. Crie um Checkbutton que habilita ou desabilita um campo de entrada quando marcado ou desmarcado, respectivamente.
21. Crie um conjunto de Checkbuttons que representam opções de idioma e altere o texto em um rótulo para o idioma selecionado.
22. Crie um grupo de Checkbuttons que permite ao usuário selecionar múltiplas opções de cores e altere o fundo da janela para as cores selecionadas.
23. Crie um conjunto de Checkbuttons que representam diferentes categorias de produtos e exiba uma lista dos produtos selecionados.
24. Crie um grupo de Checkbuttons que permite ao usuário selecionar múltiplas opções de alimentos e exiba uma mensagem informando as opções selecionadas.
25. Crie um conjunto de Checkbuttons que representam diferentes estilos de fonte e altere o estilo do texto em um rótulo com base nas opções selecionadas.
26. Crie um grupo de Checkbuttons que permite ao usuário escolher múltiplas opções de frutas e exiba uma mensagem informando as opções selecionadas.
27. Crie um grupo de Checkbuttons que permite ao usuário escolher múltiplas opções de hobbies e exiba uma mensagem informando as opções selecionadas.
28. Crie um conjunto de Checkbuttons que representam diferentes opções de formatação de texto e aplique as formatações selecionadas a um campo de label.
29. Crie um grupo de Checkbuttons que permite ao usuário selecionar múltiplas opções de filmes e exiba uma mensagem informando as opções selecionadas.

30. Crie um conjunto de Checkbuttons que representam diferentes opções de filtros de pesquisa e aplique os filtros selecionados a uma lista de resultados.
31. Crie um grupo de Checkbuttons que permite ao usuário selecionar múltiplas opções de ingredientes e exiba uma mensagem informando as opções selecionadas.
32. Crie um conjunto de Checkbuttons que representam diferentes opções de personalização de interface e aplique as configurações selecionadas.
33. Crie um grupo de Checkbuttons que permite ao usuário escolher múltiplas opções de esportes e exiba uma mensagem informando as opções selecionadas.
34. Crie um conjunto de Checkbuttons que representam diferentes opções de permissão e controle o acesso a recursos com base nas opções selecionadas.
35. Crie um grupo de Checkbuttons que permite ao usuário selecionar múltiplas opções de temas de cores e altere a aparência da interface de acordo com as opções selecionadas.