

Feuille de Route : Apprendre Tkinter de Zéro

Étape 1 : Comprendre les bases de Tkinter

1.1. Installer Tkinter

- Tkinter est préinstallé avec Python. Assurez-vous que votre version de Python inclut Tkinter.

Si une fenêtre s'ouvre, Tkinter est installé.

1.2. Créer votre première fenêtre Tkinter

Apprenez à créer une fenêtre de base.

```
import tkinter as tk

# Créer une fenêtre principale
root = tk.Tk()
root.title("Ma Première Fenêtre Tkinter") # Titre de la fenêtre
root.geometry("400x300") # Dimensions de la fenêtre (largeur x hauteur)

# Lancer la boucle principale
root.mainloop()
```

Étape 2 : Ajouter des Widgets de base

2.1. Ajouter un label (étiquette)

Un `Label` est utilisé pour afficher du texte ou des images.

```
label = tk.Label(root, text="Bienvenue dans Tkinter !", font=("Arial", 16))
label.pack() # Pack pour afficher le widget
```

2.2. Ajouter un bouton

Apprenez à ajouter des boutons et à leur associer des actions.

```
def on_click():
    label.config(text="Bouton cliqué !")

button = tk.Button(root, text="Cliquez-moi", command=on_click)
button.pack()
```

2.3. Ajouter une entrée (Entry)

Permet de saisir du texte.

Code :

```
entry = tk.Entry(root)
entry.pack()

def afficher_texte():
    text = entry.get()
    label.config(text=f"Vous avez écrit : {text}")

btn_afficher = tk.Button(root, text="Afficher Texte", command=afficher_texte)
btn_afficher.pack()
```

Étape 3 : Disposition des widgets

3.1. Disposition avec `.pack()`

Dispose les widgets de manière verticale ou horizontale.

Code :

```
label1 = tk.Label(root, text="Label 1")
label2 = tk.Label(root, text="Label 2")
label3 = tk.Label(root, text="Label 3")

label1.pack()
label2.pack(side=tk.LEFT) # Aligné à gauche
label3.pack(side=tk.RIGHT) # Aligné à droite
```

3.2. Disposition avec `.grid()`

Dispose les widgets dans une grille.

Code :

```
for i in range(3):
    for j in range(3):
        btn = tk.Button(root, text=f"({i}, {j})")
        btn.grid(row=i, column=j) # Placement dans la grille
```

Étape 4 : Ajouter des widgets avancés

4.1. Checkbuttons (Cases à cocher)

```
var1 = tk.IntVar()
var2 = tk.IntVar()

check1 = tk.Checkbutton(root, text="Option 1", variable=var1)
check2 = tk.Checkbutton(root, text="Option 2", variable=var2)

check1.pack()
check2.pack()
```

4.2. Radiobuttons (Boutons radio)

Code :

```
var = tk.StringVar(value="Option 1")

radio1 = tk.Radiobutton(root, text="Option 1", variable=var, value="Option 1")
radio2 = tk.Radiobutton(root, text="Option 2", variable=var, value="Option 2")
radio1.pack()
radio2.pack()

def afficher_selection():
    print(f"Vous avez sélectionné : {var.get()}")

btn_selection = tk.Button(root, text="Afficher Sélection", command=afficher_selection)
btn_selection.pack()
```

4.3. Listbox (Liste déroulante)

Code :

```
listbox = tk.Listbox(root)

listbox.insert(1, "Python")

listbox.insert(2, "Tkinter")

listbox.insert(3, "Django")

listbox.pack()


def afficher_element():

    selection = listbox.get(listbox.curselection())

    label.config(text=f"Vous avez choisi : {selection}")


btn_list = tk.Button(root, text="Afficher Élément", command=afficher_element)

btn_list.pack()
```

Étape 5 : Ajouter des barres de menu

Apprenez à créer des menus dans votre application.

Code :

```
menu_bar = tk.Menu(root)


# Ajouter un menu déroulant

file_menu = tk.Menu(menu_bar, tearoff=0)

file_menu.add_command(label="Nouvelle")

file_menu.add_command(label="Ouvrir")

file_menu.add_separator()

file_menu.add_command(label="Quitter", command=root.quit)
```

```
menu_bar.add_cascade(label="Fichier", menu=file_menu)

# Ajouter un menu Aide
help_menu = tk.Menu(menu_bar, tearoff=0)
help_menu.add_command(label="À propos")
menu_bar.add_cascade(label="Aide", menu=help_menu)

root.config(menu=menu_bar)
```

Étape 6 : Personnaliser les fenêtres

Apprenez à créer des boîtes de dialogue ou des fenêtres secondaires.

6.1. Fenêtres secondaires (Toplevel)

Code :

```
def ouvrir_fenetre():
    top = tk.Toplevel(root)
    top.title("Fenêtre Secondaire")
    label = tk.Label(top, text="Ceci est une autre fenêtre")
    label.pack()

    btn_ouvrir = tk.Button(root, text="Ouvrir Fenêtre", command=ouvrir_fenetre)
    btn_ouvrir.pack()
```

6.2. Utiliser des boîtes de dialogue

Code :

```
from tkinter import messagebox
```

```
def afficher_message():
    messagebox.showinfo("Information", "Ceci est une boîte de dialogue")

btn_message = tk.Button(root, text="Afficher Message", command=afficher_message)
btn_message.pack()
```

Étape 7 : Ajouter des canvas pour des dessins graphiques

Code :

```
canvas = tk.Canvas(root, width=400, height=300, bg="white")
canvas.pack()

# Dessiner des formes
canvas.create_rectangle(50, 50, 200, 150, fill="blue")
canvas.create_oval(100, 100, 300, 200, fill="red")
canvas.create_line(0, 0, 400, 300, width=3)
```

Étape 8 : Réaliser un projet complet

Projet : Une calculatrice simple

Code :

```
def cliquer(bouton):
    current = entry.get()
    entry.delete(0, tk.END)
    entry.insert(0, current + bouton)

def calculer():
```

```

result = eval(entry.get())

entry.delete(0, tk.END)

entry.insert(0, str(result))


def effacer():

    entry.delete(0, tk.END)


entry = tk.Entry(root, width=20, font=("Arial", 16), justify='right')
entry.grid(row=0, column=0, columnspan=4)


buttons = [

    '7', '8', '9', '/',

    '4', '5', '6', '*',

    '1', '2', '3', '-',

    'C', '0', '=', '+'

]


row, col = 1, 0

for button in buttons:

    if button == "=":

        btn = tk.Button(root, text=button, width=5, height=2, command=calculer)

    elif button == "C":

        btn = tk.Button(root, text=button, width=5, height=2, command=effacer)

    else:

        btn = tk.Button(root, text=button, width=5, height=2, command=lambda b=button:
cliquer(b))

```

```
btn.grid(row=row, column=col)
```

```
col += 1
```

```
if col > 3:
```

```
    col = 0
```

```
    row += 1
```