

Python Tkinter Frame

Trong Python với thư viện Tkinter, Frame là một widget dùng để chứa và quản lý các widget khác nhau hoặc hình ảnh. Nó cũng được sử dụng để tạo các đường phân cách giữa các phần của giao diện người dùng.

Sử dụng Frame để quản lý widget và hình ảnh với màu nền, viền và sắp xếp layout

Tạo một Frame:

Để tạo một Frame, bạn có thể chỉ định chiều rộng và chiều cao của nó. Nếu bạn sử dụng layout pack, hãy đặt `expand=True` để nó lấp đầy tất cả không gian có sẵn.

```
import tkinter as tk

my_w = tk.Tk()
my_w.geometry("400x200")

frame_top = tk.Frame(my_w, width=380, height=180,
                      highlightbackground='red',
                      highlightthickness=5)
frame_top.pack(expand=True)

my_w.mainloop()
```

Thêm ảnh vào Frame:

Để thêm một hình ảnh vào Frame, bạn có thể sử dụng Label hoặc Button. Để sử dụng hình ảnh jpg, bạn cần sử dụng thư viện PIL.

```
import tkinter as tk

my_w = tk.Tk()
my_w.geometry("400x400")

frame_top = tk.Frame(my_w, width=390, height=390,
                      highlightbackground='red',
                      highlightthickness=3)
frame_top.grid(row=0, column=0, padx=5, pady=5, ipadx=5,
               ipady=5)

my_img = tk.PhotoImage(file="D:\\top2.png")
l2 = tk.Label(frame_top, image=my_img)
l2.grid(row=0, column=0, padx=5, pady=5)

my_w.mainloop()
```

Màu nền của Frame:

Bạn có thể thiết lập màu nền cho Frame bằng cách sử dụng option `background='yellow'` hoặc mã Hex. Nếu sử dụng layout pack, giữ option `expand=True` để lấp đầy tất cả khu vực có sẵn.

```
import tkinter as tk

my_w = tk.Tk()
my_w.geometry("400x400")

frame_top = tk.Frame(my_w, width=390, height=390,
                     background='yellow',
                     highlightbackground='red',
                     highlightthickness=3)
frame_top.grid(row=0, column=0, padx=5, pady=5, ipadx=5,
               ipady=5)

l2 = tk.Label(frame_top, text='yellow background with red
border')
l2.grid(row=0, column=0, padx=5, pady=5)

my_w.mainloop()
```

Hiểu layout với nhiều Frame:

Trong một ứng dụng phức tạp, bạn có thể cần sử dụng nhiều Frame để sắp xếp layout. Dưới đây là một ví dụ đơn giản về cách sử dụng nhiều Frame:

```
import tkinter as tk
from tkinter import *

my_w = tk.Tk()
my_w.geometry("400x350")

# Tạo Frame đầu tiên
frame_top = tk.Frame(my_w, width=390, height=150,
                     highlightbackground='red',
                     highlightthickness=3)
frame_top.grid(row=0, column=0, padx=5, pady=5, ipadx=5,
               ipady=5, columnspan=2)

# Thêm hình ảnh vào Frame đầu tiên
my_img = tk.PhotoImage(file="D:\\top2.png")
l2 = tk.Label(frame_top, image=my_img)
l2.grid(row=0, column=0, padx=5, pady=5)

# Tạo Frame thứ hai
frame_body = tk.Frame(my_w, width=380, height=150,
                     background='lightblue',
```

```

highlightbackground='lightgreen',
highlightthickness=3)
frame_body.grid(row=1, column=0, padx=20, pady=20, ipadx=5,
ipady=5, columnspan=2)

l3 = tk.Label(frame_body, text='Name', font=16)
l3.grid(row=0, column=0, padx=10, pady=10)
t1 = tk.Entry(frame_body, width=20, font=16)
t1.grid(row=0, column=1)
b1 = tk.Button(frame_body, text='Submit', font=20,
fg='blue')
b1.grid(row=1, column=1, padx=20, pady=10)

# Tạo nút để xóa Frame thứ ba
b2 = tk.Button(my_w, text='Delete', font=20,
fg='Red', command=lambda:
frame_bottom.grid_forget())
b2.grid(row=2, column=0, padx=10, pady=10)

# Tạo nút để khôi phục Frame thứ ba
b3 = tk.Button(my_w, text='Restore', font=20, fg='Green',
command=lambda: frame_bottom.grid(row=3,
column=0, columnspan=2))
b3.grid(row=2, column=1, pady=10)

# Tạo Frame thứ ba
frame_bottom = tk.Frame(my_w, width=380, height=150)
frame_bottom.grid(row=3, column=0, columnspan=2)

l4 = tk.Label(frame_bottom, text='I am at bottom frame',
bg='lightblue', font=18)
l4.grid(row=0, column=0, pady=10, sticky='w')

my_w.mainloop()

```

Cấu hình các tùy chọn:

Bạn có thể cấu hình các tùy chọn khác nhau của Frame bằng cách sử dụng phương thức `config()`. Dưới đây là một ví dụ về cách thay đổi màu nền của Frame khi nhấp vào một nút:

```

import tkinter as tk

my_w = tk.Tk()
my_w.geometry("600x400")

def my_open():
    frame_top.config(bg='lightgreen')

frame_top = tk.Frame(my_w, width=390, height=150,
bg='yellow',
highlightbackground='red',

```

```

highlightthickness=3)
frame_top.grid(row=0, column=0, padx=25, pady=25,
columnspan=2)

my_button = tk.Button(my_w, text="8syncdev", fg="blue",
                      cursor="hand2", font=18,
                      command=my_open)
my

_button.grid(row=1, column=0, padx=20, pady=20)

my_w.mainloop()

```

Phân cách:

Frame cũng có thể được sử dụng để tạo các đường phân cách giữa các phần của giao diện người dùng. Dưới đây là một ví dụ về cách sử dụng Frame để tạo một đường phân cách giữa các widget:

```

import tkinter as tk
from tkinter import *

my_w = tk.Tk()

b1 = tk.Button(my_w, text='b1', width=10)
b1.pack(side=TOP)

b2 = tk.Button(my_w, text='b2')
b2.pack(side=TOP)

# Tạo đường phân cách bằng Frame
my_line = tk.Frame(my_w, bg='green', bd=20, height=5,
width=20,
                    relief=RAISED, highlightcolor='yellow')
my_line.pack(side=TOP, fill=X)

b3 = tk.Button(my_w, text='b3')
b3.pack(side=BOTTOM)

l4 = tk.Label(my_w, text='l4 label')
l4.pack(side=BOTTOM)

my_w.mainloop()

```

Trên đây là một số cách sử dụng Frame trong Tkinter để quản lý và sắp xếp layout của giao diện người dùng. Bạn có thể tùy chỉnh các tùy chọn của Frame để đạt được thiết kế giao diện mong muốn.

