

Trong thư viện Tkinter của Python, bạn có thể sử dụng tên của màu trực tiếp thay vì mã Hex. Dưới đây là một số tên màu được hỗ trợ và tương ứng với mã Hex và giá trị RGB:

- **Tkinter Color Names:**

Tên Màu	HEX	RGB
aliceblue	#F0F8FF	RGB(240,248,255)
antiquewhite	#FAEBD7	RGB(250,235,215)
antiquewhite1	#FFEFDB	RGB(255,239,219)
antiquewhite2	#EEDFCC	RGB(238,223,204)
antiquewhite3	#CDC0B0	RGB(205,192,176)
antiquewhite4	#8B8378	RGB(139,131,120)
...	...	...

Bạn có thể sử dụng các tên màu này trực tiếp trong các phương thức và thuộc tính của các widget Tkinter. Ví dụ, bạn có thể đặt màu nền của một Button thành "yellow" hoặc "green" như sau:

```
import tkinter as tk

root = tk.Tk()
my_font = ('Helvetica', 12)

# Khởi tạo Button với màu nền yellow và màu chữ green
b1 = tk.Button(root, text='Hi Welcome', width=20,
bg='yellow', font=my_font, fg='green')
b1.pack()

root.mainloop()
```

Nếu bạn muốn sử dụng các màu từ một danh sách hoặc DataFrame, bạn có thể sử dụng tên màu từ DataFrame đó và cập nhật màu nền của widget tương ứng. Ví dụ:

```
import pandas as pd
import tkinter as tk

root = tk.Tk()

# Đọc DataFrame từ file Excel
df = pd.read_excel("D:\\my_data\\tk-colours.xlsx") # Thay
đổi đường dẫn đến tệp của bạn
```

```
# Chuyển cột 'Color Name' thành một list
my_list = df['Color Name'].values.tolist()

# Lặp qua danh sách và tạo các Button với màu nền tương ứng
for color_name in my_list:
    b = tk.Button(root, text=color_name, bg=color_name)
    b.pack()

root.mainloop()
```

Điều này sẽ tạo ra một loạt các Button với màu nền được chọn từ danh sách màu trong DataFrame của bạn.