

Python Memory Game

Add Some Colour!

Once you have finished following the **Matchmaker_python.pdf** tutorial at <https://github.com/IndyRockenRoll/MemoryGame>, why don't we add some colour and size to our card grid.

You can use your existing script and make the following changes:

Start by importing the **tkinter.font** package, as follows:

```
import random
import time
from tkinter import Tk, Button, DISABLED, messagebox
import tkinter.font as font #<<<<<<<<<< Add this line to change font size and add colour
```

Now in the **show_symbol()** function definition make the changes highlighted in yellow below:

```
def show_symbol(x,y):  
    global first  
    global previousX, previousY  
    global moves  
    global pairs  
  
    buttons[x,y]['text'] = button_symbols[x,y]  
    buttons[x,y].update_idletasks()  
  
    if first:  
        previousX = x  
        previousY = y  
        first = False  
        moves = moves + 1  
    elif previousX != x or previousY != y:  
        if buttons[previousX, previousY]['text'] != buttons[x,y]['text']:  
            time.sleep(0.5)  
            buttons[previousX, previousY]['text'] = ''  
            buttons[x,y]['text'] = ''  
        else:  
            buttons[previousX, previousY]['command'] = DISABLED  
            buttons[previousX, previousY]['bg'] = "#cc00cc" #<<<<<<<<<< Add this line to change font size and add colour  
            buttons[x,y]['command'] = DISABLED  
            buttons[x,y]['bg'] = "#cc00cc" #<<<<<<<<<< Add this line to change font size and add colour  
            pairs = pairs + 1  
            if pairs == len(buttons)/2:  
                my_var = messagebox.showinfo('Matching', 'Number of moves: ' + str(moves)) #<<<<<<<<<< Different from original tutorial  
                if my_var == 'ok': #<<<<<<<<<<<< Different from original tutorial  
                    root.destroy() #<<<<<<<<<<<< Different from original tutorial  
            first = True
```

Now let's define our font type and size:

```
root = Tk()
root.title('Matchmaker')
root.resizable(width=False, height=True)

myFont = font.Font(family="Helvetica",size=35) #<<<<<<<<<< Add this line to change font size and add colour
```

And finally, let's add some colour to our memory cards and add our font:

```
for x in range(6):  
    for y in range(4):  
        button = Button(command=lambda x=x, y=y: show_symbol(x,y), width=4, height=2, bg='#0052cc', fg='ffffff')  
        button["font"] = myFont #<<<<<<<<<<< Change this line to change font size and add colour  
        button.grid(column=x, row=y, padx=1, pady=1)  
        buttons[x,y] = button  
        button symbols[x,y] = symbols.pop()
```

That's it! Now when you start your script your memory cards should be bigger and more colourful..

Add some tweaks: if you like, play around with the colours “#cc00cc” (purple), “#0052cc” (blue), and “#ffffff” (white). These are hexadecimal representations of colours (google “Colour Picker” for more information), but you can replace these values with “red”, “orange”, “green”, if you don’t like HEX values.

Here is the **complete adjusted code**:

```
import random
import time
from tkinter import Tk, Button, DISABLED, messagebox
import tkinter.font as font #<<<<<<<<<<<<< Add this line to change font size and add colour

buttons = {}
first = True
previousX = 0
previousY = 0
moves = 0
pairs = 0

def show_symbol(x,y):
    global first
    global previousX, previousY
    global moves
    global pairs

    buttons[x,y]['text'] = button_symbols[x,y]
    buttons[x,y].update_idletasks()

    if first:
        previousX = x
        previousY = y
        first = False
        moves = moves + 1
    elif previousX != x or previousY != y:
        if buttons[previousX, previousY]['text'] != buttons[x,y]['text']:
            time.sleep(0.5)
            buttons[previousX, previousY]['text'] = ''
            buttons[x,y]['text'] = ''
        else:
            buttons[previousX, previousY]['command'] = DISABLED
            buttons[previousX, previousY]['bg'] = '#cc00cc' #<<<<<<<<<<<<< Add this line to change font size and add colour
            buttons[x,y]['command'] = DISABLED
            buttons[x,y]['bg'] = '#cc00cc' #<<<<<<<<<<<<< Add this line to change font size and add colour
            pairs = pairs + 1
            if pairs == len(buttons)/2:
                my_var = messagebox.showinfo('Matching', 'Number of moves: ' + str(moves)) #<<<<<<<<<<<<< Different from original tutorial
                if my_var == 'ok': #<<<<<<<<<<<<< Different from original tutorial
                    root.destroy() #<<<<<<<<<<<<< Different from original tutorial
            first = True

root = Tk()
root.title('Matchmaker')
root.resizable(width=False, height=True)

myFont = font.Font(family="Helvetica",size=35) #<<<<<<<<<<<<< Add this line to change font size and add colour

button_symbols = {}
symbols = ['\u2702', '\u2702', '\u2705', '\u2705', '\u2708', '\u2708',
           '\u2709', '\u2709', '\u270A', '\u270A', '\u270B', '\u270B',
           '\u270C', '\u270C', '\u270F', '\u270F', '\u2712', '\u2712',
           '\u2714', '\u2714', '\u2716', '\u2716', '\u2728', '\u2728']
random.shuffle(symbols)

for x in range(6):
    for y in range(4):
        button = Button(command=lambda x=x, y=y: show_symbol(x,y), width=4, height=2, bg='#0052cc', fg='#ffffff') #<<<<<<<<<<<<< Change this line to change font size and add colour
        button['font'] = myFont #<<<<<<<<<<<<< Change this line to change font size and add colour
        button.grid(column=x, row=y, padx=1, pady=1)
        buttons[x,y] = button
        button_symbols[x,y] = symbols.pop()

root.mainloop()
```