

给图片增加文字滚动效果

一、项目介绍

每个人的图片库里，多多少少都会存一些好玩、有趣的照片，该项目可以给这些照片添加一行有滚动效果的文字，制作成类似像广告牌一样有滚动效果的图文。

二、项目代码与结果展示

2.1 项目代码

```
1 import tkinter as tk # 界面模块
2 import pygame        # 调用pygame模块，用于加载音乐
3
4 # 可自定义的变量
5 tk_title = "5周年纪念" # 窗体标题
6 bg_photo = r'风变科技.gif' # 背景图片
7 bg_music = r'xxx.mp3' # 背景音乐，自行传入歌曲
8 font_size = int(30) # 字符的大小
9 font_color = "#bf9bfa" # 字符颜色
10 font_bg = 'snow' # 字符背景颜色
11 words = '2020年风变 5周年纪念日' # 滚动字符
12 # 创建GUI界面
13 def defineWindow():
14     global window, photo, cutdown_label
15     window = tk.Tk() # 创建一个新的空白GUI界面
16     window.title(tk_title) # 设置界面标题
17     photo = tk.PhotoImage(file=bg_photo) # 加载图片
18     cutdown_label = tk.Label(window, text=words, font=("Calibri", font_size), fg=font_color,
19                               image=photo, compound='top', background = 'snow') # 描述标签,字体的格式
20     cutdown_label.grid(row=0, rowspan=1, column=0, columnspan=1) # 把标签放置在GUI界面上
21
22 # 定义刷新函数
23 def Refresh():
24     global words
25     words = words[1:] + words[0] # 修改滚动字符
26     cutdown_label["text"] = words # 更新滚动字符
```

```

27     window.after(1000, Refresh)      # 1000毫秒刷新一次，再次调用Refresh函数
28
29 # 加载音乐
30 def loadMusic():
31     pygame.init()          # 初始化
32     pygame.mixer.music.load(bg_music)    # 加载音乐
33     pygame.mixer.music.play(-1, 10)      # 设置循环和开始位置
34
35 # 主函数
36 def main():
37     defineWindow()
38     loadMusic()
39     Refresh()
40     window.mainloop()
41
42 if __name__ == "__main__":
43     main()

```

2.2 结果展示

原图（风变科技.gif）：



执行结果：



三、项目解析

3.1 使用指南

```
1 # 可自定义的变量
2 tk_title = "5周年纪念"          # 窗体标题
3 bg_photo = r'风变科技.gif'      # 背景图，图片支持gif、jpg、png等格式
4 bg_music = r'盗将行.mp3'        # 背景音乐
5 font_size = int(30)             # 字体的大小
6 font_color = "#bf9bfa"          # 字体颜色
7 font_bg = 'snow'                # 字符背景颜色
8 words = '2020年风变 5周年纪念日' # 滚动字符
```

需要进行DIY的同学，可以修改可自定义的变量板块中的7个变量即可。

- **窗体标题**：显示在GUI界面的左上角，可根据自己的内容主题设置；
- **背景图片和背景音乐**：如果图片或音乐和python文件不在同一个文件夹中，需要使用绝对路径；
- **字体大小**：使用英文中的"磅"，1磅=1/72英寸；
- **字体颜色**：需要16进制的颜色代码，更多的颜色代码参考：<https://www.qtccolor.com/findColor.aspx>
<<https://www.qtccolor.com/findColor.aspx>>

- **字符背景颜色**：输入背景颜色对应的英文，具体对应关系参考：
http://www.science.smith.edu/dftwiki/index.php/Color_Charts_for_TKinter
<http://www.science.smith.edu/dftwiki/index.php/Color_Charts_for_TKinter>
- **滚动字符**：字符主要包括中文、英文、数字、空格和标点符号。

注意：字符的个数主要受背景图片的宽度的影响。背景图片的宽度越大，能够容纳的字数越多。

中文字符的计算公式是：字数=图片宽度/（图片DPI * 字体磅数*1/72），其中图片的DPI默认是96dpi，公式可简化为：**字数=3/4 * 图片宽度/字体磅数**。而英文或数字或英文标点，大约是中文字符的2倍，空格大约是4倍。

在该项目中，图片的宽度是800px，字体磅数是30，故最多可以容纳20个中文或中文标点，或约40个英文或数字或英文标点，或约80个空格。

3.2 其他注意点

1. 定义defineWindow()函数时，需要把window, photo, cutdown_label这三个变量都声明，否则会报错；
2. 定义Refresh()函数时，需要把words声明为全局变量，否则会因和全局变量words冲突而报错；
3. 定义main()函数时，注意mainloop()函数会使得GUI界面进入无限循环（可以通过关掉界面终止循环），其后的代码不会再执行，所以mainloop()函数的调用应该放在代码的最后。