

# python 大作业实验报告

舒文炫

2021 年 7 月 18 日

# 目录

<b>1</b>	<b>项目介绍</b>	<b>2</b>
1.1	项目背景与意义 . . . . .	2
1.2	项目成员 . . . . .	2
<b>2</b>	<b>项目实现</b>	<b>3</b>
<b>3</b>	<b>结果</b>	<b>7</b>
<b>4</b>	<b>项目总结</b>	<b>9</b>

# Chapter 1

## 项目介绍

### 1.1 项目背景与意义

本人是一位原神玩家，原神可以说是在我心目中当代二次元开放世界手游的顶峰之作，游戏里面的角色每个都让人感觉是立体的，是有血有肉的。其中刻晴和派蒙是我最喜欢的两位，派蒙是这个游戏里面的向导，十分可爱，刻晴是玩家可以操控的角色，手感很好，特别可爱。这次大作业就借着这个机会，整个活。我尝试用 python 的库开发了这一款名叫刻晴大战派蒙的游戏，算是体现一下对原神这款游戏的热爱。也希望有越来越多的人注意到国产游戏正在不断崛起，我相信过不了多久，我国的游戏整体水平也不会逊色与国外的 3A 大作。

### 1.2 项目成员

姓名：舒文炫 (组长)

学号：PB18000029

只有一个人 www

## Chapter 2

# 项目实施

这个大作业算是我对 lab5 的一点拓展，在 lab5 中我只是初步学习了 pygame 游戏库的调用，简单实现了角色的控制，在这里，我更加深入的了解了这个库，并尝试整合进去其他元素，使得这款游戏的完整性和可玩性得到提升。基础的部分写在了 lab5 的报告里面，这里不再赘述，我会把 lab5 报告也一起放在这里方便查阅。这里我会说明我额外实现的内容。

一，给这款游戏加入了背景音乐。背景音乐是一款游戏的灵魂，恰到好处的背景音乐，可以和游戏的剧情相互促进，使得玩家游玩体验大幅提升，这里我没有太多编曲的知识，使用的是陈致逸先生给原神所做名为无数的逆境的纯音乐，音乐文件放在 music 文件夹中，我选用这首，是因为它本身体现出了一种神秘的氛围，仿佛置身一段艰难的挑战，但是无尽的黑暗之中，又能隐隐约约体现出一丝希望，我设计的这款游戏，玩法主要是打飞机，并且我加大了难度，感觉和这首音乐比较相配。

```
12 pygame.mixer.init() ##bgm
13 game_settings=Settings()
14 pygame.mixer.music.load("music/bgm.ogg")
15 pygame.mixer.music.set_volume(0.2)
```

```
g, render_events(game_settings,screen,stats,play_sound,keqing,pimeng,bullets)
if pygame.mixer.music.get_busy() == False:
    pygame.mixer.music.play()
```

这就是我添加的音乐播放模块，load 方法导入音乐，然后通过 get\_busy 方法检测是否有音乐在播放，没有就播放所 load 的背景音乐，这样可以达到一直循环播放的效果。

二，我实现了游戏的战斗模块，这里实现的战斗逻辑很简单，就是角色可以在这个屏幕上自由移动，射击，射中派蒙，派蒙就死了，如果派蒙靠近了角色，角色也死了。角色具有一定量的生命值，死的次数超过了这个限制，游戏就重新开始。下面贴上代码

```
def keqing_hit(game_settings,stats,screen,keqing,pimeng,bullets): ##keqing is hit, then clear the screen and remake.
    if stats.keqing_left>0:
        stats.keqing_left-=1
        pimeng.empty()
        bullets.empty()
        create_fleet(game_settings,screen,pimeng)
        keqing.center_keqing()
        sleep(0.5)
    else:
        stats.game_active=False
```

这里是刻晴受击模块，刻晴受击，死亡，游戏界面重置，刻晴的生命值-1。

```

103
104 def check_pimeng_bottom(game_settings,stats,screen,keqing,pimeng,bullets):
105     screen_rect=screen.get_rect()
106     for pimengs in pimeng.sprites():
107         if pimengs.rect.bottom>screen_rect.bottom:
108             keqing_hit(game_settings,stats,screen,keqing,pimeng,bullets)
109             break
110
111
112 def update_pm(game_settings,stats,screen,keqing,pimeng,bullets): ##update pimeng
113     check_fleet_edges(game_settings,pimeng)
114     pimeng.update()
115     if pygame.sprite.spritecollideany(keqing,pimeng):
116         keqing_hit(game_settings,stats,screen,keqing,pimeng,bullets)
117     check_pimeng_bottom(game_settings,stats,screen,keqing,pimeng,bullets)
118

```

这里是第一段是判断派蒙是否到屏幕底部，派蒙如果触碰到了屏幕底部，会导致刻晴生命值-1，相当于大本营被派蒙端了。这里 for 循环保证只要有其中一只派蒙碰到，就算失败，就会调用刻晴受击模块。后面更新派蒙，通过 `spritecollideany` 方法，这个方法可以循环查找该派蒙组里面是否有任意一只派蒙与刻晴碰撞，如果碰到了，会调用刻晴受击模块

当然我的刻晴不能一直让她受击，她是可以通过子弹的方式反击邪恶的派蒙大军的，下面是射杀派蒙的代码。

```

def update_bullets(game_settings,screen,bullets,pimeng): ##remove the redundant bullets and let you shoot pimeng
    bullets.update()
    for bullet in bullets.copy():
        if bullet.rect.bottom<=0:
            bullets.remove(bullet)
    collisions=pygame.sprite.groupcollide(bullets,pimeng,True,True)
    if len(pimeng)==0: ##regenerate pimeng when there is no pimeng
        bullets.empty()
        create_fleet(game_settings,screen,pimeng)

```

这一块，我直接嵌入到了更新子弹函数里面，因为只有在子弹碰到派蒙时，派蒙才会被击杀。这里调用的 `groupcollide` 方法，与上面的 `spritecollideany` 方法类似，不过这里可以对两个组操作，判断这两个组中是否有任意元素发生碰撞，如果有，就删除这些碰撞到一起的元素。但是派蒙大军无穷无尽，达到一个又会有千千万万个派蒙站起来，所有后面，我添加了一个判断，如果当前界面的派蒙全被杀死，那么会再生成一行派蒙，使得这个游戏具有一定的持续性。

```

def cente (variable) centerx: Any
    self.centerx=self.screen_rect.centerx
    self.bottom=self.screen_rect.bottom

```

```

1 class Settings():
2     def __init__(self):
3         self.width=1200 ##screen width
4         self.height=800 ##screen height
5         self.bg_color=(200,150,120) ##background color RGB mode
6         self.kq_speed_factor=1 ##the speed of the character
7         self.bullet_speed_factor=0.8
8         self.bullet_width=2
9         self.bullet_height=10
10        self.bullet_color=(10,10,10)
11        self.bullet_allowed=1
12        self.pm_speed_factor=0.8
13        self.fleet_drop_speed=30
14        self.fleet_direction=1 ##1 means right -1 menas left
15        self.keqing_limit=3 ##you have three chances
16

```

上面这张图是 `keqing` 类的 `center` 方法主要是用在刻晴死亡时，对刻晴的位置进行重置，一开始并没有想到这一点，死亡时刻晴会原地复活，这样会导致一些问题，就是如果刻晴死亡的地方刚好在派蒙刷出的地方，这样刻晴会由于碰到派蒙反复去世，去世后又复活，然后又碰到派蒙，这样游戏就卡死了。这样一个方法，使得复活刻晴时，刻晴永远在屏幕下方，居中位置，这样设计也比较合理，毕竟下方就是刻晴的大本营。

下面这张图，是我对游戏初始化参数进行了微调，修改了子弹速度，以及同时所允许持有的最大子弹数，我将其设置为 1，即在一颗子弹消失之前，你没有办法发射下一发子弹，毕竟每次只会生成 4 个派蒙，4 发全中，则能顺利击杀，不能则会有一点险，使得游戏的难度也有了提升，配合背景音乐，可以感觉到比较刺激。

三，实现了更好的人机交互，lab5 里面运行 start.py 之后游戏就直接开始了，这其实很不符合大多数人的游戏习惯，很多游戏都具有开始界面，点击开始游戏这个时候，游戏才会正式开始。这里 pygame 它本身并没有设置按钮这个库函数可以直接调用，考虑到 pygame 里面所有元素中最基本的是矩形，按钮本质上就是一个矩形，我就尝试构建一个按钮类。

```
import pygame.font
class Button(): ##the begin button
    def __init__(self,game_settings,screen,msg):
        self.screen=screen
        self.screen_rect=screen.get_rect()
        self.width,self.height=200,50
        self.button_color=(21,34,56)
        self.text_color=(255,255,255)
        self.font=pygame.font.SysFont(None,48)
        self.rect=pygame.Rect(0,0,self.width,self.height)
        self.rect.center=self.screen_rect.center
        self.prep_msg(msg)

    def prep_msg(self,msg):
        self.msg_image=self.font.render(msg,True,self.text_color,self.button_color)
        self.msg_image_rect=self.msg_image.get_rect()
        self.msg_image_rect.center=self.rect.center

    def draw_button(self):
        self.screen.fill(self.button_color,self.rect)
        self.screen.blit(self.msg_image,self.msg_image_rect)
```

这里的代码就是按钮类，传入 screen 参数，可以使得按钮在屏幕中间出现，msg 参数就是按钮所显示的字，还有一些按钮长宽，颜色，字体这样的设计，prep\_msg 方法可以进行文字的输出生，然后 draw\_button 方法将这个按钮画在屏幕上这个按钮就完成了，不过目前它只是一个按钮，并不能进行互动，实现互动靠的是下面的代码。

```
def check_events(game_settings,screen,stats,play_button,keqing,pimeng,bullets):
    for event in pygame.event.get():##get the event such as mice and keybox input
        if event.type==pygame.QUIT:
            sys.exit()
        elif event.type==pygame.MOUSEBUTTONDOWN:
            mouse_x,mouse_y=pygame.mouse.get_pos()
            check_play_button(game_settings,screen,stats,play_button,keqing,pimeng,bullets,mouse_x,mouse_y)

        elif event.type==pygame.KEYDOWN: ##to move continuously of you press the button
            check_keydown(event,game_settings,screen,keqing,bullets)

        elif event.type==pygame.KEYUP:
            check_keyup(event,keqing)
```

这里检查事件不仅仅可以检测键盘的输入，还可以检测鼠标的点击，MOUSEBUTTONDOWN 参数表示的是鼠标的点击，点击的时候会返回一个鼠标点击位置，将这个位置传给 check\_play\_button 函数，这个函数通过判断鼠标点击位置是否在 play 按钮里面，如果在就开始游戏，这里游戏开始使用 game\_active 参数，这个参数初始化为 False，表示一开始游戏没有启动，停留在开始界面，点击 play 按钮后，这个按钮消失，游戏开始。

```
1 class GameStats():
2     def __init__(self,game_settings):
3         self.game_settings=game_settings
4         self.reset_stats()
5         self.game_active=False ##the game is over when it is False
6     def reset_stats(self):
7         self.keqing_left=self.game_settings.keqing_limit
```

这里就是我设置的游戏状态类，用来记录当前游戏是启动了还是在休眠状态，里面的 reset 方法，就是用来重设刻晴的生命，当刻晴死亡导致游戏结束时，画面会暂停，play 按钮会显示出来，这是会使用 reset 方法，赋予刻晴初始生命值，一开始没有注意这一点，导致重新开始时刻晴的生命为 0，无法重新开始游戏。

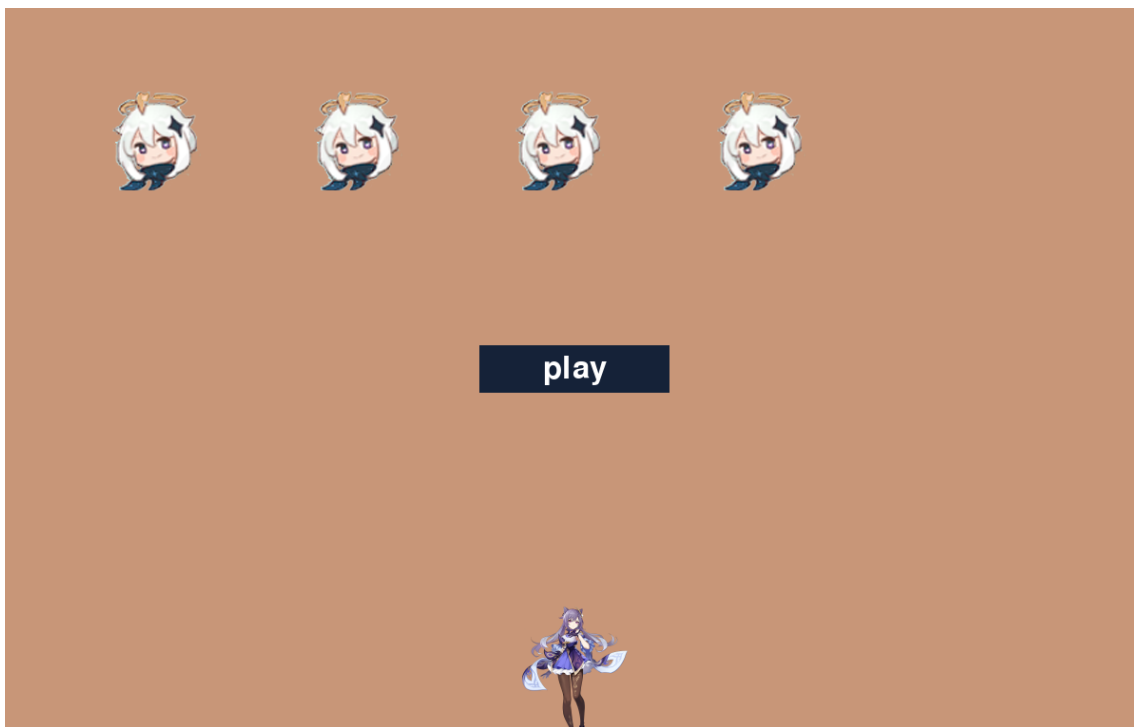
```
34 def check_play_button(game_settings,screen,stats,play_button,keqing,pimeng,bullets,mouse_x,mouse_y):
35     if play_button.rect.collidepoint(mouse_x,mouse_y) and not stats.game_active: ##press play key
36         stats.reset_stats()
37         stats.game_active=True
38         pimeng.empty()
39         bullets.empty() (parameter) pimeng: Any
40         create_fleet(game_settings,screen,pimeng)
41         keqing.center_keqing()
42
```

这个就是具体的 check\_play\_button 函数，鼠标位置在按钮里面的话，用上面提到的 reset 方法赋予刻晴生命，然后游戏状态变为启动，初始化派蒙，刻晴位置。

## Chapter 3

# 结果

此时运行 `start.py` 函数可以看到如下图形化窗口

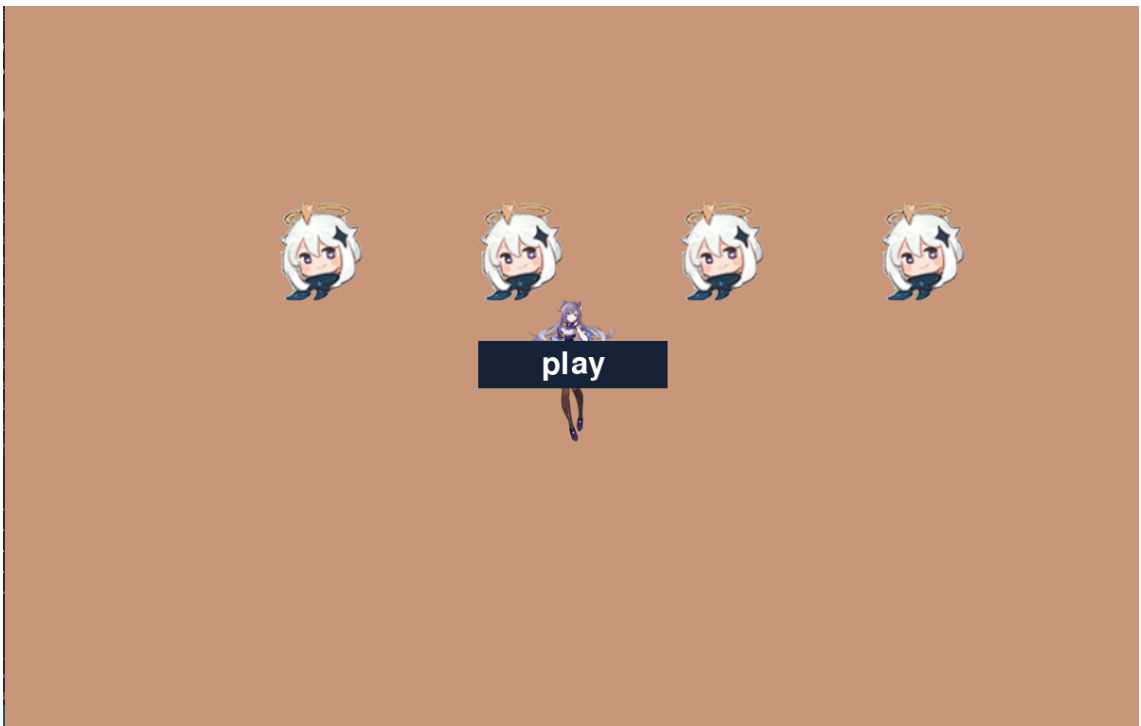


用刻晴进行射杀





刻晴死亡，游戏重来



按 q 键可以退出游戏

## Chapter 4

# 项目总结

以上便是我本次大作业的所有内容，因为对一款游戏的热爱，我尝试整了这个活，也算是初步圆了我以前想做游戏的梦想，不过有一说一，游戏的制作确实是一件耗时耗力的事，要想做到尽善尽美的程度，需要大量的人力物力财力，这里我也对所有独立游戏制作者表示出深深的敬佩。限于时间，精力，这款游戏我初步规划的很多功能不能一一实现，也存在一定的遗憾，不过在这个过程中，我切实体会到了 python 的强大，实用。