Python程序设计语言

课程设计报告

**专 业数字媒体技术（中职升本）**

**年 级 2018级**

**班 级 中职**

**学 号 18660021**

**姓 名 周昊**

1. **设计题目**

利用Python技术，完成作品设计，我设计的是飞机大战

1. **题目要求**

1. 音效

2. 设置关卡

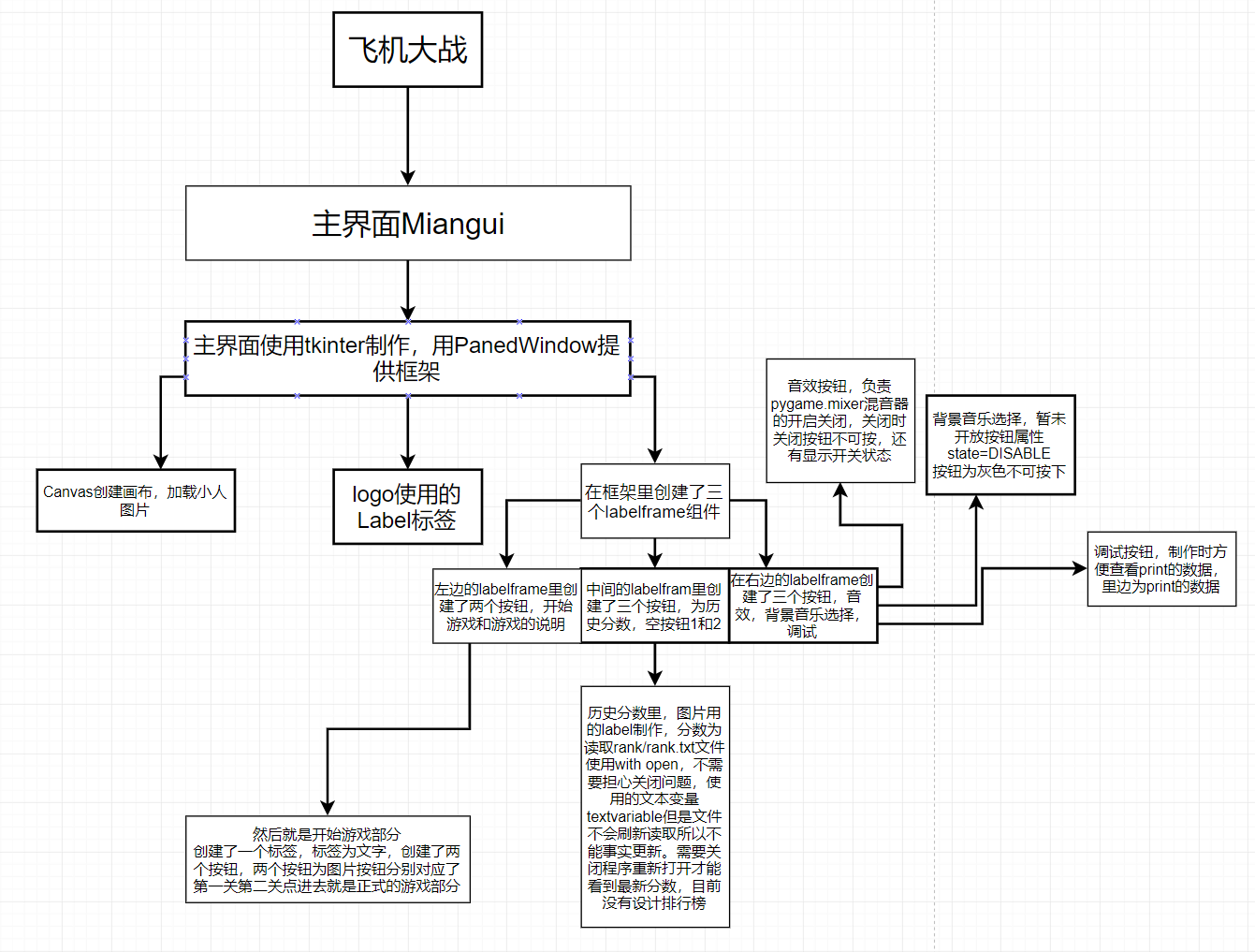
3. 界面美观

4.项目无明显漏洞

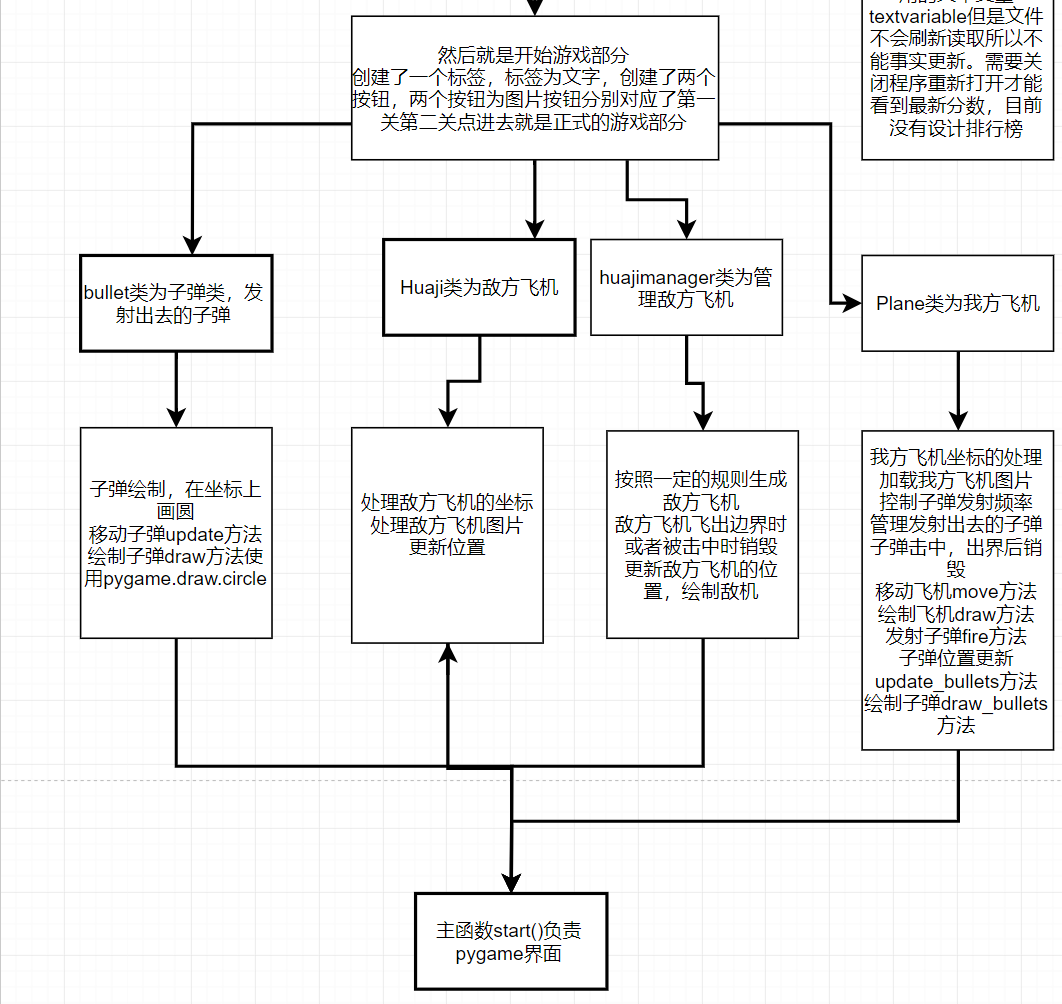
5.帮助说明

1. **程序结构**

**主界面流程图**



游戏核心流程



1. **代码设计**
2. 运用了面向对象和模块化的设计方法
3. 界面采用tkinter制作，使用了框架设计界面，button和label父集为labelframe，labelframe父集为panedwindow框架，框架父集为Tk界面。设计的界面。

音频使用pygame.mixer.init是否挂载，查询传回的值，0为关闭1为开启，再把这两个传入label里，来显示开启和关闭，然后传回关闭按钮，若混音器已经关闭那么，传给按钮state属性为DISABLE为灰色。不可以在点击。

按钮开启音效，按下按钮播放音效。

按钮关闭音效，因为按下按钮要执行播放，和卸载混音器，但是程序执行过快所以在播放后引入time.sleep(1)然后继续执行混音器的卸载

Pygame.mixer.quit()

为了方便显示print 的值，所以我加入了个DEBUG按钮来显示我print的值。点击debug弹出子顶级窗口Toplevel，如果使用tk的话，pygame的图片无法加载，经查询是这个问题，只能有一个tk。

然后就是历史分数这一选项。

点进去会显示分数，这个分数存储在rank文件夹下的rank.txt

游戏说明，为游戏的说明帮助

游戏开始，点击后提供两个关卡

一个简单关卡1提供三个生命值，敌方飞机速度慢，我方子弹频率更高

一个困难关卡2提供三个生命值，敌方飞机巨大速度快，我方子弹频率更低，增加了难度。

然后就是游戏的核心部分。

游戏帧率，窗口大小，pygame的初始化

创建pygame的主要窗口

控制刷新频率对象pygame.time.clock，防止这个程序运行过快，让程序固定速度运行

Clock.tick控制游戏的fps，fps越高游戏越流畅

然后添加键盘监听事件上下左右和wasd还有esc

Esc用于暂停，由于不可名状的原因，不关闭tk主程序窗口的话无法继续游戏所以暂不操作

然后就是玩家飞机的实现，

创建一个父控件（绘制飞机）

确定飞机的坐标

确定飞机的图像，加载文件

要实现一些方法来移动飞机到指定坐标

绘制玩家的飞机

在循环中每次绘制界面清空背景，不然就是一连串的飞机

键盘监听事件使用判断按下松开来连续执行移动，否则移动飞机需要一下一下的点击键盘。

还要防止飞机飞出边界

制作子弹类实现子弹的基本功能

Pygame.draw.circle来绘制子弹

玩家控制飞机，需要发射子弹

子弹发射频率可控制

还有失效子弹清除，防止过多卡顿

敌方飞机的实现

父控件，敌方飞机坐标

敌方飞机图片加载

更新位置

绘制位置

敌方飞机的管理

按规则生成敌方飞机

被打烂的飞机清除，超界的飞机清理掉

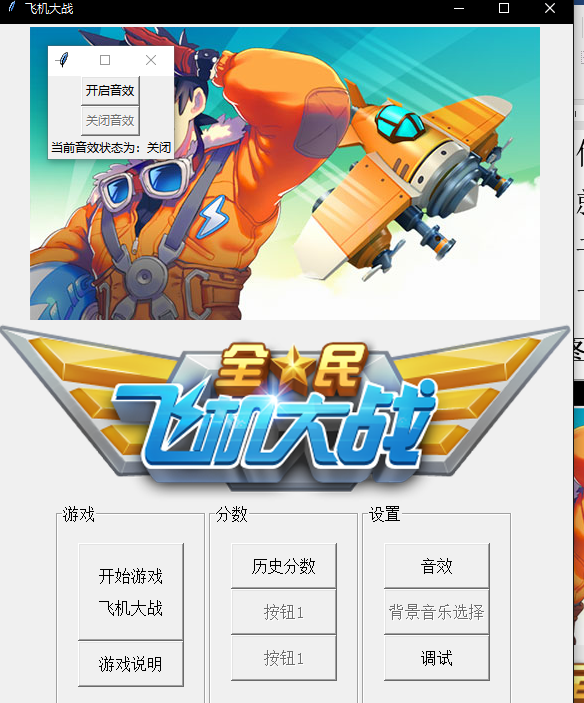
更新位置，绘制敌方飞机

最后就是碰撞的检测

自己子弹和敌方飞机的碰撞

自己飞机和敌方飞机的碰撞

1. **界面截图**



音效的关闭





历史分数和关卡选择和游戏战斗界面和分数，困难关卡和简单关卡和成绩写入和游戏的暂停界面

1. **收获总结**

**主要学会了界面的设计，学习了Tkinter的操作，按钮进度条滚动条文字标签等等，对于pygame有一定的学习。了解pygame流程**

**对于程序设计的思路有一定的了解。**

1. **关键代码说明**

**界面代码**

*# -\*- Coding: utf-8 -\*-  
# @Time : 2020/11/28 14:59  
# @Author : 周昊  
# @FileName: miangui.py  
# @Software: PyCharm  
# @Github : https://github.com/r4x2t7/jw  
# @Exercises:*

*from* tkinter *import* \*  
*import* sys  
*from* tkinter.messagebox *import* \*  
*import* time  
*import* pygame  
*from* GAME.app *import* start  
*from* GAME2.app *import* start2  
*import* os  
pygame.init()  
*for* event *in* pygame.event.get():  
 *if* event.type == pygame.QUIT:  
 *# 判断当前事件是否为点击右上角退出键* pygame.quit()  
 sys.exit()  
root = Tk()  
root.title(**"飞机大战"**)  
cv=Canvas(root,width=512,height=294)  
img10=PhotoImage(file=**"images/1.png"**)  
cv.create\_image(260,150,image=img10)  
cv.pack()  
*# 创建PanedWindow对象*logo = PhotoImage(file=**"images/LOGO.png"**)  
logo1=Label(root,image=logo).pack()  
pw = PanedWindow(orient=HORIZONTAL)  
*# 创建labelFrame当做子对象  
# 左边包含的  
def* game11():  
 game11=Toplevel()  
 game11.title(**"关卡选择"**)  
 background1 =PhotoImage(file=**"images/diyiguan.png"**)  
 background2 =PhotoImage(file=**"images/dierguan.png"**)  
 guankalabel=Label(game11,text=**"难度选择"**,font=300,pady=30).pack()  
 diyiguanbutton=Button(game11,image=background2,command=start).pack()  
 dierguanbutton=Button(game11,image=background1,command=start2).pack()  
 game11.mainloop()  
*def* gameinfo():  
 gameinfo=Toplevel()  
 gameinfo.title(**"游戏说明"**)  
 background1 = PhotoImage(file=**"images/Icon-114.png"**)  
 ranklabelimage=Label(gameinfo,image=background1).pack()  
 infolabel=Label(gameinfo,text=**"1.按键操作：wasd移动上下左右也可以移动**\n**2.如何开始游戏加载完毕点击开始游戏即可开始游戏。**\n**3.游戏目标合理操作，消灭所有的敌人吧!"**,font=150)  
 infolabel.pack()  
 gameinfo.mainloop()  
leftframe = LabelFrame(pw, text=**"游戏"**,font=66,padx=20,pady=20)  
leftButton1 = Button(leftframe, text=**"开始游戏**\n\n**飞机大战"**,font=66,width=12, pady=20,command=game11 ).pack()  
leftButton2 = Button(leftframe, text=**"游戏说明"**,font=66,width=12, pady=10,command=gameinfo).pack()  
*# leftButton3 = Button(leftframe, text="按钮3",font=66,width=12, pady=10).pack()*leftframe.pack()  
pw.add(leftframe)  
*# 中间布局和功能  
# 分数数据读取，写入  
######################################################################################  
def* rankmode():*#rank.txt文件读取，使用with，with执行完灰关闭文件，减少占用  
 global* list1  
 *with open*(file=**'rank/rank.txt'**,mode=**'r'**)*as* f:  
 list1=f.read()  
rankmode()  
*def* rank():*#rank.txt数据返回到list1* rankmode()  
 *return* (list1)  
*def* rankmodeui():*#分数的界面* rank=Toplevel()  
 rank.title(**"历史分数"**)  
 background1 = PhotoImage(file=**"images/ui\_new\_word\_png (2).png"**)  
 val=StringVar()  
 val.set(list1)  
 ranklabelimage=Label(rank,image=background1).pack()  
 ranklabel=Label(rank,textvariable=val,font=150).pack()  
 rank.mainloop()  
*######################################################################################*midframe = LabelFrame(pw, text=**"分数"**,font=66,padx=20,pady=20)  
midButton1 = Button(midframe, text=**"历史分数"**,font=66, width=12, pady=10,command=rankmodeui).pack()  
midButton2 = Button(midframe, text=**"按钮1"**, font=66,width=12, pady=10,state=DISABLED).pack()  
midButton3 = Button(midframe, text=**"按钮1"**,font=66, width=12, pady=10,state=DISABLED).pack()  
midframe.pack()  
pw.add(midframe)  
*# 右边包含  
#声音的处理  
######################################################################################  
def* settingmusicon():*#设置中的声音开启* loadmusic()  
 waveat()  
*def* settingmusicoff():*#设置中的声音关闭  
 if* judgemusicin == 1:  
 wavpause()  
 time.sleep(1)  
 pygame.mixer.quit()  
 *else*:  
 pygame.mixer.quit()  
*def* loadmusic():*#挂载混音器* pygame.mixer.pre\_init(44100, -16, 2, 512)  
 pygame.mixer.init()  
*def* unloadmusic():*#卸载混音器* pygame.mixer.quit()  
*def* waveat():*#声音* waveat = pygame.mixer.Sound(**"audio/eat.wav"**)  
 waveat.play()  
*def* wavpause():*#声音* wavpasue = pygame.mixer.Sound(**"audio/zanting.wav"**)  
 wavpasue.play()  
*def* judgemusic():*#混音器的开启关闭判断及返回的文本  
 global* judgemusicin,info*# 全局变量判断音频是否关闭* judgemusicin = 1  
 *if* pygame.mixer.get\_init() == *None*:  
 judgemusicin = 0  
 info=**"当前音效状态为：关闭"** *else*:  
 judgemusicin = 1  
 info=**"当前音效状态为：开启"**loadmusic()  
*def* DEBUG():*# 程序调试 ，类似print，因为写了面以后看不到print的反馈* DEBUG=Toplevel()  
 DEBUG.title(**"调试"**)  
 DEBUG.geometry(**"300x210"**)  
 DEBUGPRINT = Label(DEBUG, text=info)  
 DEBUGPRINT.pack()  
 DEBUG.mainloop()  
*######################################################################################  
def* music():*#音效设置的界面* music=Toplevel()  
 music.title(**"音效设置"**)  
 judgemusic()  
 musiclabel = Label(music,text=info)  
 musicbuttonon=Button(music,text=**"开启音效"**,command=settingmusicon)  
 musicinfo = Button(music,text=**"关闭音效"**,command=settingmusicoff)  
 *if* judgemusicin == 0:  
 musicinfo.config(state=DISABLED)  
 *else*:  
 musicinfo.config(state=NORMAL)  
 musicbuttonon.pack()  
 musicinfo.pack()  
 musiclabel.pack()  
 music.mainloop()  
rightframe = LabelFrame(pw, text=**"设置"**,font=66,padx=20,pady=20)  
rightButton1 = Button(rightframe, text=**"音效"**,font=66, width=12, pady=10,command=music).pack()  
rightButton2 = Button(rightframe, text=**"背景音乐选择"**,font=66,width=12, pady=10,state=DISABLED).pack()  
rightButton3 = Button(rightframe, text=**"调试"**,font=66, width=12, pady=10,command=DEBUG).pack()  
rightframe.pack()  
pw.add(rightframe)  
pw.pack()  
root.mainloop()

**游戏核心代码**

*# -\*- Coding: utf-8 -\*-  
# @Time : 2020/12/10 14:59  
# @Author : 周昊  
# @FileName: app.py  
# @Software: PyCharm  
# @Github : https://github.com/r4x2t7/jw  
# @Exercises:  
import* os  
*import* random  
*import* math  
*import* pygame  
*import* sys  
*import* time  
*from* tkinter *import* \*  
*from* tkinter *import* messagebox  
PLANEIMG = **"images/wsparticle\_test\_001.png"**BULLET = **"images/img\_bullet.png"**PLANESIZE = 90 *# 飞机对象直径（近似圆形）*SMALLSIZE = 100  
count = 0  
*def* start():  
 FPS=60 *# 游戏帧率* WINWIDTH = 512 *# 窗口宽度* WINHEIGHT = 768 *# 窗口高度* MOVESTEP=5 *# 移动速度* pygame.init() *# pygame初始化，必须有，且必须在开头* pygame.mixer.init()  
 pygame.mixer.music.load(**"audio/背景音乐.mp3"**)  
 pygame.mixer.music.play(-1)  
 sound = pygame.mixer.Sound(**"audio/爆炸.ogg"**)  
 *# 创建主窗体* clock=pygame.time.Clock() *# 用于控制循环刷新频率的对象* win=pygame.display.set\_mode((WINWIDTH,WINHEIGHT))  
 background = pygame.image.load(**"images/img\_bg\_level\_1.jpg"**)  
  
 plane=Plane(win,200,600)  
 hm=HuajiManager(win)  
 mx,my=0,0 *# 记录移动方向  
 def* paused():  
 pausedmenu= Toplevel()  
 *def* destory():  
 pausedmenu.destroy()  
 imgjixu=PhotoImage(file=**"images/jixuyouxi.png"**)  
 labeltext=Label(pausedmenu,text=**"游戏暂停"**)  
 labeltext.pack()  
 pausedmenu.overrideredirect(*True*)  
 width,height=300,100  
 screenwidth=pausedmenu.winfo\_screenwidth()  
 screenheight=pausedmenu.winfo\_screenheight()  
 alighstr=**'%dx%d+%d+%d'**%(width,height,(screenwidth-width)/2,(screenheight/2))  
 pausedmenu.geometry(alighstr)  
 buttonpause=Button(pausedmenu,image=imgjixu,command=destory)  
 buttonpause.pack()  
 pausedmenu.title(**"暂停/继续"**)  
 pausedmenu.resizable(width=*False*,height=*False*)  
 pausedmenu.mainloop()  
 *while True*:  
 win.blit(background,(0,0))  
 *# plane.check\_crash(hm.score1)  
 # score=hm.score1\*10  
 # score=bullet.check\_hit(count)\*10* scorefont=pygame.font.SysFont(**"华文彩云"**,30)  
 text\_surface=scorefont.render(**u"分数：%s"**%*str*(count),*True*,(255,255,255))  
 win.blit(text\_surface,(10,20))  
  
 *# 获取所有事件  
 for* event *in* pygame.event.get():  
 *if* event.type == pygame.QUIT:  
 *# 判断当前事件是否为点击右上角退出键* pygame.quit()  
  
 *if* event.type == pygame.KEYDOWN:  
 *if* event.key==pygame.K\_LEFT *or* event.key == *ord*(**'a'**):  
 mx=-1  
 *if* event.key==pygame.K\_RIGHT *or* event.key == *ord*(**'d'**):  
 mx=1  
 *if* event.key==pygame.K\_UP *or* event.key == *ord*(**'w'**):  
 my=-1  
 *if* event.key==pygame.K\_DOWN *or* event.key == *ord*(**'s'**):  
 my=1  
 *if* event.key == pygame.K\_ESCAPE:  
 paused()  
  
 *if* event.type == pygame.KEYUP:  
 *if* event.key==pygame.K\_LEFT *or* event.key == *ord*(**'a'**):  
 *if* mx==-1:  
 mx=0  
 *if* event.key==pygame.K\_RIGHT *or* event.key == *ord*(**'d'**):  
 *if* mx==1:  
 mx=0  
 *if* event.key==pygame.K\_UP *or* event.key == *ord*(**'w'**):  
 *if* my==-1:  
 my=0  
 *if* event.key==pygame.K\_DOWN *or* event.key == *ord*(**'s'**):  
 *if* my==1:  
 my=0  
 *# if event.key == pygame.K\_ESC or event.key == ord('^['):* plane.check\_all\_hit(hm.huajilist)  
 plane.check\_crash(hm.huajilist)  
 *if* plane.lives<=0:  
 Tk().wm\_withdraw()  
 messagebox.showwarning(**"你死了"**,message=**"正在写入分数请等待。。。"**)  
 *with open*(file=**'rank/rank.txt'**, mode=**'a'**)*as* f:  
 f.write(*str*(count)+**'**\n**'**)  
 time.sleep(1)  
 pygame.quit()  
  
 hm.generate()  
 hm.update()  
 hm.draw()  
  
 plane.move(mx\*MOVESTEP,my\*MOVESTEP)  
 plane.draw()  
  
 plane.fire()  
 plane.update\_bullets()  
 plane.draw\_bullets()  
  
 clock.tick(FPS) *# 控制循环刷新频率,每秒刷新FPS对应的值的次数* pygame.display.update()  
  
  
  
*# 敌机 - 滑稽  
class* Huaji():  
 imgpath = **"images/img\_plane\_enemy (2).png"** speed = 2  
  
 *def \_\_init\_\_*(*self*, master, x, y=0):  
 *self*.\_master = master *# 父控件  
 self*.image = pygame.image.load(*self*.imgpath)  
 *self*.x = x  
 *self*.y = y  
 *self*.lives = 1  
  
 *# 移动敌机，更新敌机位置  
 def* update(*self*):  
 *self*.y += *self*.speed  
  
 *def* draw(*self*):  
 *self*.\_master.blit(*self*.image, (*self*.x, *self*.y))  
  
 *def* inWindow(*self*):  
 *if self*.y < 0 *or self*.y > *self*.\_master.get\_height():  
 *return False  
 return True  
  
 def* get\_center\_XY(*self*):  
 *# 获取圆心坐标  
 return* (*self*.x + SMALLSIZE / 2, *self*.y + SMALLSIZE / 2)  
  
 *def* get\_radius(*self*):  
 *# 获取半径  
 return* SMALLSIZE / 2  
  
  
*class* HuajiManager():  
 cd = 35 *# 生成滑稽的时间间隔  
  
 def \_\_init\_\_*(*self*, master):  
 *self*.\_master = master  
 *self*.t = 0  
 *self*.huajilist = []  
  
 *def* generate(*self*):  
 *self*.t += 1  
 *if self*.t % *self*.cd == 0:  
 x = random.randint(0, *self*.\_master.get\_width() - SMALLSIZE)  
 ji = Huaji(*self*.\_master, x, 0)  
 *self*.huajilist.append(ji)  
  
 *def* update(*self*):  
 survive = []  
 *for* huaji *in self*.huajilist:  
 huaji.update()  
 *if* huaji.inWindow() *and* huaji.lives > 0:  
 survive.append(huaji)  
 *self*.huajilist = survive  
  
 *def* draw(*self*):  
 *for* huaji *in self*.huajilist:  
 huaji.draw()  
  
  
  
  
  
*class* Plane():  
 firedelay = 15 *# 发射子弹时间间隔  
  
 def \_\_init\_\_*(*self*, master, x, y, img\_path=PLANEIMG, img\_path\_bullets=BULLET):  
 *self*.\_master = master *# 父控件  
 self*.image = pygame.image.load(img\_path) *# 飞机图像  
 self*.image1 = pygame.image.load(img\_path\_bullets)  
 *# 飞机位置-坐标  
 self*.x = x  
 *self*.y = y  
 *self*.lives = 3  
 *self*.t = 0  
 *self*.bullets = [] *# 发射的子弹  
  
 # 移动飞机  
 def* move(*self*, x, y):  
 *if* 0 <= *self*.x + PLANESIZE / 2 + x <= *self*.\_master.get\_width():  
 *self*.x += x  
 *if* 0 <= *self*.y + PLANESIZE / 2 + y <= *self*.\_master.get\_height():  
 *self*.y += y  
  
 *# 绘制飞机  
 def* draw(*self*):  
 *self*.\_master.blit(*self*.image, (*self*.x, *self*.y))  
  
 *# 发射子弹  
 def* fire(*self*):  
 *self*.t += 1  
 *if self*.t >= *self*.firedelay:  
 *self*.t = 0  
 *# 子弹初始坐标* bx = *self*.x + *int*(*self*.image.get\_width() / 2)  
 by = *self*.y  
 bullet = Bullet(*self*.\_master, bx, by)  
 *self*.bullets.append(bullet)  
  
 *# 更新子弹位置，清除失效的子弹  
 def* update\_bullets(*self*):  
 survive = []  
 *for* b *in self*.bullets:  
 b.update()  
 *if* b.on:  
 survive.append(b)  
 *self*.bullets = survive  
  
 *# 绘制子弹  
 def* draw\_bullets(*self*):  
 *for* b *in self*.bullets:  
 b.draw()  
  
 *def* check\_all\_hit(*self*, huajilist):  
 survive = []  
 *for* b *in self*.bullets:  
 b.check\_hit(huajilist)  
 *if* b.on:  
 survive.append(b)  
 *self*.bullets = survive  
  
 *def* get\_distance(*self*, xy):  
 x, y = xy  
 cx = *self*.x + PLANESIZE / 2  
 cy = *self*.y + PLANESIZE / 2  
 *return* math.sqrt(math.pow(cx - x, 2) + math.pow(cy - y, 2))  
  
 *def* check\_crash(*self*, huajilist):  
 *for* huaji *in* huajilist:  
 *if* huaji.lives > 0 *and* huaji.inWindow():  
 d = *self*.get\_distance(huaji.get\_center\_XY())  
 *if* d <= PLANESIZE / 2 + huaji.get\_radius():  
 *# hit  
 self*.lives -= 1  
 huaji.lives -= 1  
  
  
*class* Bullet():  
 speed = 2 *# 速度* color = (255, 0, 0) *# 颜色* radius = 10 *# 半径  
  
 def \_\_init\_\_*(*self*, master, x, y, ):  
 *self*.\_master = master *# 父控件  
 self*.x = x  
 *self*.y = y  
 *self*.on = *True # 记录子弹状态，初始为True，子弹失效（超出边界或者碰到敌机）时为False  
  
 # 更新子弹位置，移动子弹  
 def* update(*self*):  
 *self*.y -= *self*.speed  
  
 *if self*.y <= 0:  
 *self*.on = *False  
  
 # 绘制飞机  
 def* draw(*self*):  
 pygame.draw.circle(*self*.\_master, *self*.color, (*self*.x, *self*.y), *self*.radius, 2)  
  
 *def* get\_distance(*self*, xy):  
 x, y = xy  
 *return* math.sqrt(math.pow(*self*.x - x, 2) + math.pow(*self*.y - y, 2))  
  
 *def* check\_hit(*self*, huajilist, ):  
 *global* count  
 *if not self*.on:  
 *return  
 for* huaji *in* huajilist:  
 *if* huaji.lives > 0 *and* huaji.inWindow():  
 d = *self*.get\_distance(huaji.get\_center\_XY())  
 *if* d <= huaji.get\_radius():  
 *# hit  
 self*.on = *False* huaji.lives -= 1  
 count += 100  
 *self*.count1 = count  
 *print*(*self*.count1)