# 准备工作：把05文件复制一份改名06-dad-son-planewar.py

|  |
| --- |
|  |

# 开发开始

## 1.新建一个enemy模块，先在里面定义一些数据以及编写一些代码加载图片

|  |
| --- |
|  |

## 2，定义一个Enemy类继承自pygame.sprite.Sprite

|  |
| --- |
|  |

## 3.给敌机类添加一个update方法

|  |
| --- |
|  |

## 4.在constants模块里面新建一个敌机精灵组

|  |
| --- |
|  |

## 5.在funcs模块里面新建一个创建敌人的函数new\_enemy函数

|  |
| --- |
|  |

## 6.在主程序里面创建敌机

|  |
| --- |
|  |

# 这一节学习到此为止，有更新代码的模块更新后的内容如下

## enemy.py

|  |
| --- |
| import pygame as pg  from constants import \*  from funcs import \*  import random as rnd  enemies\_images = [] #保存敌机图片对象的列表  enemies\_list = [      'dj1.png',      'dj2.png',      'dj3.png'  ]  # 加载敌机  for img in enemies\_list:      enemies\_img = pg.image.load(pic\_path+img)      enemies\_img = pg.transform.scale(enemies\_img,(80,60))      enemies\_images.append(enemies\_img)  # 敌机类  class Enemy(pg.sprite.Sprite):      def \_\_init\_\_(self) -> None:          pg.sprite.Sprite.\_\_init\_\_(self) # 调用父类构造函数          self.image\_orig = rnd.choice(enemies\_images) # 随机获取一张图片          self.image\_orig.set\_colorkey(BLACK)          self.image = self.image\_orig.copy()          self.rect = self.image.get\_rect()          self.radius = int(self.rect.width\*.90/2)          self.rect.x = rnd.randrange(0,WIDTH-self.rect.width)          self.rect.y = rnd.randrange(-150,-100)          self.speedy = rnd.randrange(2,5)          self.speedx = rnd.randrange(-3,3)          self.shoot\_delay = 1000          self.last\_shot = pg.time.get\_ticks()      def update(self):          self.rect.x += self.speedx          self.rect.y += self.speedy          if rnd.randrange(10) >= 6:              self.enemy\_shoot()          # 超出范围敌机重生          if(self.rect.top > HEIGHT+10) or (self.rect.left < -25) or (self.rect.right > WIDTH+20):              self.rect.x = rnd.randrange(0,WIDTH-self.rect.width)              self.rect.y = rnd.randrange(-100,-40)              self.speedy = rnd.randrange(1,5)          # 碰到两边会反弹          if self.rect.left < 0 :              self.speedx = -self.speedx          if self.rect.right > WIDTH:              self.speedx = -self.speedx      def enemy\_shoot(self):          now = pg.time.get\_ticks()          if now - self.last\_shot > self.shoot\_delay:              self.last\_shot = now              pass |

## constans.py

|  |
| --- |
| import pygame as pg  from os import path  pg.init()       #初始化pygame模块  pg.mixer.init() #初始化pygame模块的音效模块  font\_name = pg.font.match\_font('arial') # 获取系统里面安装了的字体名称  # 实战图片和音效图片路径  pic\_path = './res/images/'  sound\_path = './res/sounds/'  WIDTH = 480      #窗口宽度  HEIGHT = 600     #窗口高度  SIZE = (WIDTH,HEIGHT) #  FPS = 30   # 帧率  # 定义颜色  BLACK = (0,0,0)  WHITE = (255,255,255)  RED = (255,0,0)  GREEN = (0,255,0)  BLUE = (0,0,255)  YELLOW = (255,255,0)  bg = pg.image.load(pic\_path+'startfield.jpg')  bg = pg.transform.scale(bg,(WIDTH,1536))  height = -936  POWERUP\_TIME = 5000  #飞机的火力持续时间  # 精灵组  all\_sprites = pg.sprite.Group()  bullets = pg.sprite.Group()  # 敌机精灵组  enemies = pg.sprite.Group()  # 分数  score = 0  BAR\_LENGTH =100  #血条长度  BAR\_HEIGHT = 10  #血条高度 |

## funcs.py

|  |
| --- |
| import pygame as pg  from constants import \*  from plane import \*  from enemy import \*  #绘制文本的函数  def draw\_text(surf,text,size,x,y):      font = pg.font.Font(font\_name,size)      text\_surface = font.render(text,True,WHITE)      text\_rect = text\_surface.get\_rect()      text\_rect.midtop = (x,y)      surf.blit(text\_surface,text\_rect)  def draw\_screen\_text(screen,player1,player2):      draw\_text(screen,str(score),18,WIDTH/2,10) #显示分数      draw\_shield\_bar(screen,5,5,player1.shield)      draw\_shield\_bar(screen,WIDTH-105,5,player2.shield)      draw\_lives(screen,10,20,player1.lives,player\_mini\_img1)      draw\_lives(screen,WIDTH-100,20,player2.lives,player\_mini\_img2)  # 绘制血条  def draw\_shield\_bar(screen,x,y,pct):      pct = max(pct,0)      fill = (pct/100) \* BAR\_LENGTH      outline\_rect = pg.Rect(x,y,BAR\_LENGTH,BAR\_HEIGHT)      fill\_rect = pg.Rect(x,y,fill,BAR\_HEIGHT)      pg.draw.rect(screen,GREEN,fill\_rect)      pg.draw.rect(screen,WHITE,outline\_rect,2)  def draw\_lives(surf,x,y,lives,img):      for i in range(lives):          img\_rect = img.get\_rect()          img\_rect.x = x + 30\*i          img\_rect.y = y          surf.blit(img,img\_rect)  # 创建敌机的函数  def new\_enemy():      enemy = Enemy()      all\_sprites.add(enemy)      enemies.add(enemy) |

## 06-dad-son-planewar.py

|  |
| --- |
| # 第6个版本，敌机来袭  import sys  from constants import \*  import menu  from plane import \*  from funcs import \*  screen = pg.display.set\_mode(SIZE)  pg.display.set\_caption("飞机大战")  clock = pg.time.Clock()  def main():      # 1.显示启动画面      menu.menu\_display(screen) # 调用menu模块的显示菜单方法      # 2.精灵组可以直接使用constants模块里面的all\_sprites和bullets      # 3.创建玩家飞机      player1 = Plane(player\_img1,pg.K\_LEFT,pg.K\_RIGHT,pg.K\_UP,pg.K\_DOWN)      player2 = Plane(player\_img2,pg.K\_a,pg.K\_d,pg.K\_w,pg.K\_s)      # 4.将他们添加到精灵组,这是必须的，因为只有精灵组才有绘制方法      all\_sprites.add(player1)      all\_sprites.add(player2)      # 创建敌机      for i in range(4):          new\_enemy() # 这个方法可以创建敌机并且添加到all\_sprites和enemies精灵组里面      # 调用精灵组的更新方法      all\_sprites.update()      global height      running = True      while running: # 游戏主循环            clock.tick(FPS)          for event in pg.event.get():              if event.type == pg.QUIT:                  running = False          # 背景图片向下滚动          screen.blit(bg,(0,height))          height += 2          if height > -168:              height = -936          all\_sprites.update()       # 调用精灵组的更新方法          all\_sprites.draw(screen)   # 绘制精灵          draw\_screen\_text(screen,player1,player2) # 绘制血条和飞机架数          pg.display.update()        pg.quit()      sys.exit()  if \_\_name\_\_ == '\_\_main\_\_':      main() |