#import modules  
import pygame as pg  
import random  
import math  
pg.init()  
pg.mixer.init()  
  
#COLORS  
BLACK=(0,0,0)  
WHITE=(255,255,255)  
RED=(255,0,0)  
GREEN=(0,255,0)  
BLUE=(0,0,255)  
CYAN=(0,255,255)  
  
#display settings  
WIDTH=800  
HEIGHT=600  
icon=pg.image.load("elephant.png")  
screen=pg.display.set\_mode((WIDTH,HEIGHT))  
pg.display.set\_icon(icon)  
pg.display.set\_caption("space\_invader")  
bg=pg.image.load("d\_bg.png")  
pv=8  
bv=10  
gv=False  
out=False  
FPS=30  
lyf=3  
clock=pg.time.Clock()  
  
#functions&class  
class players(pg.sprite.Sprite):  
    def \_\_init\_\_(self):  
        pg.sprite.Sprite.\_\_init\_\_(self)  
        self.image=pg.image.load("spaceship.png")  
        #self.image.fill(GREEN)  
        #self.image.set\_colorkey(BLACK)  to avoid backroud color  
        self.rect=self.image.get\_rect()  
        self.rect.center=(WIDTH/2,570)  
        self.last\_update=pg.time.get\_ticks()  
  
    def update(self):  
        keys=pg.key.get\_pressed()  
        if keys[pg.K\_RIGHT] and self.rect.right<792:  
            self.rect.x+=pv  
        if keys[pg.K\_LEFT] and self.rect.left>8:  
            self.rect.x-=pv  
        if keys[pg.K\_UP] and self.rect.top>50:  
            self.rect.y-=pv  
        if keys[pg.K\_DOWN] and self.rect.bottom<599:  
            self.rect.y+=pv  
        if keys[pg.K\_SPACE]:  
            s=pg.mixer.Sound("shoot.wav")  
            s.play()  
            self.shoot()  
         
    def shoot(self):  
        now=pg.time.get\_ticks()  
        if (now-self.last\_update) > 200: #200ms delay  
            self.last\_update=now  
            bt=bullet(self.rect.center)  
            all\_sprites.add(bt)  
            bullets.add(bt)  
         
class mobs(pg.sprite.Sprite):  
    def \_\_init\_\_(self):  
        pg.sprite.Sprite.\_\_init\_\_(self)  
        self.image=pg.image.load(random.choice(["stone.png","stone1.png","stone2.png","stone3.png"]))            
        self.rect=self.image.get\_rect()  
        self.rect.x=random.randrange(770)  
        self.rect.bottom=random.randrange(-40,-10)  
        self.eyv=random.randrange(4,10)  
        self.exv=random.randrange(-7,7)  
  
    def update(self):  
        self.rect.x+=self.exv  
        self.rect.y+=self.eyv  
        if self.rect.y>610 or self.rect.x<-20 or self.rect.x>810:  
            self.rect.x=random.randrange(770)  
            self.rect.bottom=random.randrange(-40,-10)  
            self.eyv=random.randrange(1,8)  
            self.exv=random.randrange(-5,5)  
  
class bullet(pg.sprite.Sprite):  
    def \_\_init\_\_(self,x):  
        pg.sprite.Sprite.\_\_init\_\_(self)  
        self.image=pg.image.load("bullet.png")  
        self.rect=self.image.get\_rect()  
        self.rect.center=x[0],x[1]-30  
        #print("b=",self.rect.bottom,x)  
         
    def update(self):  
        self.rect.y-=bv  
        if self.rect.bottom<0:  
            self.kill()  
class explosion(pg.sprite.Sprite):  
    def \_\_init\_\_(self,center,size):  
        pg.sprite.Sprite.\_\_init\_\_(self)  
        self.size=size  
        self.image=exp\_anm[self.size][0]  
        self.rect=self.image.get\_rect()  
        self.rect.center=center  
        self.frame=0  
        self.frame\_rate=50  
        self.last\_update=pg.time.get\_ticks()  
    def update(self):  
        now=pg.time.get\_ticks()  
        if now - self.last\_update>self.frame\_rate:  
            self.last\_update=now  
            self.frame+=1  
            if self.frame==len(exp\_anm[self.size]):  
                self.kill()  
            else:  
                center=self.rect.center  
                self.image=exp\_anm[self.size][self.frame]  
                self.rect=self.image.get\_rect()  
                self.rect.center=center  
                 
class power(pg.sprite.Sprite):  
    def \_\_init\_\_(self,center):  
        pg.sprite.Sprite.\_\_init\_\_(self)  
        self.image=pg.image.load("heart.png")  
        self.rect=self.image.get\_rect()  
        self.rect.center=center  
        self.lyf=lyf  
  
    def update(self):  
        #lyf=self.lyf  
        #self.lyf=lyf  
        #center=self.rect.center  
        #self.rect.center=center  
        self.rect.top+=5  
        if (self.rect.top>HEIGHT):  
            self.kill()  
         
             
         
             
         
     
  
def newmob():  
    m=mobs()  
    mob.add(m)  
    all\_sprites.add(m)  
  
def print(score,x,y,z,clr):  
    font=pg.font.Font('freesansbold.ttf',z)  
    points=font.render(score,True,clr)  
    screen.blit(points,(x,y))  
  
def start():  
    screen.fill(BLACK)  
    screen.blit(bg,(0,0))  
    print("Press 5 to continue",200,150,50,WHITE)  
    if gv:  
        print("GAME OVER",220,250,62,CYAN)  
        print("SCORE:"+str(score),300,330,45,GREEN)  
    pg.display.flip()  
    loop=True  
    while loop:  
        clock.tick(FPS)  
        for event in pg.event.get():  
            if event.type == pg.QUIT:  
                pg.quit()  
            if event.type == pg.KEYUP:  
                if event.key== pg.K\_5:  
                    loop=False  
  
exp\_anm={}  
exp\_anm['lg']=[]  
exp\_anm['sm']=[]  
for i in range(9):  
    img1=pg.transform.scale((pg.image.load("regularExplosion0{}.png".format(i))),(75,75))  
    exp\_anm['lg'].append(img1)  
    img2=pg.transform.scale((pg.image.load("regularExplosion0{}.png".format(i))),(60,60))  
    exp\_anm['sm'].append(img2)  
  
#pg.mixer.music.load("background.wav")  
#pg.mixer.music.play(-1)  
  
#gaming loop  
entry=True  
run=True  
while run:  
     
    if entry:  
        start()  
        entry=False  
        #sprites  
        all\_sprites=pg.sprite.Group()  
        mob=pg.sprite.Group()  
        bullets=pg.sprite.Group()  
        player=players()  
        pwr\_grp=pg.sprite.Group()  
        all\_sprites.add(player)  
        for i in range(12):  
            newmob()  
        score=0  
         
               
    clock.tick(FPS)  
    screen.fill(BLACK)  
    screen.blit(bg,(0,0))  
  
    #events  
    for event in pg.event.get():  
        if event.type == pg.QUIT:  
            pg.quit()  
    #inputs  
    all\_sprites.update()  
     
    hits=pg.sprite.groupcollide(bullets,mob,True,True)  
    for hit  in hits:  
                newmob()  
                 
    if hits:  
        exp=explosion(hit.rect.center,"lg")  
        all\_sprites.add(exp)  
        R=random.random()  
        if R>0.9:  
             
            pwr=power(hit.rect.center)  
            all\_sprites.add(pwr)  
            pwr\_grp.add(pwr)  
        s=pg.mixer.Sound("bomb.wav")  
        s.play()  
        score+=1  
    hits=pg.sprite.spritecollide(player,pwr\_grp,True)  
    if hits and lyf<3:  
        lyf+=1  
    print("SCORE:"+str(score),20,10,20,BLUE)  
    hits=pg.sprite.spritecollide(player,mob,True)  
    if hits:  
        lyf-=1  
         
        s=pg.mixer.Sound("ggvs.wav")  
        s.play()  
        for hit in hits:  
            exp=explosion(hit.rect.center,"sm")  
            all\_sprites.add(exp)  
            newmob()  
        pg.time.delay(100)  
    if hits and lyf==0:  
                player.kill()  
                s=pg.mixer.Sound("gv.wav")  
                s.play()  
                gv=True  
                out=True  
                entry=True  
    if lyf!=0:  
        print("LYF:",550,20,20,RED)  
        q=610  
        for i in range(lyf):  
            screen.blit(pg.transform.scale(player.image,(40,40)),(q,10))  
            q+=50  
     
         
    all\_sprites.draw(screen)  
    pg.display.update()  
pg.quit()  
  
#import modules  
import pygame as pg  
import random  
import math  
pg.init()  
pg.mixer.init()  
  
#COLORS  
BLACK=(0,0,0)  
WHITE=(255,255,255)  
RED=(255,0,0)  
GREEN=(0,255,0)  
BLUE=(0,0,255)  
CYAN=(0,255,255)  
  
#display settings  
WIDTH=800  
HEIGHT=600  
icon=pg.image.load("elephant.png")  
screen=pg.display.set\_mode((WIDTH,HEIGHT))  
pg.display.set\_icon(icon)  
pg.display.set\_caption("space\_invader")  
bg=pg.image.load("d\_bg.png")  
pv=8  
bv=10  
gv=False  
out=False  
FPS=30  
lyf=3  
clock=pg.time.Clock()  
  
#functions&class  
class players(pg.sprite.Sprite):  
    def \_\_init\_\_(self):  
        pg.sprite.Sprite.\_\_init\_\_(self)  
        self.image=pg.image.load("spaceship.png")  
        #self.image.fill(GREEN)  
        #self.image.set\_colorkey(BLACK)  to avoid backroud color  
        self.rect=self.image.get\_rect()  
        self.rect.center=(WIDTH/2,570)  
        self.last\_update=pg.time.get\_ticks()  
  
    def update(self):  
        keys=pg.key.get\_pressed()  
        if keys[pg.K\_RIGHT] and self.rect.right<792:  
            self.rect.x+=pv  
        if keys[pg.K\_LEFT] and self.rect.left>8:  
            self.rect.x-=pv  
        if keys[pg.K\_UP] and self.rect.top>50:  
            self.rect.y-=pv  
        if keys[pg.K\_DOWN] and self.rect.bottom<599:  
            self.rect.y+=pv  
        if keys[pg.K\_SPACE]:  
            s=pg.mixer.Sound("shoot.wav")  
            s.play()  
            self.shoot()  
         
    def shoot(self):  
        now=pg.time.get\_ticks()  
        if (now-self.last\_update) > 200: #200ms delay  
            self.last\_update=now  
            bt=bullet(self.rect.center)  
            all\_sprites.add(bt)  
            bullets.add(bt)  
         
class mobs(pg.sprite.Sprite):  
    def \_\_init\_\_(self):  
        pg.sprite.Sprite.\_\_init\_\_(self)  
        self.image=pg.image.load(random.choice(["stone.png","stone1.png","stone2.png","stone3.png"]))            
        self.rect=self.image.get\_rect()  
        self.rect.x=random.randrange(770)  
        self.rect.bottom=random.randrange(-40,-10)  
        self.eyv=random.randrange(4,10)  
        self.exv=random.randrange(-7,7)  
  
    def update(self):  
        self.rect.x+=self.exv  
        self.rect.y+=self.eyv  
        if self.rect.y>610 or self.rect.x<-20 or self.rect.x>810:  
            self.rect.x=random.randrange(770)  
            self.rect.bottom=random.randrange(-40,-10)  
            self.eyv=random.randrange(1,8)  
            self.exv=random.randrange(-5,5)  
  
class bullet(pg.sprite.Sprite):  
    def \_\_init\_\_(self,x):  
        pg.sprite.Sprite.\_\_init\_\_(self)  
        self.image=pg.image.load("bullet.png")  
        self.rect=self.image.get\_rect()  
        self.rect.center=x[0],x[1]-30  
        #print("b=",self.rect.bottom,x)  
         
    def update(self):  
        self.rect.y-=bv  
        if self.rect.bottom<0:  
            self.kill()  
class explosion(pg.sprite.Sprite):  
    def \_\_init\_\_(self,center,size):  
        pg.sprite.Sprite.\_\_init\_\_(self)  
        self.size=size  
        self.image=exp\_anm[self.size][0]  
        self.rect=self.image.get\_rect()  
        self.rect.center=center  
        self.frame=0  
        self.frame\_rate=50  
        self.last\_update=pg.time.get\_ticks()  
    def update(self):  
        now=pg.time.get\_ticks()  
        if now - self.last\_update>self.frame\_rate:  
            self.last\_update=now  
            self.frame+=1  
            if self.frame==len(exp\_anm[self.size]):  
                self.kill()  
            else:  
                center=self.rect.center  
                self.image=exp\_anm[self.size][self.frame]  
                self.rect=self.image.get\_rect()  
                self.rect.center=center  
                 
class power(pg.sprite.Sprite):  
    def \_\_init\_\_(self,center):  
        pg.sprite.Sprite.\_\_init\_\_(self)  
        self.image=pg.image.load("heart.png")  
        self.rect=self.image.get\_rect()  
        self.rect.center=center  
        self.lyf=lyf  
  
    def update(self):  
        #lyf=self.lyf  
        #self.lyf=lyf  
        #center=self.rect.center  
        #self.rect.center=center  
        self.rect.top+=5  
        if (self.rect.top>HEIGHT):  
            self.kill()  
         
             
         
             
         
     
  
def newmob():  
    m=mobs()  
    mob.add(m)  
    all\_sprites.add(m)  
  
def print(score,x,y,z,clr):  
    font=pg.font.Font('freesansbold.ttf',z)  
    points=font.render(score,True,clr)  
    screen.blit(points,(x,y))  
  
def start():  
    screen.fill(BLACK)  
    screen.blit(bg,(0,0))  
    print("Press 5 to continue",200,150,50,WHITE)  
    if gv:  
        print("GAME OVER",220,250,62,CYAN)  
        print("SCORE:"+str(score),300,330,45,GREEN)  
    pg.display.flip()  
    loop=True  
    while loop:  
        clock.tick(FPS)  
        for event in pg.event.get():  
            if event.type == pg.QUIT:  
                pg.quit()  
            if event.type == pg.KEYUP:  
                if event.key== pg.K\_5:  
                    loop=False  
  
exp\_anm={}  
exp\_anm['lg']=[]  
exp\_anm['sm']=[]  
for i in range(9):  
    img1=pg.transform.scale((pg.image.load("regularExplosion0{}.png".format(i))),(75,75))  
    exp\_anm['lg'].append(img1)  
    img2=pg.transform.scale((pg.image.load("regularExplosion0{}.png".format(i))),(60,60))  
    exp\_anm['sm'].append(img2)  
  
#pg.mixer.music.load("background.wav")  
#pg.mixer.music.play(-1)  
  
#gaming loop  
entry=True  
run=True  
while run:  
     
    if entry:  
        start()  
        entry=False  
        #sprites  
        all\_sprites=pg.sprite.Group()  
        mob=pg.sprite.Group()  
        bullets=pg.sprite.Group()  
        player=players()  
        pwr\_grp=pg.sprite.Group()  
        all\_sprites.add(player)  
        for i in range(12):  
            newmob()  
        score=0  
         
               
    clock.tick(FPS)  
    screen.fill(BLACK)  
    screen.blit(bg,(0,0))  
  
    #events  
    for event in pg.event.get():  
        if event.type == pg.QUIT:  
            pg.quit()  
    #inputs  
    all\_sprites.update()  
     
    hits=pg.sprite.groupcollide(bullets,mob,True,True)  
    for hit  in hits:  
                newmob()  
                 
    if hits:  
        exp=explosion(hit.rect.center,"lg")  
        all\_sprites.add(exp)  
        R=random.random()  
        if R>0.9:  
             
            pwr=power(hit.rect.center)  
            all\_sprites.add(pwr)  
            pwr\_grp.add(pwr)  
        s=pg.mixer.Sound("bomb.wav")  
        s.play()  
        score+=1  
    hits=pg.sprite.spritecollide(player,pwr\_grp,True)  
    if hits and lyf<3:  
        lyf+=1  
    print("SCORE:"+str(score),20,10,20,BLUE)  
    hits=pg.sprite.spritecollide(player,mob,True)  
    if hits:  
        lyf-=1  
         
        s=pg.mixer.Sound("ggvs.wav")  
        s.play()  
        for hit in hits:  
            exp=explosion(hit.rect.center,"sm")  
            all\_sprites.add(exp)  
            newmob()  
        pg.time.delay(100)  
    if hits and lyf==0:  
                player.kill()  
                s=pg.mixer.Sound("gv.wav")  
                s.play()  
                gv=True  
                out=True  
                entry=True  
    if lyf!=0:  
        print("LYF:",550,20,20,RED)  
        q=610  
        for i in range(lyf):  
            screen.blit(pg.transform.scale(player.image,(40,40)),(q,10))  
            q+=50  
     
         
    all\_sprites.draw(screen)  
    pg.display.update()  
pg.quit()