#Import and Initialize

Import pygame and the Sprite document(called mySprites for now)

call the pygame.init()

generate a game window

def main():

#Display

assign a name for the game window

#Entities

player\_initial\_armor = 100

player\_xspeed = 1

player\_yspeed = 1

stage\_number = 1

increased\_armor = 0

increased\_lives = 0

increased\_missile\_power = 0

score = mySprites.ScoreKeeper(screen)

lives = mySprites.Lives(screen)

stage = mySprites.StageCounter(screen,stage\_number)

armor = mySprites.Armor(screen,player\_initial\_armor, increased\_armor)

boss = mySprites.Boss(screen,5 \* stage.current\_stage)

boss\_missile = []

player = mySprites.Player()

player\_missile = []

enemy = []

enemy\_missile = []

playermissileGroup = pygame.sprite.Group(player\_missile)

enemymissileGroup = pygame.sprite.Group(enemy\_missle)

bossmissileGroup = pygame.sprite.Group(boss\_missle)

aircraftGroup = pygame.sprite.Group(player, boss)

otherGroup = pygame.sprite.Group(score, lives, stage, armor)

#Action

#Assign

keepGoing = True

show\_update\_interface = False

clock = pygame.time.Clock()

timer = 0

boss\_up = False

#Loop

while keepGoing:

clock.tick(30)

#Events

for event in pygame.event.get():

time += 1

if timer == 30 and timer <=300:

enemy.append(mySprite.Enemy(1 \* stage.current\_stage()))

if time % 10 ==0:

enemy\_missile.append(5 \* stage.current\_stage())

if event.type == pygame.QUIT:

keepGoing = False

elif event.type == pygame.KEYDOWN:

if event.key == pygame.K\_LEFT:

player.change\_direction((player\_xspeed, 0))

if event.key == pygame.K\_RIGHT:

player.change\_direction((-player\_xspeed, 0))

if event.key == pygame.K\_UP:

player.change\_direction((0, player\_yspeed))

if event.key == pygame.K\_DOWN:

player.change\_direction((0, -player\_yspeed))

if event.key == pygame.K\_SPACE:

player\_missile.append(increase\_missile\_power)

elif event.type == pygame.KEYUP:

player.change\_direction((0, 0))

if pygame.sprite.spritecollide(player,enemymissileGroup,True):

armor.armor\_decrease(5 \* stage.current\_stage())

if len(enemy) == 10 and enemy[10].current\_health() == 0:

boss.append(mySprites.Boss(20 \* stage.current\_stage()))

if boss\_up == True and timer % 10 == 0:

bossMissile.append(mySprites.BossMissile(20 \* stage.current\_stafe()))

if pygame.sprite.spritecollide(player,bossmissileGroup,True):

armor.armor\_decrease(20 \* stage.current\_stage())

for a in range(len(enemy)):

if pygame.sprite.spritecollide(enemy[a],playermissileGroup,True):

enemy.lose\_health(player\_missile.current\_missile\_power())

if pygame.sprite.spritecollide(boss,playermissileGroup,True):

boss.lose\_health(player\_missile.current\_missile\_power())

if boss.current\_health() ==0 and stage.current\_stage() != 10 :

show\_update\_interface = True

if show\_update\_interface:

update\_speed = mySprites.UpdateSpeed

update\_speed = mySprites.UpdateArmor

update\_speed = mySprites.UpdateLives

for event in pygame.event.get():

if event.type == pygame.KEYDOWN:

if event.key == pygame.K\_J:

increased\_missile\_power += 1

stage\_number += 1

score = \_\_init\_\_(screen, initial\_score)

lives = \_\_init\_\_(screen)

stage = \_\_init\_\_(screen,stage\_number)

armor = \_\_init\_\_(screen,player\_initial\_armor, increased\_armor)

boss = \_\_init\_\_(screen,5 \* stage.current\_stage())

elif event.key == pygame.K\_K:

increased\_armor+= 1

stage\_number += 1

score = \_\_init\_\_(screen, initial\_score)

lives = \_\_init\_\_(screen)

stage = \_\_init\_\_(screen,stage\_number)

armor = \_\_init\_\_(screen,player\_initial\_armor, increased\_armor)

boss = \_\_init\_\_(screen,5 \* stage.current\_stage())

elif event.key == pygame.K\_L:

player\_xspeed, player\_yspeed += 1

stage\_number += 1

score = \_\_init\_\_(screen, initial\_score)

lives = \_\_init\_\_(screen)

stage = \_\_init\_\_(screen,stage\_number)

armor = \_\_init\_\_(screen,player\_initial\_armor, increased\_armor)

boss = \_\_init\_\_(screen,5 \* stage.current\_stage())

if armor.current\_armor <= 0:

lives.lose\_lives(1)

if lives.current\_lives() == 0:

keepGoing = False

if boss.current\_health() == 0 and stage\_current.stage() == 10:

keepGoing = False

playermissileGroup.clear(screen,background)

enemymissileGroup.clear(screen,background)

bossmissileGroup.clear(screen,background)

aircraftGroup.clear(screen,background)

otherGroup.clear(screen,background)

playermissileGroup.update()

enemymissileGroup.update()

bossmissileGroup.update()

aircraftGroup.update()

otherGroup.update()

playermissileGroup.draw(screen)

enemymissileGroup.draw(screen)

bossmissileGroup.draw(screen)

aircraftGroup.draw(screen)

otherGroup.draw(screen)

pygame.quit()

main()