* Create a folder on Onedrive-> Call it Space Invader game.
* In VS Code, open ‘Space invader game’ folder
* In Visual studio, click View->Terminal. Type-> pip install pygame
* Add a file in the project called main.py

In main.py:

import **pygame**

*#Initialise the pygame*

**pygame**.**init**()

*#create the screen: width 800, height 600*

screen=**pygame**.**display**.**set\_mode**((800,600))

*#Title & Icon*

**pygame**.**display**.**set\_caption**("Space Invaders")

* Start the game loop and check for the quit button

*#Game loop*

*#Checking for QUIT button pressed*

running = True

while running:

    for event in **pygame**.**event**.**get**():

        if event.type==**pygame**.QUIT:

            running=False

* Go to flaticon.com. Search for spaceship. Download 32x32 size(in pixels). Download ‘png’. Add it to the project in VS code.

icon= **pygame**.**image**.**load**("ufo.png")

**pygame**.**display**.**set\_icon**(icon)

* Change background. Anything which must be consistent, should be inside the game loop. Add a tuple inside the fill().

<https://www.rapidtables.com/convert/color/hex-to-rgb.html>

#RGB- Red, Green & Blue

*#Background color*

    screen.**fill**((0,0,0))

**pygame**.**display**.**update**()

* Add an image to the game window. Go to flaticon.com. Search for ‘arcade space’. Choose ‘png’. Download 62x64 size(in pixels). Add it to the project in VS code.

*#Player*

playerImg=**pygame**.**image**.**load**('player.png')

playerX=370

playerY=480

def **player**(x,y):

*#Draws the player on the screen*

    screen.**blit**(playerImg, (x, y))

* Add player() before the pygame .display.update(), inside the while loop

**player**(playerX, playerY)

**Keyboard input controls & key pressed event**

* Add this inside the while loop:

*#if keystroke is pressed check whether its right or left*

    if event.type==**pygame**.KEYDOWN:

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

            playerX\_change=-0.3

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

            playerX\_change=0.3

    if event.type==**pygame**.KEYUP:

        if event.key==**pygame**.K\_LEFT or event.key==**pygame**.K\_RIGHT:

            playerX\_change=0

*#Change the position of the player horizontally*

    playerX+=playerX\_change

* Add this under #player

playerX\_change=0

**Adding boundaries to the game**

*#Adding boundaries to the movement of the player.*

    if playerX<=0:

        playerX=0

    elif playerX>=736:  *#736=800(total width of screen)-64(size of spaceship 64x64)*

        playerX=736

**Creating the enemy**

* Go to flaticon.com. Search for ‘space invader’. Download 64x64 size (in pixels). Download ‘png’. Add it to the project in VS code. Call it enemy.png
* Add this code outside the while loop

*#Enemy*

enemyImg=**pygame**.**image**.**load**('enemy.png')

enemyX=370

enemyY= 490

enemyX\_change=0

def **enemy**(x,y):

    screen.**blit**(enemyImg,(x,y))

**Randomising the position of the enemy**

*#Enemy*

*enemyImg=pygame.image.load('enemy.png')*

*enemyX=random.randint(0,800)*

*enemyY= random.randint(50,150)*

**Changing the position of the enemy in X & Y direction**

#Enemy

enemyImg=pygame.image.load('enemy.png')

enemyX=random.randint(0,800)

enemyY= random.randint(50,150)

enemyX\_change=0.3

enemyY\_change=40

**Changing the position of the enemy horizontally inside the while loop**

    #Change the position of the enemy horizontally

    enemyX+=enemyX\_change

**Adding boundaries to the position of the enemy.**

   #Adding boundaries to the movement of the enemy.

    if enemyX<=0:

        enemyX\_change=0.1 #When the enemy reaches the left boundary, the position of x should be positive i.e. it should move right

        enemyY+=enemyY\_change

    elif enemyX>=736:

        enemyX\_change=-0.1  #When the enemy reaches the right boundary, the position of x should be negative i.e. it should move left

        enemyY+=enemyY\_change

**Go to freepik.com. Select an image of size 800 x 600px. Download a space image as background. Rename the image to background.avif.**

#Background

background=pygame.image.load("background.jpg")

**Add this code inside the while loop**

#Background color

    screen.fill((0,0,0))

    screen.blit(background, (0,0))

**Creating Bullets for shooting**

**Goto flaticon.com. Find a picture of the bullet. Size 32x32.**

#Initial state of the Bullet

#bullet\_state="ready" You can't see the bullet on the screen

#bullet\_state="fire", The bullet is currently moving

bulletImg=pygame.image.load('bullet.png')

bulletX=playerX

bulletY= playerY

bulletX\_change=0

bulletY\_change=-0.3

global bullet\_state

bullet\_state="ready"

def fire\_bullet(x,y):

    #Place the  bullet relative to the spaceship

    #x+16 & y+10 The bullet should appear on the centre of the spaceship

    screen.blit(bulletImg,(x+16,y+10))

**Inside the while loop:**

 #If the spacebar is pressed, the bullet should fire

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

                if bullet\_state == "ready":

                    bulletY\_change = -0.3

                    bulletX=playerX

                    bullet\_state="fire"

  if bullet\_state == "fire":

        bulletY +=bulletY\_change

        fire\_bullet(bulletX, bulletY)

    #To re-fire the bullet

    if bulletY <=0:

        bulletY = playerY

        bullet\_state = "ready"

**Collision Detection:**

**Inside the while loop**

 #Check for Collision detection

    collision = is\_collision(enemyX, enemyY, bulletX, bulletY)

    if collision:

        bulletX = playerX

        bullet\_state = "ready"

        score+=1

        print(score)

        #Generate new position of the enemy

        enemyX=random.randint(0,WIDTH)

        enemyY= random.randint(50,150)

    #end if

#Collision Detection

# Distance between two points and the midpoint: D= square root of(square of (x2-x1)+square of(y2-y1))

def is\_collision(enemyX, enemyY, bulletX, bulletY):

    distance = math.sqrt((math.pow(enemyX-bulletX,2))+ (math.pow(enemyY-bulletY, 2))) #Calculation for collision

    if distance <= COLLISION\_DISTANCE:

        return True

    else:

        return False

    #end if

# end def

**Creating multiple enemies:**

#Initial state of the Enemy

#Creating  multiple enemies

enemyImg=[]

enemyX=[]

enemyY=[]

enemyX\_change=[]

enemyY\_change=[]

for i in range(NUMBER\_OF\_ENEMIES):

    enemyImg.append(pygame.image.load('enemy.png'))

    enemyX.append(random.randint(0,WIDTH))

    enemyY.append(random.randint(50,150))

    enemyX\_change.append(0.1)

    enemyY\_change.append(10)

**Inside the While loop:**

 for i in range(NUMBER\_OF\_ENEMIES):

        #Change the position of the enemy horizontally

        enemyX[i]+=enemyX\_change[i]

        #Adding boundaries to the movement of the enemy.

        if enemyX[i]<=0:

            enemyX\_change[i]=0.1 #When the enemy reaches the left boundary, the position of x should be positive i.e. it should move right

            enemyY[i]+=enemyY\_change[i]

        elif enemyX[i]>=736:

            enemyX\_change[i]=-0.1  #When the enemy reaches the right boundary, the position of x should be negative i.e. it should move left

            enemyY[i]+=enemyY\_change[i]

        #end if

        #Check for Collision detection

        collision = is\_collision(enemyX[i], enemyY[i], bulletX, bulletY)

        if collision:

            bulletX = playerX

            bullet\_state = "ready"

            score+=1

            print(score)

            #Generate new position of the enemy

            enemyX[i]=random.randint(0,WIDTH)

            enemyY[i]= random.randint(50,150)

        #end if

        enemy(enemyImg[i],enemyX[i], enemyY[i])

      #end for loop

**Adding text and displaying the score:**

**Add Constants**

SCORE\_X=10

SCORE\_Y=10

#Variables

score=0

score\_font=pygame.font.Font('freesansbold.ttf',12)

def show\_score(score):

    score\_text=score\_font.render(f"Score:{score}",True,'#FFFFFF')

    screen.blit(score\_text, (SCORE\_X,SCORE\_Y))#draws the score on the top-left corner

    pygame.display.update()

    time.sleep(2)

#Check for Collision detection

        collision = is\_collision(enemyX[i], enemyY[i], bulletX, bulletY)

        if collision:

            bulletX = playerX

            bullet\_state = "ready"

            score+=1

            show\_score(score)

            #Generate new position of the enemy

            enemyX[i]=random.randint(0,WIDTH)

            enemyY[i]= random.randint(50,150)

        #end if

**Adding sounds and background music:**

**Tutorial:** [**https://gamedevacademy.org/pygame-mixer-tutorial-complete-guide/**](https://gamedevacademy.org/pygame-mixer-tutorial-complete-guide/)

**Go to** [**https://pixabay.com/sound-effects/search/boom/**](https://pixabay.com/sound-effects/search/boom/)

**Background music:** [**https://pixabay.com/sound-effects/search/videogame/**](https://pixabay.com/sound-effects/search/videogame/)

[**https://pixabay.com/sound-effects/8bit-music-for-game-68698/**](https://pixabay.com/sound-effects/8bit-music-for-game-68698/)

from pygame import mixer

#Background sound

mixer.music.load('background.mp3')

mixer.music.play(-1) #-1 to play the background in a loop, not just once

 #If the spacebar is pressed, the bullet should fire

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

                if bullet\_state == "ready":

                    bulletY\_change = -0.3

                    bulletX=playerX

                    bullet\_state="fire"

                    bullet\_sound=mixer.Sound('shoot.mp3')

                    bullet\_sound.play()

                #end if

            #end if

#Check for Collision detection

        collision = is\_collision(enemyX[i], enemyY[i], bulletX, bulletY)

        if collision:

            explosion\_sound=mixer.Sound('explosion.mp3')

            explosion\_sound.play()

            bulletX = playerX

            bullet\_state = "ready"

            score+=1

            show\_score(score)

            #Generate new position of the enemy

            enemyX[i]=random.randint(0,WIDTH)

            enemyY[i]= random.randint(50,150)

        #end if

**Game Over Screen:**

game\_over\_font=pygame.font.Font('freesansbold.ttf',20)

def game\_over\_text():

    game\_over\_text=game\_over\_font.render("Game over! Your score = "+str(score),True,'#FFFFFF')

    screen.blit(game\_over\_text, (SCORE\_X,30))#draws the score on the top-left corner

    pygame.display.update()

    time.sleep(5)

for i in range(NUMBER\_OF\_ENEMIES):

        #Game Over condition: When one of the enemies comes to 440 pixel, then it should display game over screen with the score

        if enemyY[i]>440:

            #Move all the enemies out of the screen i.e. place them with Y coordinate as 2000

            for j in range(NUMBER\_OF\_ENEMIES):

                enemyY[j]=2000

            game\_over\_text()

            break

        #end if

#Check for Collision detection

        collision = is\_collision(enemyX[i], enemyY[i], bulletX, bulletY)

        if collision:

            explosion\_sound=mixer.Sound('explosion.mp3')

            explosion\_sound.play()

            bulletX = playerX

            bullet\_state = "ready"

            score+=1

            show\_score(score)

            #Generate new position of the enemy

            enemyX[i]=random.randint(0,WIDTH)

            enemyY[i]= random.randint(50,150)

        #end if

Space Invader game: <https://www.youtube.com/watch?v=FfWpgLFMI7w&t=1370s>