**Space Invaders**

<https://github.com/MCKesav/Space-Invaders>

1. [Features](#Features)
2. [FIles](#FIles)

from pygame import \*

from os import \*

# a = input()

font.init()

mixer.init()

#diaplay

WIDTH, HEIGHT = 900, 700

WIN = display.set\_mode((WIDTH, HEIGHT))

display.set\_caption("Space Invaders")

icon = image.load(path.join('Assets', 'icon.png'))

display.set\_icon(icon)

#colors

WHITE = (255, 255, 255)

BLACK = (0, 0, 0)

RED = (255, 0, 0)

YELLOW = (255, 255, 0)

BLUE = (0, 0, 255)

CYAN = (0, 255, 238)

DARK\_CYAN = (0, 107, 100)

# black line

BORDER = Rect(WIDTH / 2 - 5, 0, 10, HEIGHT)

#sound

BULLET\_FIRE\_SOUND = mixer.Sound(path.join('Assets', 'bulletfiresound.mp3'))

BULLET\_HIT\_SOUNT = mixer.Sound(path.join('Assets', 'gunhitsound.mp3'))

#adding text

HEALTH\_FONT = font.SysFont("Bradley Hand ITC", 45, True)

WINNER\_FONT = font.SysFont("Bradley Hand ITC", 120, True)

SPACESHIP\_SIZE = (65, 45)

FPS = 60

vel = 5

bullets\_vel = 7

max\_bullets = 3

# health values

player1\_health = 10 # TODO

player2\_health = 10 # TODO

health\_swap\_value = 0 # TODO

YELLOW\_HIT = USEREVENT + 1

RED\_HIT = USEREVENT + 2

#images

YELLOW\_SPACESHIP\_IMG = image.load(path.join('Assets', 'spaceship\_yellow.png'))

YELLOW\_SPACESHIP = transform.rotate(

transform.scale(YELLOW\_SPACESHIP\_IMG, SPACESHIP\_SIZE), 90)

RED\_SPACESHIP\_IMG = image.load(path.join('Assets', 'spaceship\_red.png'))

RED\_SPACESHIP = transform.rotate(

transform.scale(RED\_SPACESHIP\_IMG, SPACESHIP\_SIZE), 270)

def bg\_changer(bg):

space = transform.scale(image.load(path.join('Assets', f's{bg}.webp')),(WIDTH, HEIGHT))

WIN.blit(space, (0, 0))

def draw\_window(red, yellow, red\_bullets, yellow\_bullets, player1\_health,player2\_health, bg):

bg\_changer(bg)

draw.rect(WIN, BLACK, BORDER)

# displaying health

player1\_text = HEALTH\_FONT.render("Health : " + str(player1\_health), 1,WHITE)

player2\_text = HEALTH\_FONT.render("Health : " + str(player2\_health), 1,WHITE)

WIN.blit(player2\_text, (WIDTH - player2\_text.get\_width() - 10, 7))

WIN.blit(player1\_text, (10, 10))

#displaying spaceship

WIN.blit(YELLOW\_SPACESHIP, (yellow.x, yellow.y))

WIN.blit(RED\_SPACESHIP, (red.x, red.y))

#displaying bullets

for bullet in red\_bullets:

draw.rect(WIN, RED, bullet)

for bullet in yellow\_bullets:

draw.rect(WIN, YELLOW, bullet)

display.update()

def yellow\_keys\_movements(key\_pressed, yellow):

if key\_pressed[K\_a] and (yellow.x - vel + 4 > 0):

yellow.x -= vel

if key\_pressed[K\_d] and (yellow.x + vel + yellow.width - 4 < BORDER.x):

yellow.x += vel

if key\_pressed[K\_w] and (yellow.y - vel - 50 > 0):

yellow.y -= vel

if key\_pressed[K\_s] and (yellow.y + vel + yellow.height - 4 < HEIGHT):

yellow.y += vel

def red\_keys\_movements(key\_pressed, red):

if key\_pressed[K\_LEFT] and (red.x - vel + 4 > BORDER.x + BORDER.width):

red.x -= vel

if key\_pressed[K\_RIGHT] and (red.x + vel + red.width - 4 < WIDTH):

red.x += vel

if key\_pressed[K\_UP] and (red.y - vel - 50 > 0):

red.y -= vel

if key\_pressed[K\_DOWN] and (red.y + vel + red.height - 4 < HEIGHT):

red.y += vel

def handle\_bullets(yellow\_bullets, red\_bullets, yellow, red):

for bullet in yellow\_bullets:

bullet.x += bullets\_vel

#detect collisions

if red.colliderect(bullet):

event.post(event.Event(RED\_HIT))

yellow\_bullets.remove(bullet)

elif bullet.x > WIDTH:

yellow\_bullets.remove(bullet)

for bullet in red\_bullets:

bullet.x -= bullets\_vel

#detecting collisions

if yellow.colliderect(bullet):

event.post(event.Event(YELLOW\_HIT))

red\_bullets.remove(bullet)

elif bullet.x < 0:

red\_bullets.remove(bullet)

def draw\_winner(text):

draw\_winner\_text = WINNER\_FONT.render(text, 1, WHITE)

WIN.blit(draw\_winner\_text, (WIDTH / 2 - draw\_winner\_text.get\_width() / 2, HEIGHT / 2 - draw\_winner\_text.get\_height() / 2))

display.update()

time.delay(5000)

#cheatcode for reset

def reset(bg, health\_reset\_value1, health\_reset\_value2): # TODO

global player1\_health

global player2\_health

global health\_swap\_value

if player1\_health > 10:

player1\_health = 10

if player2\_health > 10:

player2\_health = 10

if health\_swap\_value & 1 == 1:

player1\_health, player2\_health = player2\_health, player1\_health

health\_swap\_value -= 1

if health\_reset\_value1 == 1:

player1\_health -= 5

if health\_reset\_value2 == 1:

player2\_health -= 5

bg\_changer(0)

def main():

bg = 0

if bg == 0:

bg\_changer(0)

red = Rect(675 - SPACESHIP\_SIZE[0] / 2, 300, SPACESHIP\_SIZE[0] - 17,SPACESHIP\_SIZE[1] + 20)

yellow = Rect(450 / 2 - SPACESHIP\_SIZE[0] / 2, 300, SPACESHIP\_SIZE[0] - 17,SPACESHIP\_SIZE[1] + 20)

red\_bullets = []

yellow\_bullets = []

health\_reset\_value1 = 0

health\_reset\_value2 = 0

vel = 5

bullets\_vel = 7

max\_bullets = 3

global player1\_health # TODO

global player2\_health # TODO

global health\_swap\_value # TODO

clock = time.Clock()

run = True

#running loop

while run:

clock.tick(FPS)

for evt in event.get():

if evt.type == QUIT:

run = False

if evt.type == KEYDOWN:

if evt.key == K\_LCTRL and len(yellow\_bullets) < max\_bullets:

bullet = Rect(yellow.x + yellow.width,yellow.y + yellow.height / 2 - 2, 10, 5)

yellow\_bullets.append(bullet)

BULLET\_FIRE\_SOUND.play()

if evt.key == K\_RCTRL and len(red\_bullets) < max\_bullets:

bullet = Rect(red.x, red.y + red.height / 2 - 2, 10, 5)

red\_bullets.append(bullet)

BULLET\_FIRE\_SOUND.play()

# Cheat Codes

if evt.key == K\_1:

player1\_health += 5

if player1\_health < 10 and player1\_health - 5 != 0:

health\_reset\_value1 = 1

else:

health\_reset\_value1 = 0

if evt.key == K\_0:

player2\_health += 5

if player2\_health < 10 and player2\_health - 5 != 0:

health\_reset\_value2 = 1

else:

health\_reset\_value1 = 0

if evt.key == K\_2 and player2\_health - 5 > 0:

player2\_health -= 5

if evt.key == K\_9 and player1\_health - 5 > 0:

player1\_health -= 5

if evt.key == K\_h:

health\_swap\_value += 1

player1\_health, player2\_health = player2\_health, player1\_health

if evt.key == K\_b:

if bg == 5:

bg = 0

bg\_changer(0)

else:

bg += 1

bg\_changer(bg)

if evt.key == K\_r:

bg = 0

reset(bg, health\_reset\_value1, health\_reset\_value2)

#deducting health

if evt.type == YELLOW\_HIT:

player1\_health -= 1

BULLET\_HIT\_SOUNT.play()

if evt.type == RED\_HIT:

player2\_health -= 1

BULLET\_HIT\_SOUNT.play()

#display winner

winner\_text = ''

if player1\_health <= 0:

winner\_text = "Player 2 wins"

if player2\_health <= 0:

winner\_text = "Player 1 wins"

if winner\_text != '':

draw\_winner(winner\_text)

break

key\_pressed = key.get\_pressed()

yellow\_keys\_movements(key\_pressed, yellow)

red\_keys\_movements(key\_pressed, red)

handle\_bullets(yellow\_bullets, red\_bullets, yellow, red)

draw\_window(red, yellow, red\_bullets, yellow\_bullets, player1\_health,player2\_health, bg)

if \_\_name\_\_ == "\_\_main\_\_":

main()

Features

|  |  |
| --- | --- |
| **Player 1** | **Player 2** |
| **Controls** | **Controls** |
| ‘W’ for moving spaceship up | ‘Up key’ for moving spaceship up |
| ‘S’ for moving spaceship down | ‘Down key’ for moving spaceship down |
| ‘A’ for moving spaceship left | ‘Left key’ for moving spaceship left |
| ‘D’ for moving spaceship right | ‘Right key’ for moving spaceship right |
| ‘Left CTRL’ for shooting the bullets | ‘Right CTRL’ for shooting the bullets |
| **Cheat Codes** | **Cheat Codes** |
| ‘1’ for increasing your health by 5 | ‘0’ for increasing your health by 5 |
| ‘2’ for decrease other player’s health by 5 | ‘9’ for decrease other player’s health by 5 |

|  |
| --- |
| **Other Cheat Codes** |
| ‘B’ To change the background image (you can choose from 6 different backgrounds) |
| ‘R’ To undo the cheat codes |
| ‘H’ To swap the player’s health |

Files

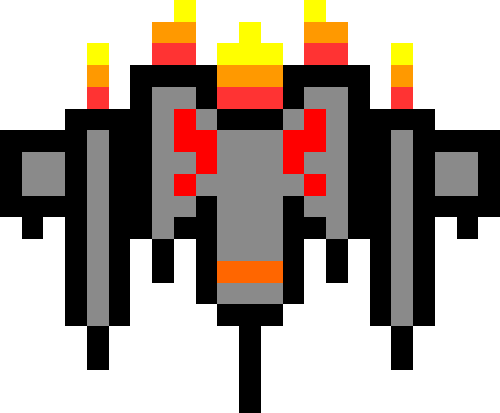
Logo



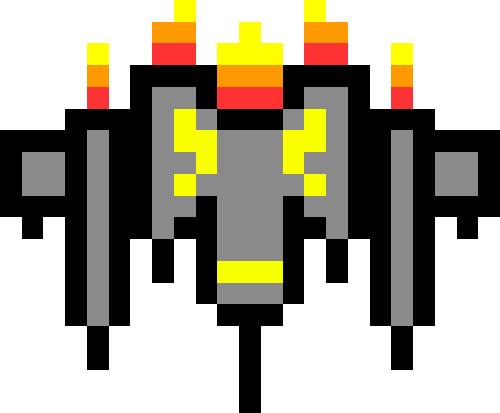
backgrounds

spaceships

red spaceship



Yellow spaceship



Mp3



The main code



The video

