PPSL PROJECT

MINI GAME

**PRN**: 21070122189

**NAME**: YASH MANGALE

**MEMBERS**:

RUTVI PORWAL, UDAY BADOLA

OBJECTIVE AND MOTIVATION:

**Project objective:**

* Design a project using python programming language.
* Manipulate different types of data using appropriate data structure such as list, string etc.
* Perform input/output function.
* Practice analyzing and debugging techniques.
* Develop good coding habits.
* To have users have fun playing the game with friends.

INTRODUCTION OF PROJECT:

The project is code based on python programming language which is an object-oriented that helps programmers write clear, logical code for small scale and large-scale projects.

The project is a multiplayer game which can be played with friends or strangers alike. It’s an arcade shooting game consisting of “spaceships”. In the game one player must hit the other by shooting bullets.

Each player has 10 lives, whoever loses all ten loses the game and the other player wins.

SOURCE CODE

import pygame

import os

pygame.font.init()

pygame.mixer.init()

WIDTH, HEIGHT = 900, 500 **#GAME WINDOW SIZE**

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

pygame.display.set\_caption("First Game!")

WHITE = (255, 255, 255) **#COLOUR NUMBERS**

BLACK = (0, 0, 0)

RED = (255, 0, 0)

YELLOW = (255, 255, 0)

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

HEALTH\_FONT = pygame.font.SysFont('comicsans', 40)

WINNER\_FONT = pygame.font.SysFont('comicsans', 100)

FPS = 60

VEL = 5 **#SPACESHIP MOVEMENT SPEED**

BULLET\_VEL = 7 **#BULLET MOVEMENT SPEED**

MAX\_BULLETS = 3 **#MAX BULLETS IN ONE GO**

SPACESHIP\_WIDTH, SPACESHIP\_HEIGHT = 55, 40

YELLOW\_HIT = pygame.USEREVENT + 1 **#UNIQUE CUSTOM EVENT**

RED\_HIT = pygame.USEREVENT + 2

YELLOW\_SPACESHIP\_IMAGE = pygame.image.load(

os.path.join('Assets', 'spaceship\_yellow.png')) **#LOADS THE IMAGE FROM FILE**

YELLOW\_SPACESHIP = pygame.transform.rotate(pygame.transform.scale(

YELLOW\_SPACESHIP\_IMAGE, (SPACESHIP\_WIDTH, SPACESHIP\_HEIGHT)), 90) **#RESCALES AND ROTATES THE IMAGE**

RED\_SPACESHIP\_IMAGE = pygame.image.load(

os.path.join('Assets', 'spaceship\_red.png'))

RED\_SPACESHIP = pygame.transform.rotate(pygame.transform.scale(

RED\_SPACESHIP\_IMAGE, (SPACESHIP\_WIDTH, SPACESHIP\_HEIGHT)), 270)

SPACE = pygame.transform.scale(pygame.image.load(

os.path.join('Assets', 'space.png')), (WIDTH, HEIGHT)) **#BACKGROUND**

def draw\_window(red, yellow, red\_bullets, yellow\_bullets, red\_health, yellow\_health): **#SHOWS COLOUR IN THE RUNNING WIDOW**

WIN.blit(SPACE, (0, 0))

pygame.draw.rect(WIN, BLACK, BORDER)

red\_health\_text = HEALTH\_FONT.render(

"Health: " + str(red\_health), 1, WHITE)

yellow\_health\_text = HEALTH\_FONT.render(

"Health: " + str(yellow\_health), 1, WHITE)

WIN.blit(red\_health\_text, (WIDTH - red\_health\_text.get\_width() - 10, 10))

WIN.blit(yellow\_health\_text, (10, 10))

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

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

for bullet in red\_bullets:

pygame.draw.rect(WIN, RED, bullet)

for bullet in yellow\_bullets:

pygame.draw.rect(WIN, YELLOW, bullet)

pygame.display.update()

def yellow\_handle\_movement(keys\_pressed, yellow): **#APPLIES AND CHECKS PLAYER MOVEMENT**

if keys\_pressed[pygame.K\_a] and yellow.x - VEL > 0: # LEFT

yellow.x -= VEL

if keys\_pressed[pygame.K\_d] and yellow.x + VEL + yellow.width < BORDER.x: # RIGHT

yellow.x += VEL

if keys\_pressed[pygame.K\_w] and yellow.y - VEL > 0: # UP

yellow.y -= VEL

if keys\_pressed[pygame.K\_s] and yellow.y + VEL + yellow.height < HEIGHT - 15: # DOWN

yellow.y += VEL

def red\_handle\_movement(keys\_pressed, red): **#APPLIES AND CHECKS PLAYER MOVEMENT**

if keys\_pressed[pygame.K\_LEFT] and red.x - VEL > BORDER.x + BORDER.width: # LEFT

red.x -= VEL

if keys\_pressed[pygame.K\_RIGHT] and red.x + VEL + red.width < WIDTH: # RIGHT

red.x += VEL

if keys\_pressed[pygame.K\_UP] and red.y - VEL > 0: # UP

red.y -= VEL

if keys\_pressed[pygame.K\_DOWN] and red.y + VEL + red.height < HEIGHT - 15: # DOWN

red.y += VEL

def handle\_bullets(yellow\_bullets, red\_bullets, yellow, red): **#CHECKS IF BULLET HIT**

for bullet in yellow\_bullets:

bullet.x += BULLET\_VEL

if red.colliderect(bullet):

pygame.event.post(pygame.event.Event(RED\_HIT)) **#CREATES THE UNIQUE EVENT**

yellow\_bullets.remove(bullet)

elif bullet.x > WIDTH:

yellow\_bullets.remove(bullet)

for bullet in red\_bullets:

bullet.x -= BULLET\_VEL

if yellow.colliderect(bullet):

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

red\_bullets.remove(bullet)

elif bullet.x < 0:

red\_bullets.remove(bullet)

def draw\_winner(text): **#SHOWS WHO THE WINNER IS**

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

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

2, HEIGHT/2 - draw\_text.get\_height()/2))

pygame.display.update()

pygame.time.delay(5000)

def main(): **#OPENS THE MAIN FUNCTION**

red = pygame.Rect(700, 300, SPACESHIP\_WIDTH, SPACESHIP\_HEIGHT)

yellow = pygame.Rect(100, 300, SPACESHIP\_WIDTH, SPACESHIP\_HEIGHT)

red\_bullets = []

yellow\_bullets = []

red\_health = 10

yellow\_health = 10

clock = pygame.time.Clock()

run = True

while run:

clock.tick(FPS)

for event in pygame.event.get():

if event.type == pygame.QUIT:

run = False

pygame.quit()

if event.type == pygame.KEYDOWN:

if event.key == pygame.K\_LCTRL and len(yellow\_bullets) < MAX\_BULLETS:

bullet = pygame.Rect(

yellow.x + yellow.width, yellow.y + yellow.height//2 - 2, 10, 5)

yellow\_bullets.append(bullet)

if event.key == pygame.K\_RCTRL and len(red\_bullets) < MAX\_BULLETS:

bullet = pygame.Rect(

red.x, red.y + red.height//2 - 2, 10, 5)

red\_bullets.append(bullet) **#MOVES THE BULLET**

if event.type == RED\_HIT:

red\_health -= 1

if event.type == YELLOW\_HIT:

yellow\_health -= 1

winner\_text = ""

if red\_health <= 0:

winner\_text = "Yellow Wins!"

if yellow\_health <= 0:

winner\_text = "Red Wins!"

if winner\_text != "":

draw\_winner(winner\_text)

break

keys\_pressed = pygame.key.get\_pressed()

yellow\_handle\_movement(keys\_pressed, yellow)

red\_handle\_movement(keys\_pressed, red)

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

draw\_window(red, yellow, red\_bullets, yellow\_bullets,

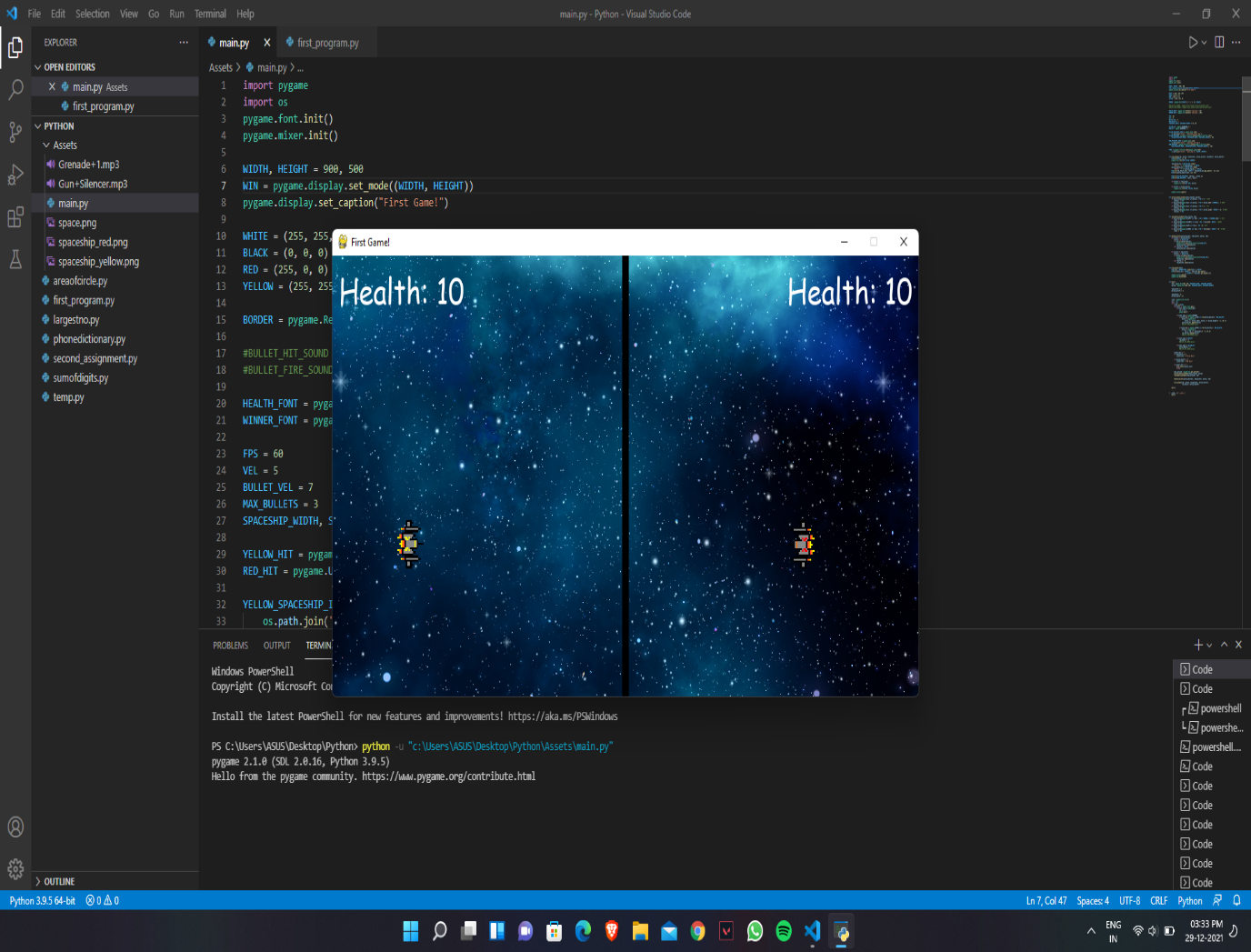
red\_health, yellow\_health)

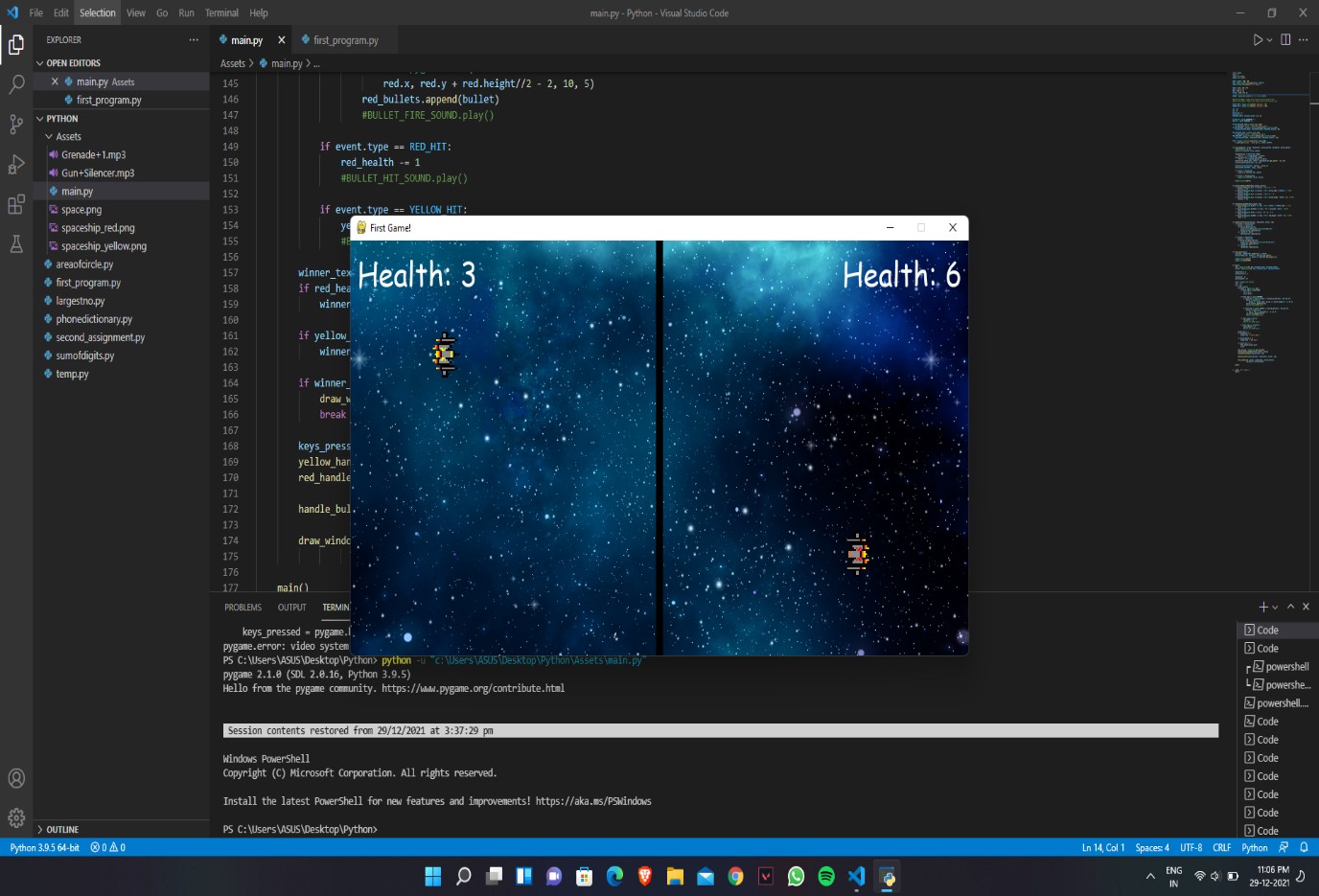
main()

if name == "main":

main()

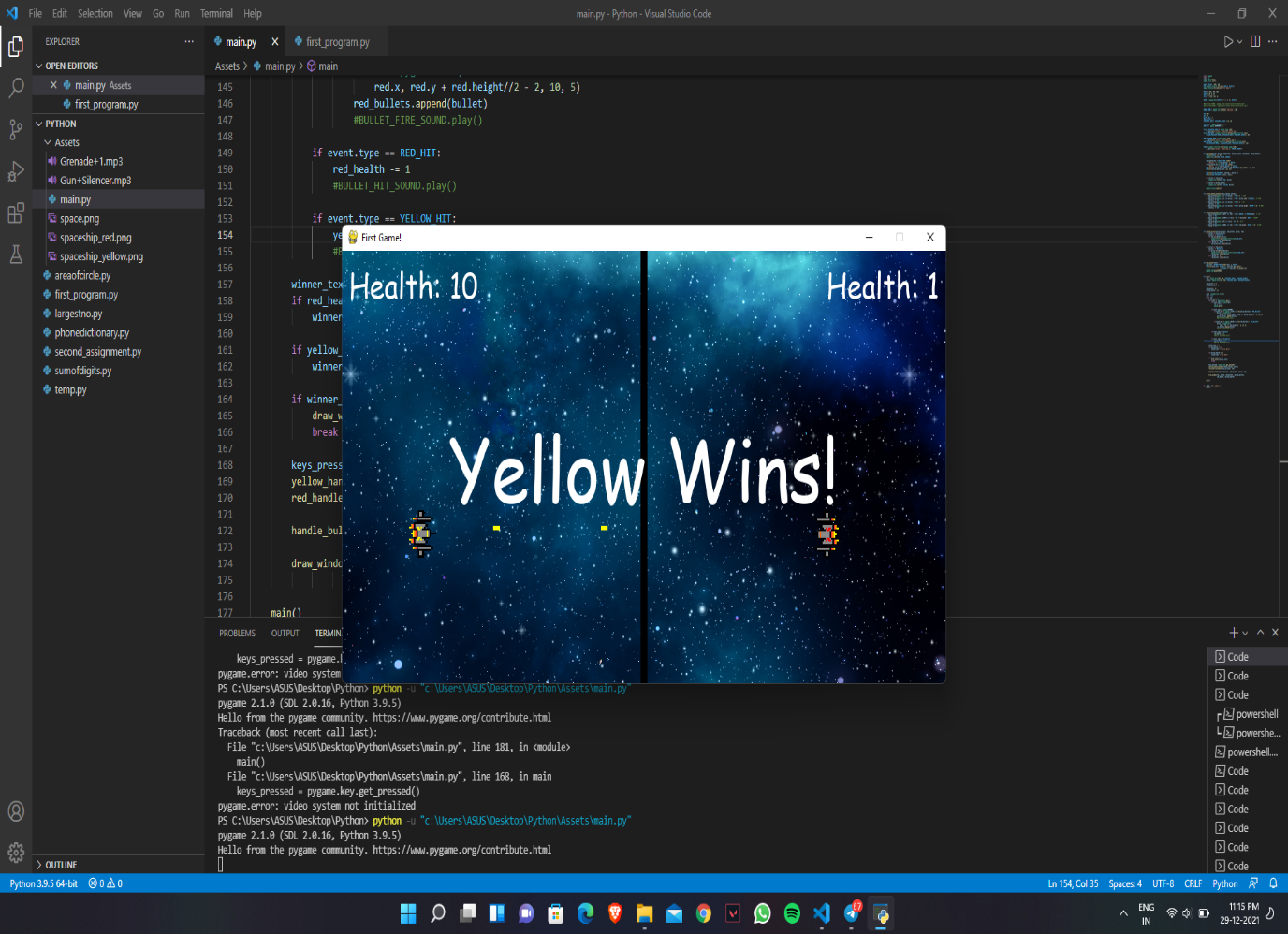
output and result





A screenshot of a computer

Description automatically generated with medium confidence



CONCLUSION

This is a project made with the python programming language. The program is designed for playing and having fun with friends. This program contains functions, modules. This program can be modified to do more things.

Thanks to all who helped me to complete this project, especially our PPS teacher.

Reference

<https://youtu.be/jO6qQDNa2UY>

thank you