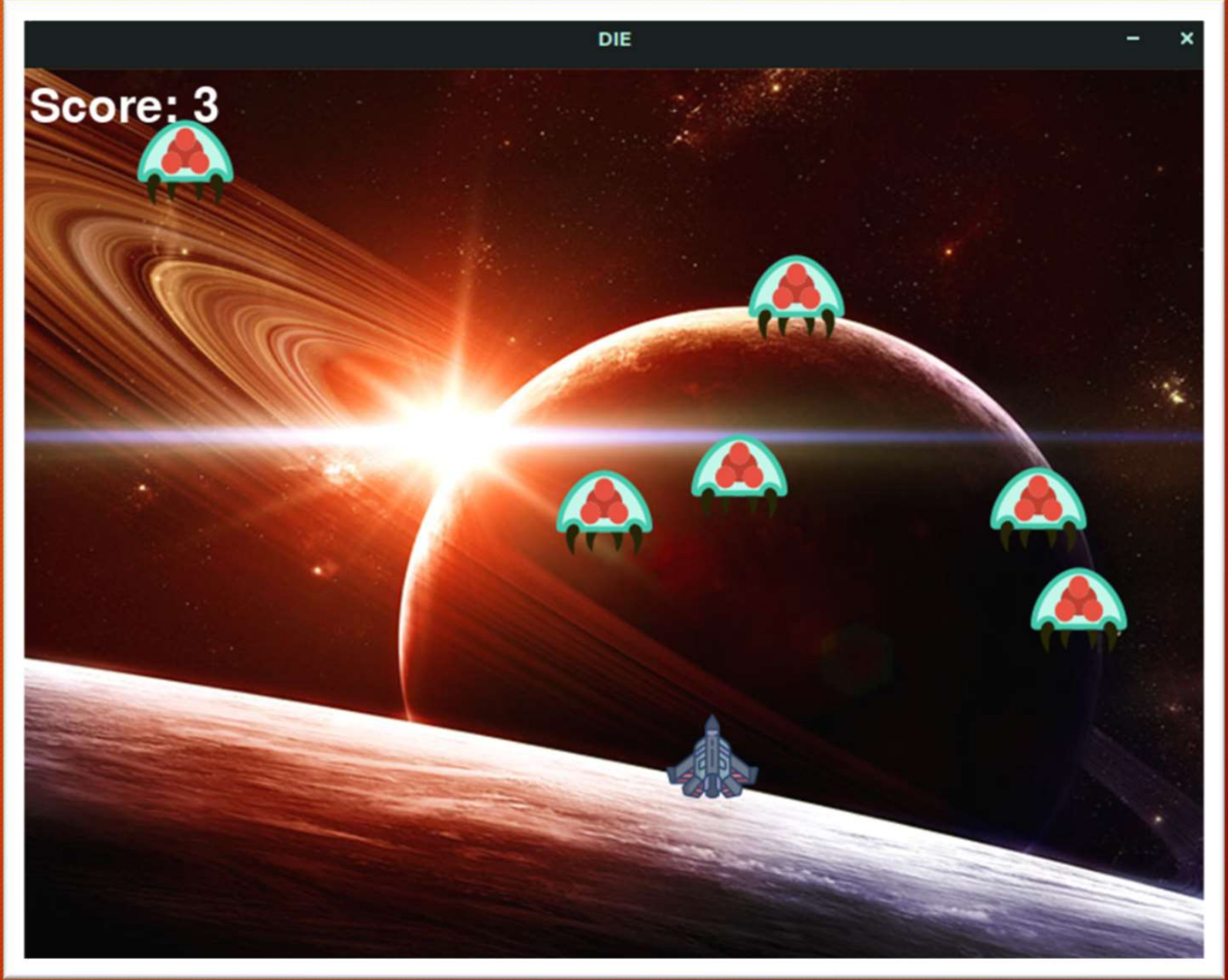
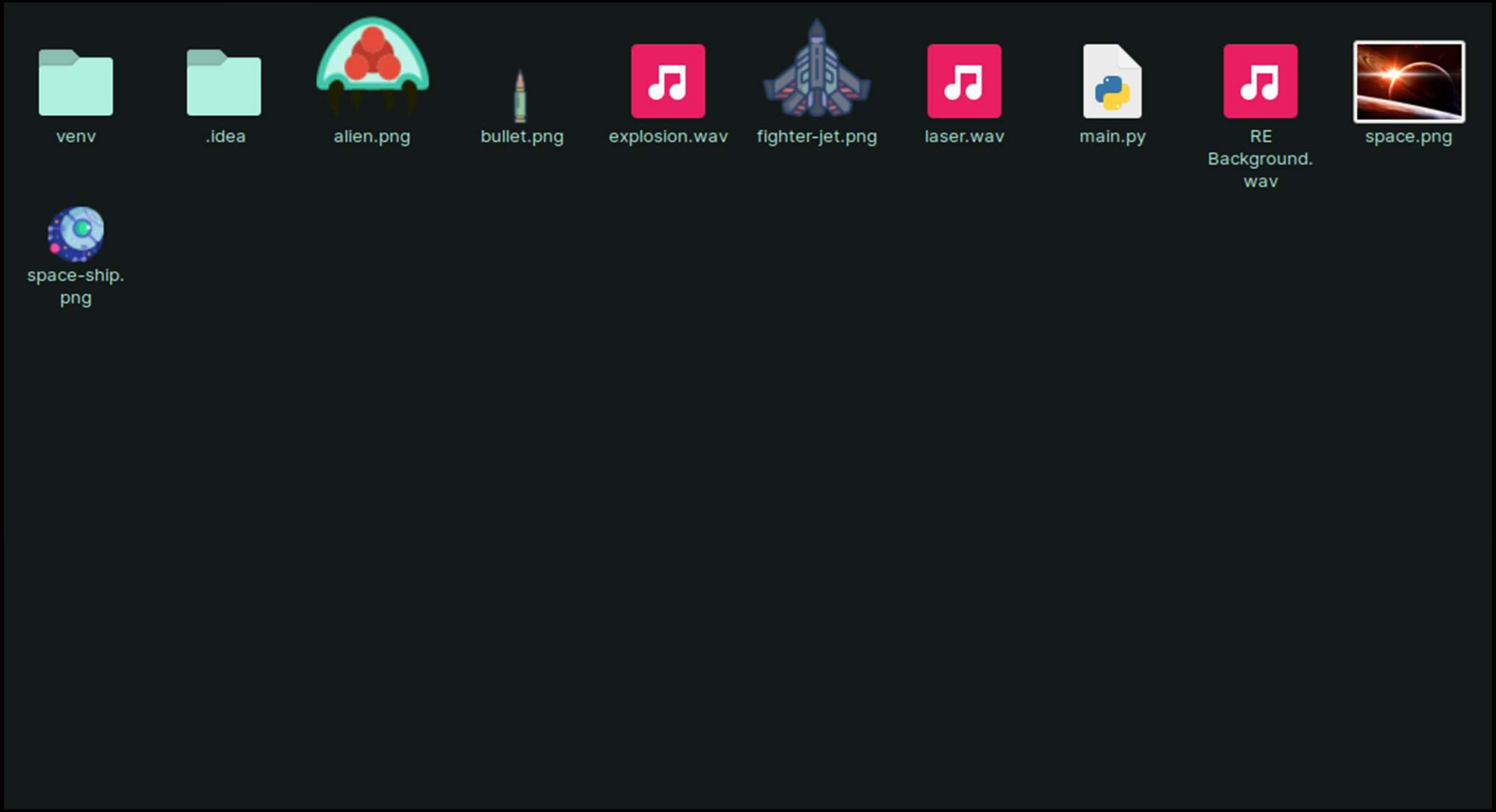
Pygame Project

Brief Introduction - Maaz

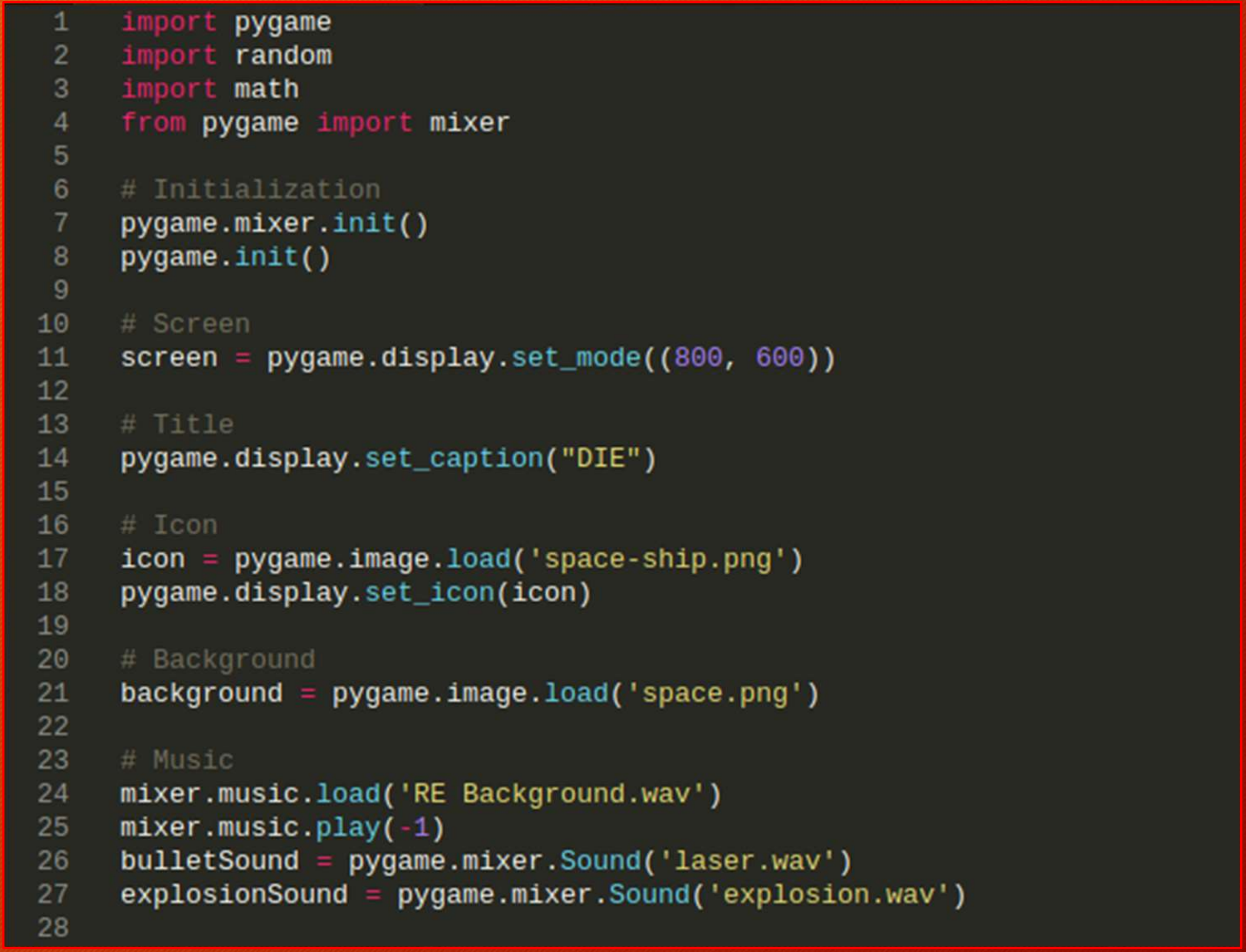
- Arcade Style
- Uses pygame
- Multiple Game Over mechanisms
- Score Counter




venv
.idea
alien.png
bullet.png
explosion.wav
fighter-jet.png
laser.wav
main.py
RE Background.wav
space.png
space-ship.png

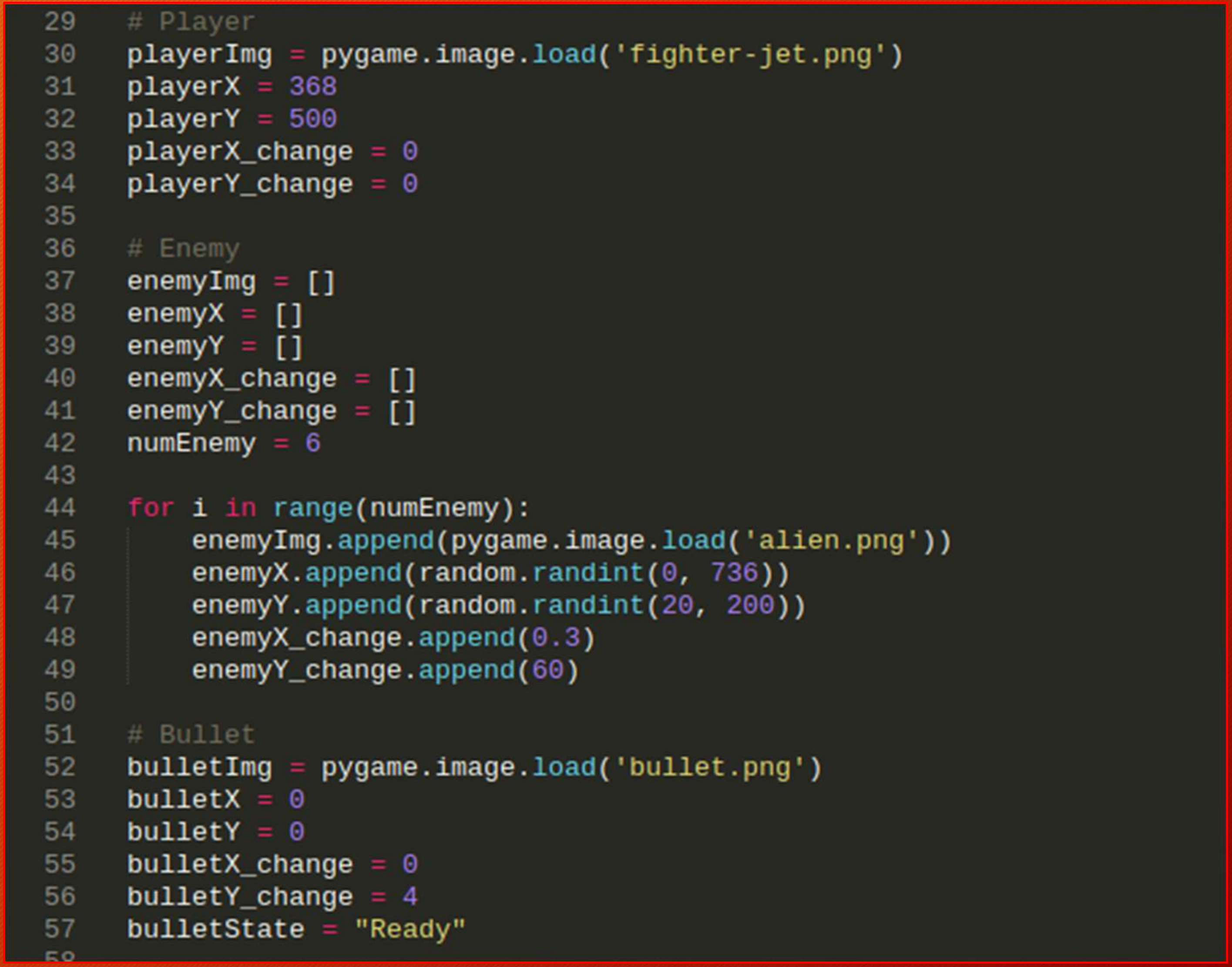
Adding Elements - Amrit

- Importing Modules
- Initializing
- Creating Display
- Adding Title
- Adding Images
- Adding Sound
- Python 3.7



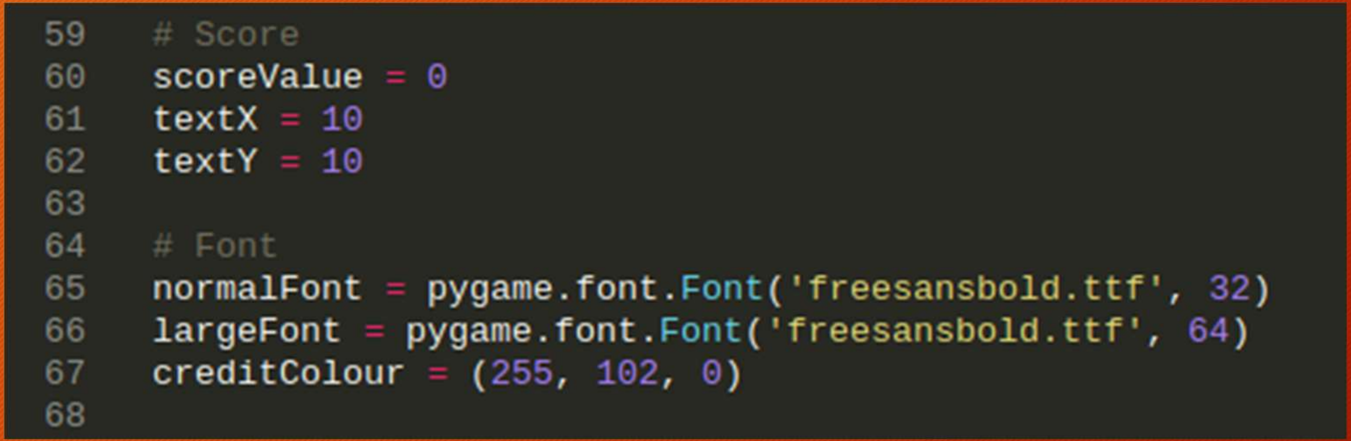
```python
import pygame
import random
import math
from pygame import mixer

# Initialization
pygame.mixer.init()
pygame.init()

# Screen
screen = pygame.display.set_mode((800, 600))

# Title
pygame.display.set_caption("DIE")

# Icon
icon = pygame.image.load('space-ship.png')
pygame.display.set_icon(icon)

# Background
background = pygame.image.load('space.png')

# Music
mixer.music.load('RE Background.wav')
mixer.music.play(-1)
bulletSound = pygame.mixer.Sound('laser.wav')
explosionSound = pygame.mixer.Sound('explosion.wav')

```

- Adding Images
- Coordinates
- Multiple enemies
- Random Spawn
- Bullet Movement

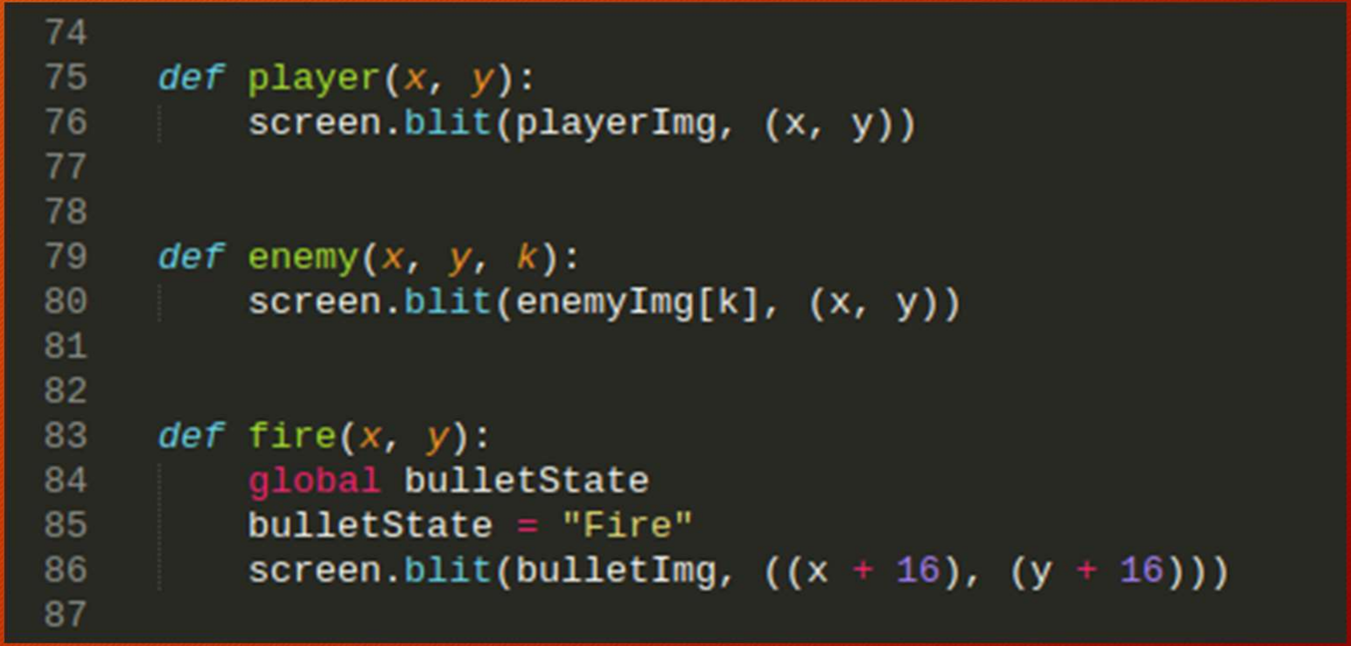


```python
# Player
playerImg = pygame.image.load('fighter-jet.png')
playerX = 368
playerY = 500
playerX_change = 0
playerY_change = 0

# Enemy
enemyImg = []
enemyX = []
enemyY = []
enemyX_change = []
enemyY_change = []
numEnemy = 6

for i in range(numEnemy):
    enemyImg.append(pygame.image.load('alien.png'))
    enemyX.append(random.randint(0, 736))
    enemyY.append(random.randint(20, 200))
    enemyX_change.append(0.3)
    enemyY_change.append(60)

# Bullet
bulletImg = pygame.image.load('bullet.png')
bulletX = 0
bulletY = 0
bulletX_change = 0
bulletY_change = 4
bulletState = "Ready"
```

- Score
- Fonts
- Using blit



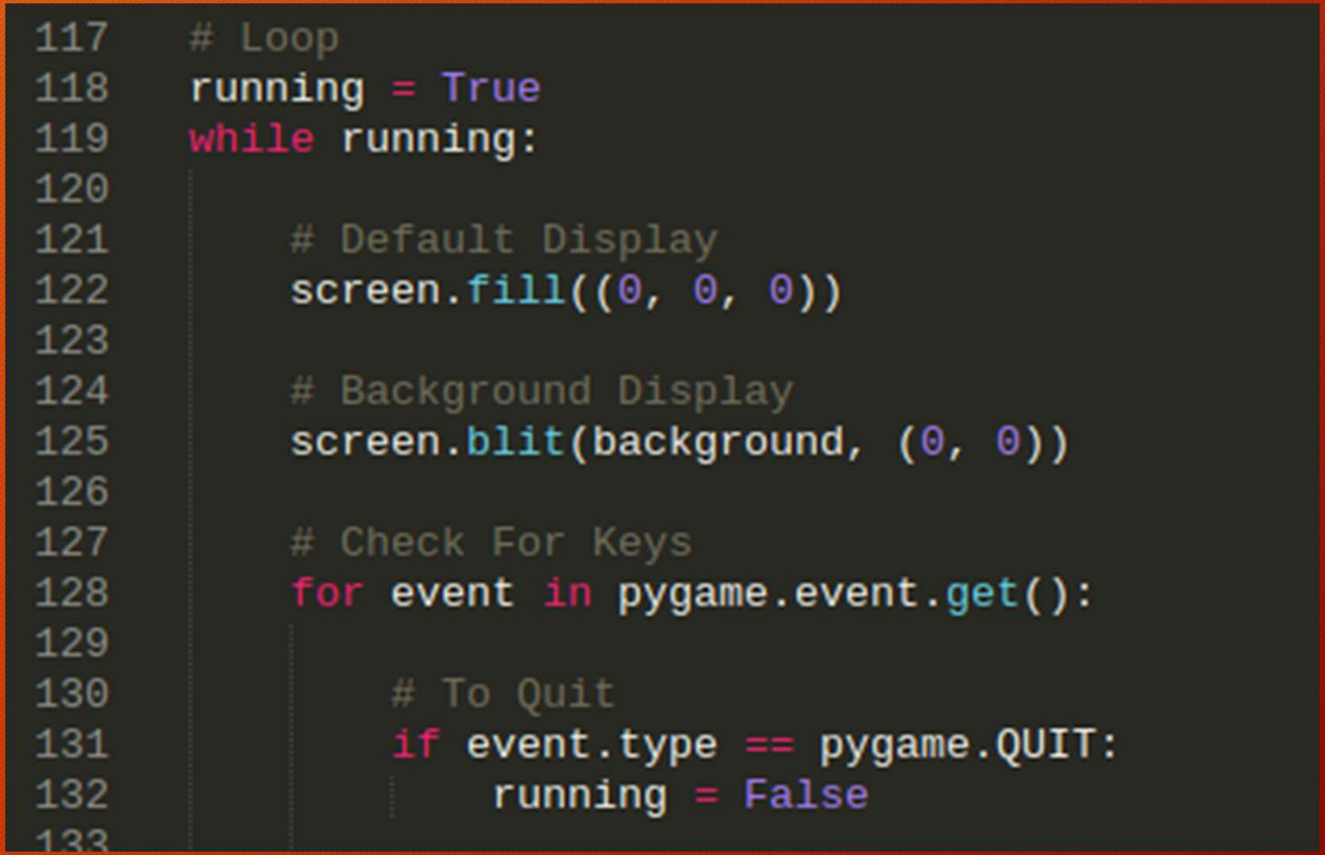
```python
# Score
scoreValue = 0
textX = 10
textY = 10

# Font
normalFont = pygame.font.Font('freesansbold.ttf', 32)
largeFont = pygame.font.Font('freesansbold.ttf', 64)
creditColour = (255, 102, 0)

```



```python

def player(x, y):
    screen.blit(playerImg, (x, y))


def enemy(x, y, k):
    screen.blit(enemyImg[k], (x, y))


def fire(x, y):
    global bulletState
    bulletState = "Fire"
    screen.blit(bulletImg, ((x + 16), (y + 16)))

```

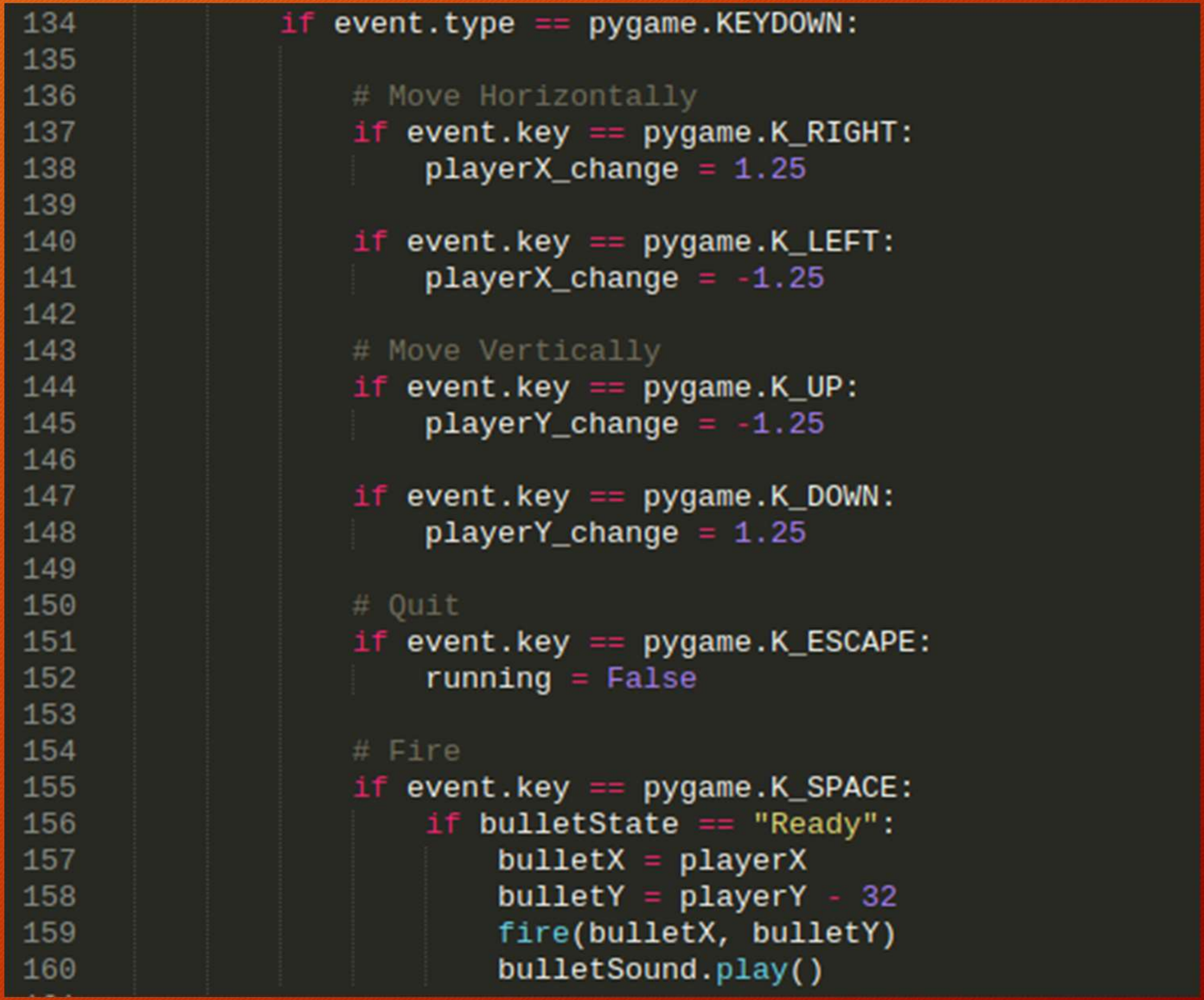
Controls and Boundary - Sarthak

- Condition for running
- Default Background
- Quit Condition



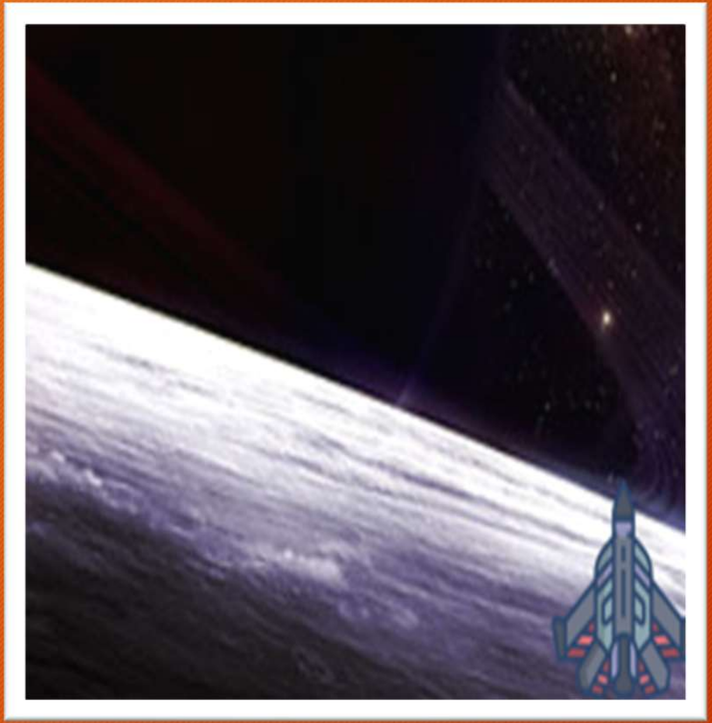
```
# Loop
running = True
while running:

    # Default Display
    screen.fill((0, 0, 0))

    # Background Display
    screen.blit(background, (0, 0))

    # Check For Keys
    for event in pygame.event.get():

        # To Quit
        if event.type == pygame.QUIT:
            running = False
```

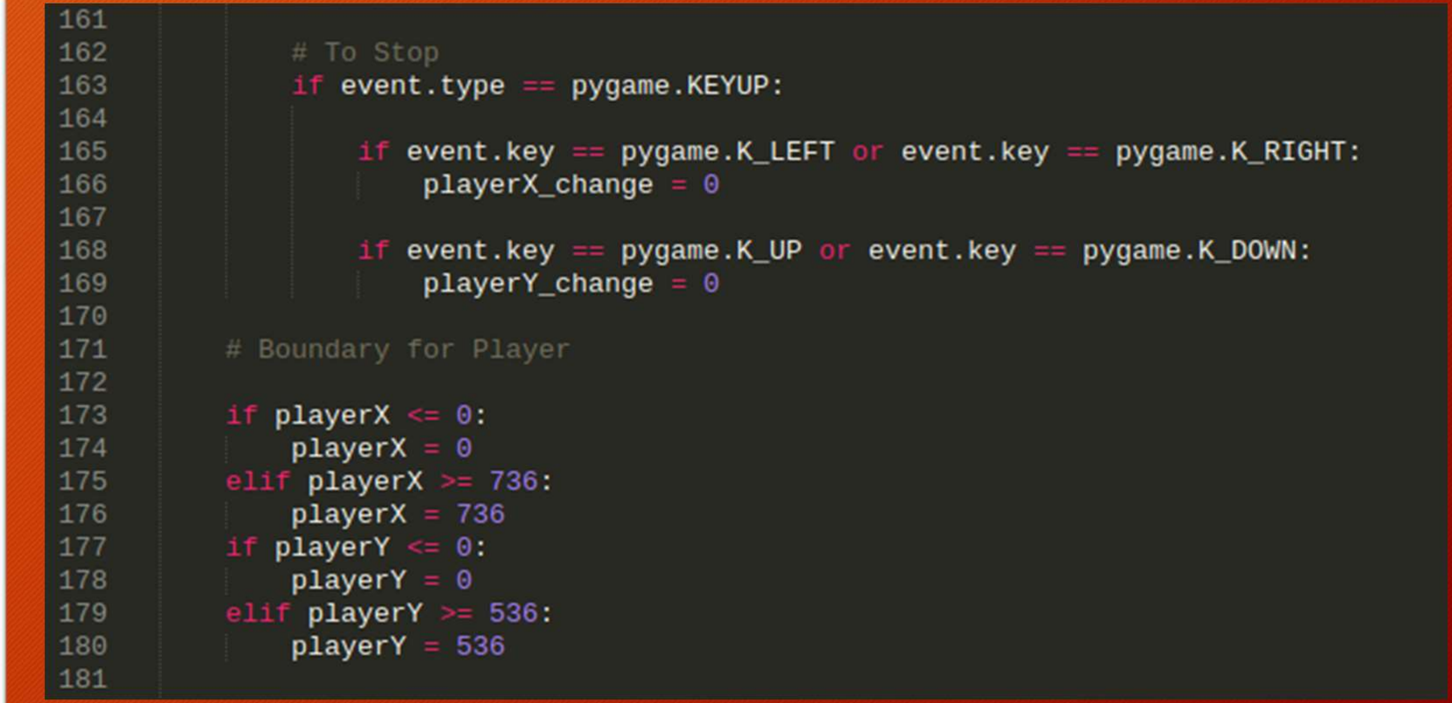
- Keys Used to manipulate values
- Keydown checks whether key is pressed



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

            # Move Horizontally
            if event.key == pygame.K_RIGHT:
                playerX_change = 1.25

            if event.key == pygame.K_LEFT:
                playerX_change = -1.25

            # Move Vertically
            if event.key == pygame.K_UP:
                playerY_change = -1.25

            if event.key == pygame.K_DOWN:
                playerY_change = 1.25

            # Quit
            if event.key == pygame.K_ESCAPE:
                running = False

            # Fire
            if event.key == pygame.K_SPACE:
                if bulletState == "Ready":
                    bulletX = playerX
                    bulletY = playerY - 32
                    fire(bulletX, bulletY)
                    bulletSound.play()
```

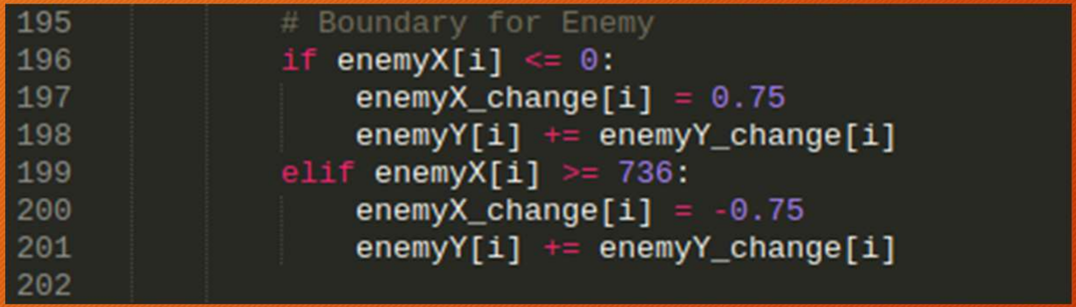
- Keyup to stop movement
- Boundary Condition



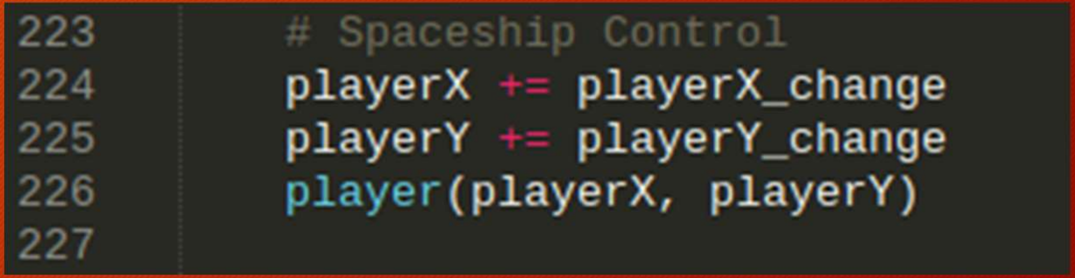


```
        # To Stop
        if event.type == pygame.KEYUP:

            if event.key == pygame.K_LEFT or event.key == pygame.K_RIGHT:
                playerX_change = 0

            if event.key == pygame.K_UP or event.key == pygame.K_DOWN:
                playerY_change = 0

    # Boundary for Player

    if playerX <= 0:
        playerX = 0
    elif playerX >= 736:
        playerX = 736
    if playerY <= 0:
        playerY = 0
    elif playerY >= 536:
        playerY = 536
```

- Changes in enemy x and y axis
- Constantly blitting spaceship



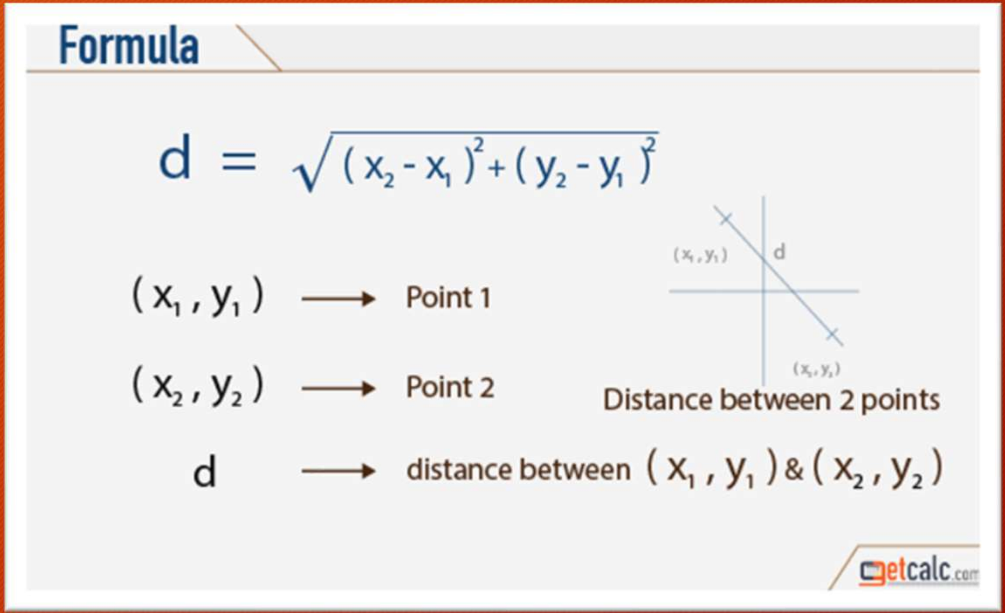
```
        # Boundary for Enemy
        if enemyX[i] <= 0:
            enemyX_change[i] = 0.75
            enemyY[i] += enemyY_change[i]
        elif enemyX[i] >= 736:
            enemyX_change[i] = -0.75
            enemyY[i] += enemyY_change[i]

```



```
    # Spaceship Control
    playerX += playerX_change
    playerY += playerY_change
    player(playerX, playerY)

```

Collision and Movement - Harshita

- Coordinates



Formula

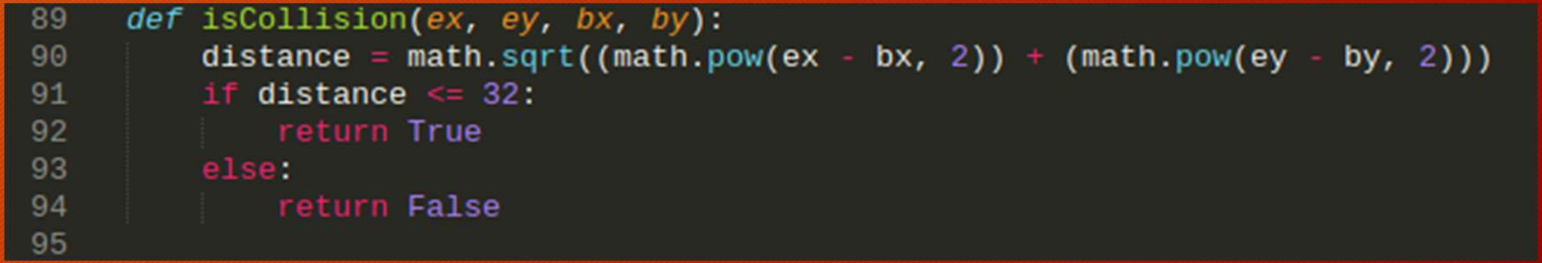
$$d = \sqrt{(x_2 - x_1)^2 + (y_2 - y_1)^2}$$

(x_1, y_1) → Point 1

(x_2, y_2) → Point 2

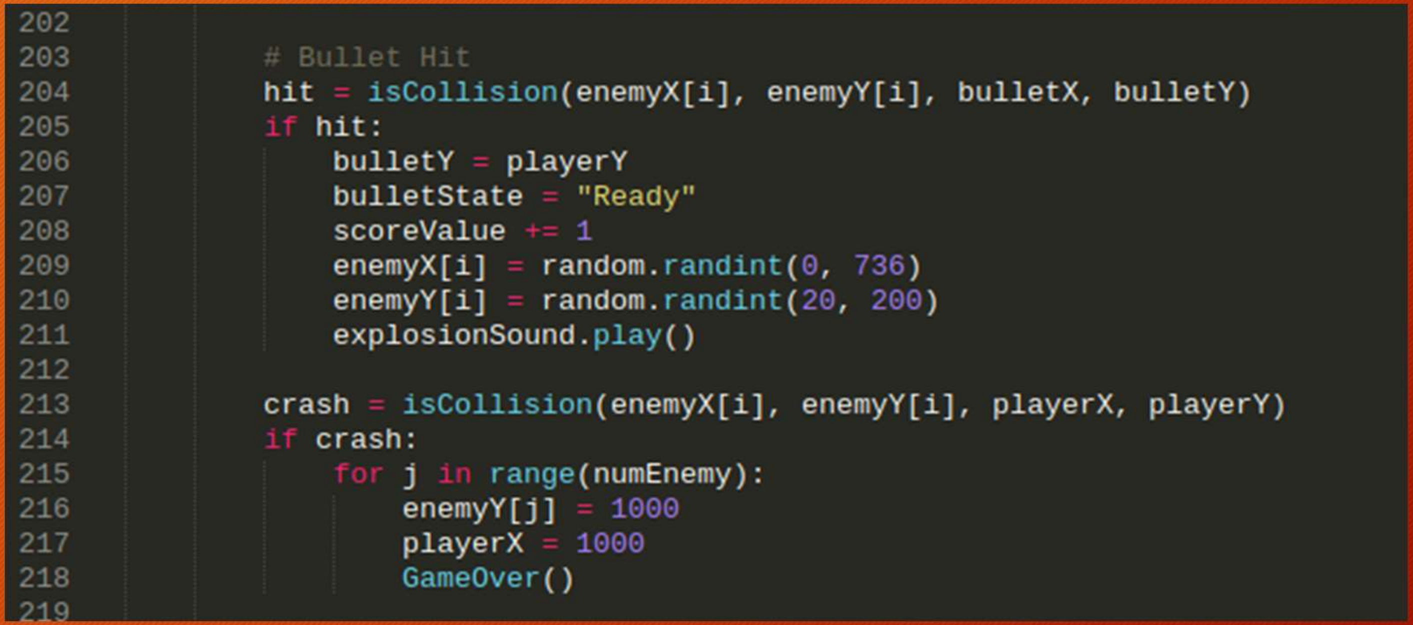
d → distance between (x_1, y_1) & (x_2, y_2)

Distance between 2 points

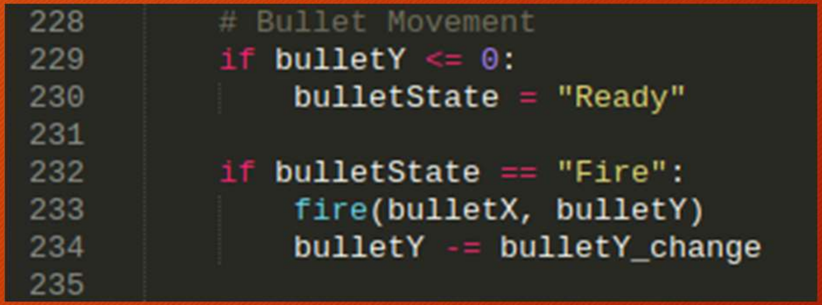


```python
def isCollision(ex, ey, bx, by):
    distance = math.sqrt((math.pow(ex - bx, 2)) + (math.pow(ey - by, 2)))
    if distance <= 32:
        return True
    else:
        return False

```

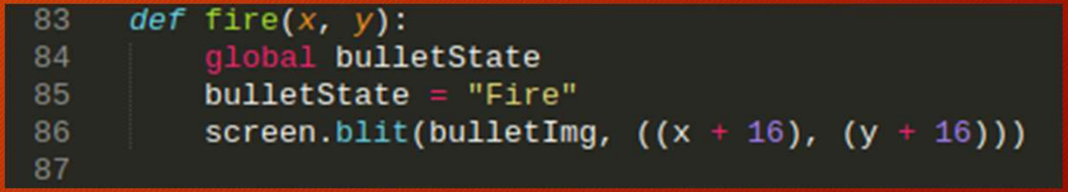
- Conditions of Collision
- Sounds
- Bullet States



```
            # Bullet Hit
            hit = isCollision(enemyX[i], enemyY[i], bulletX, bulletY)
            if hit:
                bulletY = playerY
                bulletState = "Ready"
                scoreValue += 1
                enemyX[i] = random.randint(0, 736)
                enemyY[i] = random.randint(20, 200)
                explosionSound.play()

            crash = isCollision(enemyX[i], enemyY[i], playerX, playerY)
            if crash:
                for j in range(numEnemy):
                    enemyY[j] = 1000
                    playerX = 1000
                    GameOver()
```



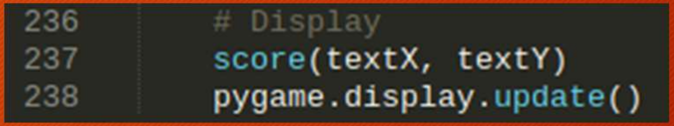
```
        # Bullet Movement
        if bulletY <= 0:
            bulletState = "Ready"

        if bulletState == "Fire":
            fire(bulletX, bulletY)
            bulletY -= bulletY_change
```



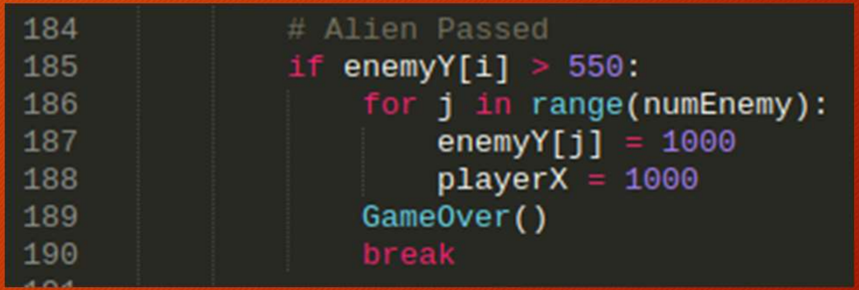
```
def fire(x, y):
    global bulletState
    bulletState = "Fire"
    screen.blit(bulletImg, ((x + 16), (y + 16)))
```

Conclusion and Gameplay - Suman

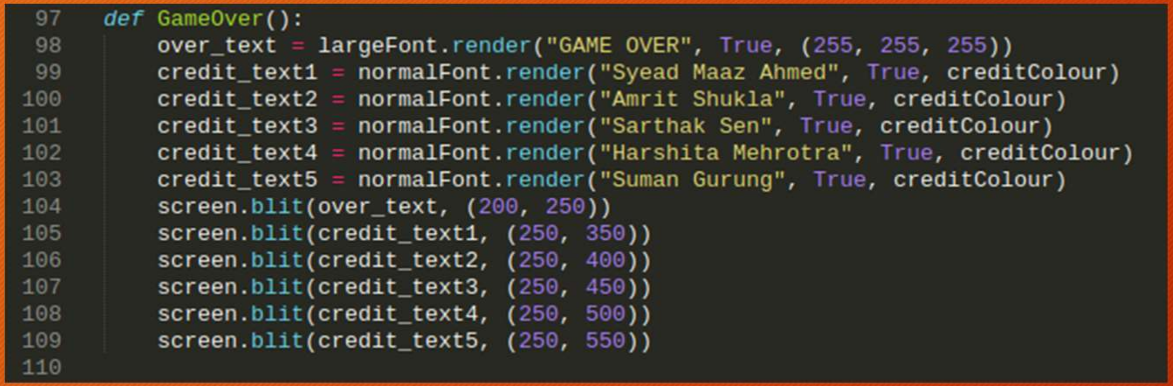
- Conditions of Game Over
- What Exactly Happens?
- Note- Score Updates



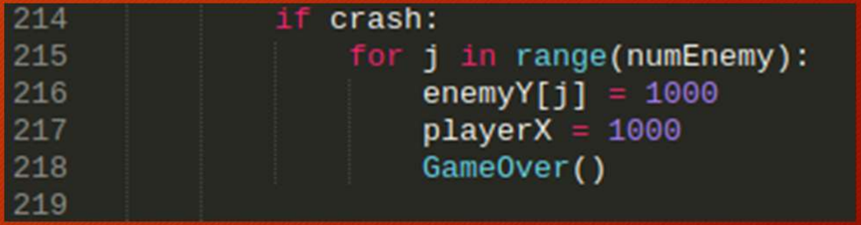
```
        # Display
        score(textX, textY)
        pygame.display.update()
```



```
            # Alien Passed
            if enemyY[i] > 550:
                for j in range(numEnemy):
                    enemyY[j] = 1000
                    playerX = 1000
                GameOver()
                break
```

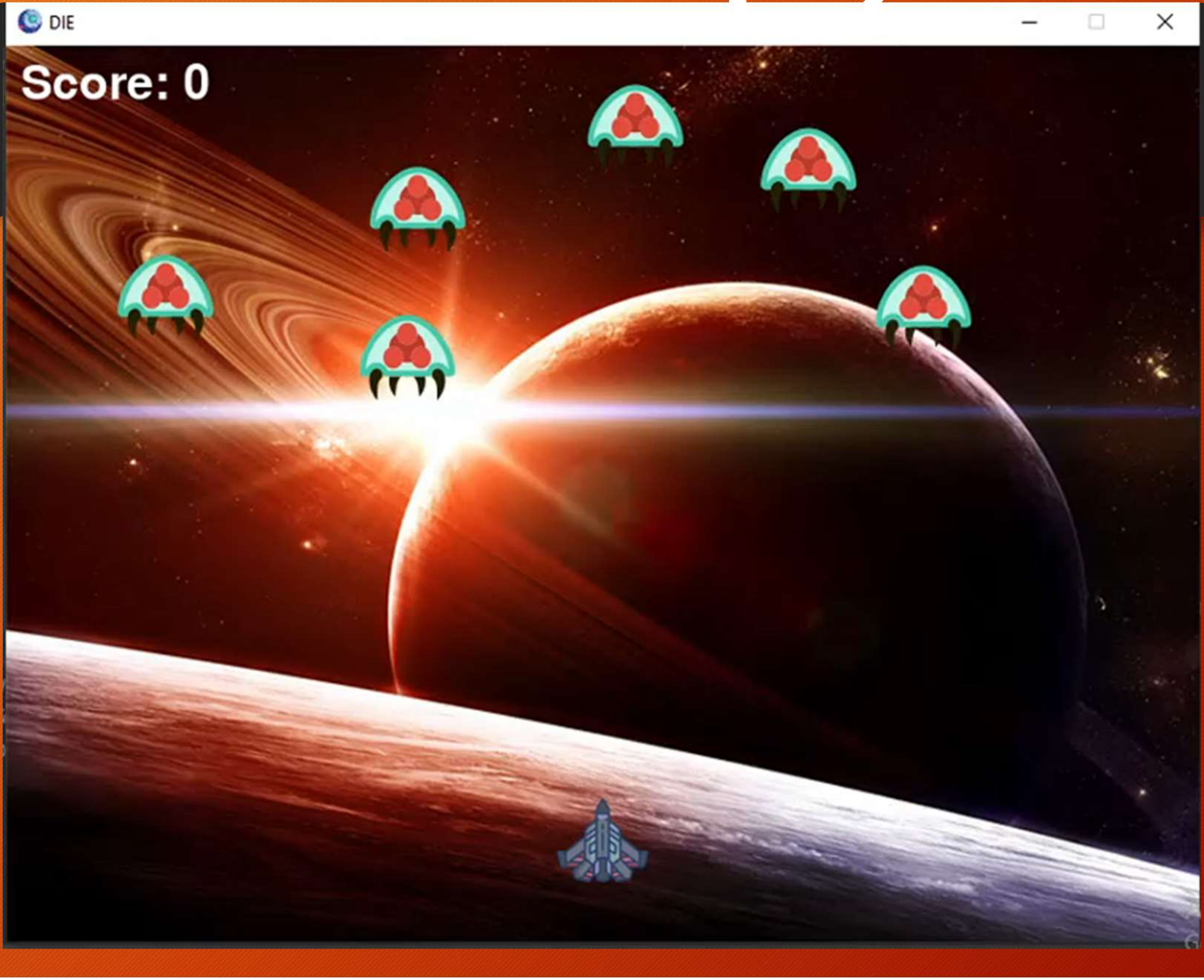


```
def GameOver():
    over_text = largeFont.render("GAME OVER", True, (255, 255, 255))
    credit_text1 = normalFont.render("Syead Maaz Ahmed", True, creditColour)
    credit_text2 = normalFont.render("Amrit Shukla", True, creditColour)
    credit_text3 = normalFont.render("Sarthak Sen", True, creditColour)
    credit_text4 = normalFont.render("Harshita Mehrotra", True, creditColour)
    credit_text5 = normalFont.render("Suman Gurung", True, creditColour)
    screen.blit(over_text, (200, 250))
    screen.blit(credit_text1, (250, 350))
    screen.blit(credit_text2, (250, 400))
    screen.blit(credit_text3, (250, 450))
    screen.blit(credit_text4, (250, 500))
    screen.blit(credit_text5, (250, 550))
```



```
            if crash:
                for j in range(numEnemy):
                    enemyY[j] = 1000
                    playerX = 1000
                    GameOver()
```

And Now… Some Gameplay



Thank You