

9.	24/09/25	Implement Exceptions and		
		Exception Handling in python		
10.	24/09/25	Use the Matplotlib Module for		
		Plotting in Python		
11.	08/10/25	Use the Tkinter Module for UI		
		design		
12.	08/10/25	Simulate gaming concepts Using		
		pygame		

15

15/10/25

R. DMV



## Task-12: Simulate Gaming Concepts using Py game:

Aim:-

To simulate basic concepts using pygame module by converting a simple interactive game, where a player object can move with arrow keys, an enemy move automatically, and collision detection ends the game.

Algorithm:-

1. Import and initialize pygame:

Import the pygame module and initialize it using `pygame.init()`

2. Create a player and enemy object using rectangles.

3. Move player using arrow keys.

4. Move enemy automatically.

5. If player collides with enemy show "Game over" and stop the game.

Program:-

```
key = pygame.key.get_pressed()
```

```
if not game-over:
```

```
    if keys [pygame.K-LEFT] and player.x > 0;
```

```
        player.x -= 5
```

```
    if keys [pygame.K - RIGHT] and player.x < 370;
```

```
        player.x += 5
```

```
    enemy.x += 4
```

```
    if enemy.x < -30;
```

```
        enemy.x = 400
```

```
    if player.colliderect(enemy):
```

```
        game-over = True
```

```
win.fill ((255, 255, 255))
```

```
pygame.draw.rect (win(0, 0, 255), Player)
```

```
pygame.draw.rect (win, (255, 0, 0), enemy)
```

```
if game-over:
```

```
    text = font.render ("Game-over", True, (0, 0, 0))
```

```
    win.blit (text, (130, 130))
```



Input:

Use ← and → arrow keys to move.

Output:

- A window with a moving blue player square.
- A red enemy square moves toward the player.
- On collision: "Game over" is shown.

```
def move_player(x, y):
    if key == 'a':
        x -= 1
    elif key == 'd':
        x += 1
    elif key == 'w':
        y += 1
    elif key == 's':
        y -= 1
    return x, y

def move_enemy(x, y):
    x += 1
    return x, y

def collision(player_x, player_y, enemy_x, enemy_y):
    if player_x == enemy_x and player_y == enemy_y:
        return True
    return False

def game_over():
    screen.fill((255, 255, 255))
    text = font.render("Game over", True, (255, 0, 0))
    screen.blit(text, (200, 200))
    pygame.display.flip()
    pygame.time.wait(2000)
    pygame.quit()
```

pygame. display. update()

clock. tick(30)

VEL TECH-CSE	
EX NO.	12
PERFORMANCE (5)	5
RESULT AND ANALYSIS (5)	5
VIVA VOCE (5)	5
RECORD (5)	5
TOTAL (20)	25
DATE WITH DATE	

Result:- Thus, the simulate gaming concepts using pygame is executed successfully.