

Pong Pygame

Ενότητα 1: Δημιουργία του ενός παίκτη (ρακέτας)

Στόχος:

- Να ανοίγει ένα παράθυρο Pygame.
- Να εμφανίζεται μια ρακέτα (ένα ορθογώνιο) στη μία πλευρά της οθόνης.

```
import pygame
import sys
pygame.init()

WIDTH, HEIGHT = 800, 600
screen = pygame.display.set_mode((WIDTH, HEIGHT))
pygame.display.set_caption("Pong")

BLACK = (0, 0, 0)
WHITE = (255, 255, 255)

PADDLE_WIDTH, PADDLE_HEIGHT = 10, 100
paddle_x = 50
paddle_y = HEIGHT // 2 - PADDLE_HEIGHT // 2

running = True
while running:
    for event in pygame.event.get():
        if event.type == pygame.QUIT:
            running = False

    screen.fill(BLACK)

    pygame.draw.rect(screen, WHITE, (paddle_x, paddle_y, PADDLE_WIDTH,
PADDLE_HEIGHT))
    pygame.display.flip()

pygame.quit()
sys.exit()
```

Τι κάνει ο κώδικας:

- Ανοίγει ένα παράθυρο μεγέθους 800x600.
- Ορίζει ένα ορθογώνιο (την “ρακέτα”) στο αριστερό μέρος της οθόνης.
- Εμφανίζει συνεχώς τη ρακέτα μέχρι να κλείσουμε το παράθυρο.

Ενότητα 2: Κίνηση της ρακέτας με το πληκτρολόγιο

```
keys = pygame.key.get_pressed()
if keys[pygame.K_w]:
    paddle_y -= paddle_speed

    if paddle_y < 0:
        paddle_y = 0
    if paddle_y > HEIGHT - PADDLE_HEIGHT:
        paddle_y = HEIGHT - PADDLE_HEIGHT
```

Τι παρατηρείτε;

```
clock.tick(FPS)
```

Προσθέστε την κίνηση προς τα κάτω καθώς και έναν 2^o παίκτη!

Ενότητα 3: Προσθήκη της μπάλας

```
# Ball settings
BALL_SIZE = 15
ball_x = WIDTH // 2
ball_y = HEIGHT // 2
ball_speed_x = 4
ball_speed_y = 4
```

```
# Ball movement
ball_y += ball_speed_y

if ball_y <= 0 or ball_y >= HEIGHT - BALL_SIZE:
    ball_speed_y *= -1
pygame.draw.circle(.....)
```

Συγκρουση!

1. Δημιουργία Rect για ρακέτα και μπάλα:

```
paddle_rect = pygame.Rect(paddle_x, paddle_y, PADDLE_WIDTH, PADDLE_HEIGHT)
ball_rect = pygame.Rect(ball_x - BALL_SIZE // 2, ball_y - BALL_SIZE // 2,
BALL_SIZE, BALL_SIZE)
```

2. Έλεγχος σύγκρουσης:

```
if paddle_rect.colliderect(ball_rect):
    ball_speed_x *= -1 # Αντιστροφή κατεύθυνσης της μπάλας
    ball_x = paddle_x + PADDLE_WIDTH + BALL_SIZE // 2 # Μετακίνηση έξω
από τη ρακέτα για να μην κολλήσει
```

3. Αντικατάσταση σχεδίασης ρακέτας με χρήση Rect:

```
pygame.draw.rect(screen, WHITE, paddle_rect)
pygame.draw.circle(screen, WHITE, (ball_x, ball_y), BALL_SIZE // 2)
```

4. Αφαιρεση αριστερών ορίων

```
# if ball_x <= 0 or ball_x >= WIDTH - BALL_SIZE:
#     ball_speed_x *= -1
```

Βαθμολογία

```
left_score = 0
right_score = 0
font = pygame.font.Font(None, 74) # Default font, size 74

if ball_x < 0:
    right_score += 1
    print(f"Score: {left_score} - {right_score}")
    ball_x = WIDTH // 2
    ball_y = HEIGHT // 2
    ball_speed_x *= -1
    continue

if ball_x > WIDTH:
    left_score += 1
    print(f"Score: {left_score} - {right_score}")
    ball_x = WIDTH // 2
    ball_y = HEIGHT // 2
    ball_speed_x *= -1
    continue

# Draw scores
score_text = font.render(f"{left_score} - {right_score}", True, WHITE)
text_rect = score_text.get_rect(center=(WIDTH // 2, 50))
screen.blit(score_text, text_rect)
```