PyGame Pong Tutorial - Part 6

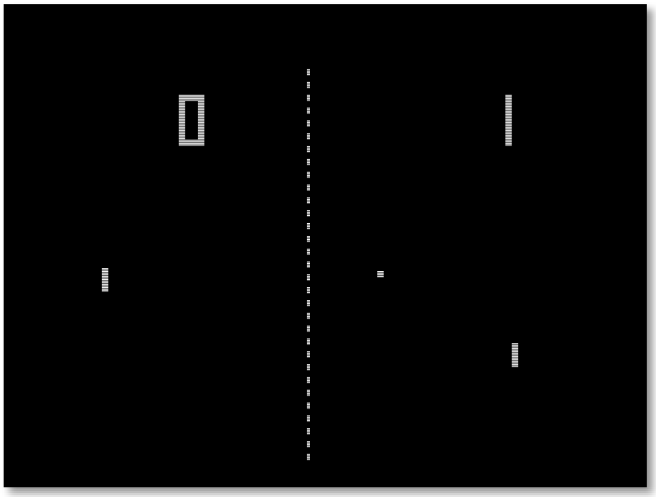
Contents

Time required: 30 minutes

Preview of the Game

Atari. - the year: 1973 - the date: - November 29th - The game is Pong.

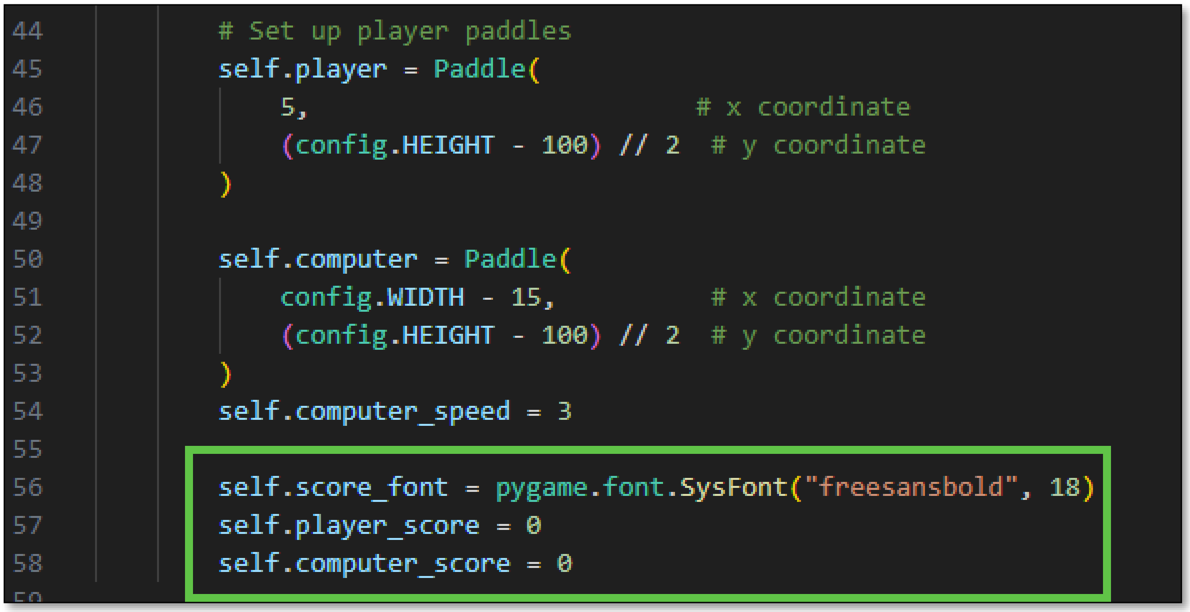
Pong Demo Video



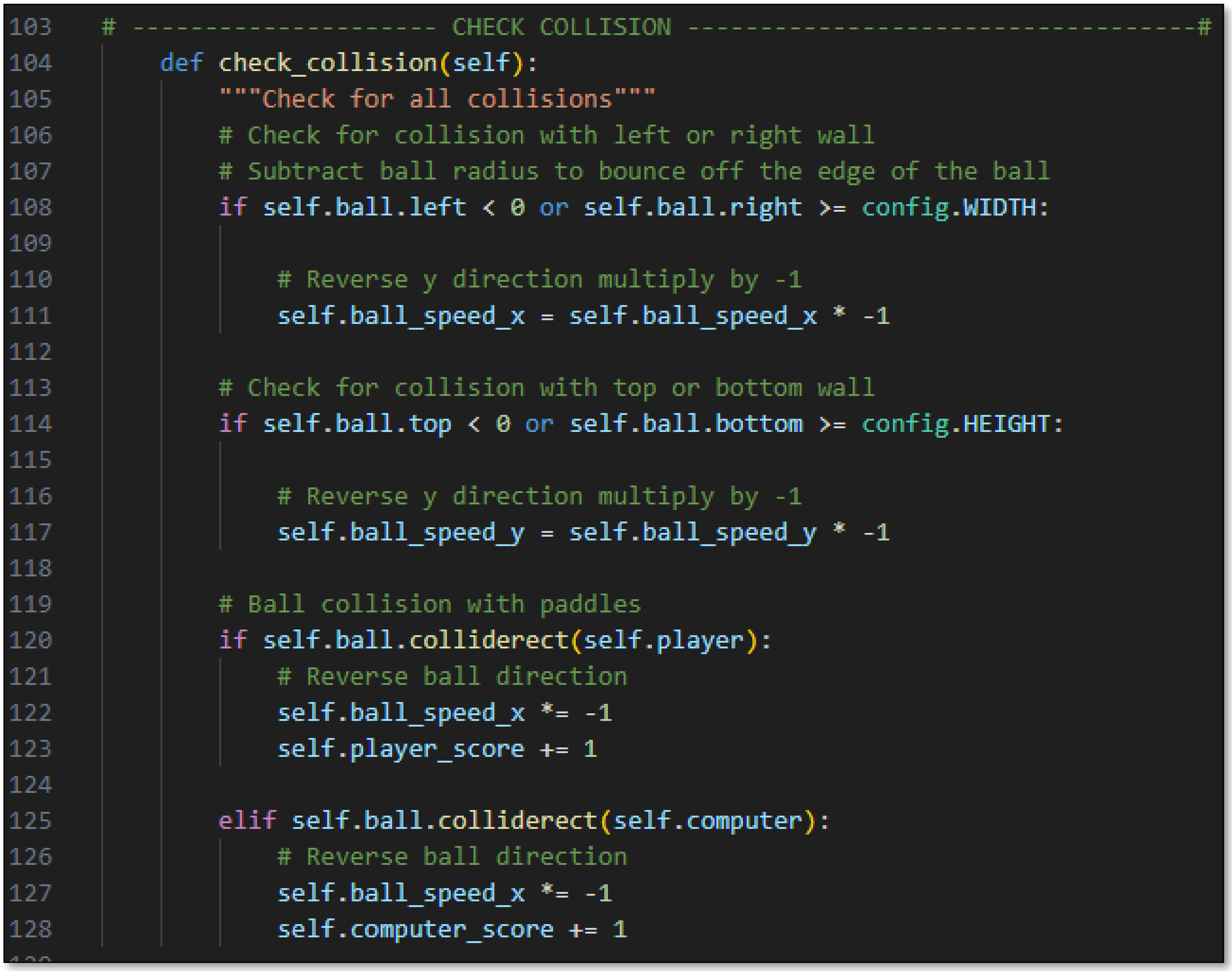
Scoring Time

Taking names . . . . keeping score.

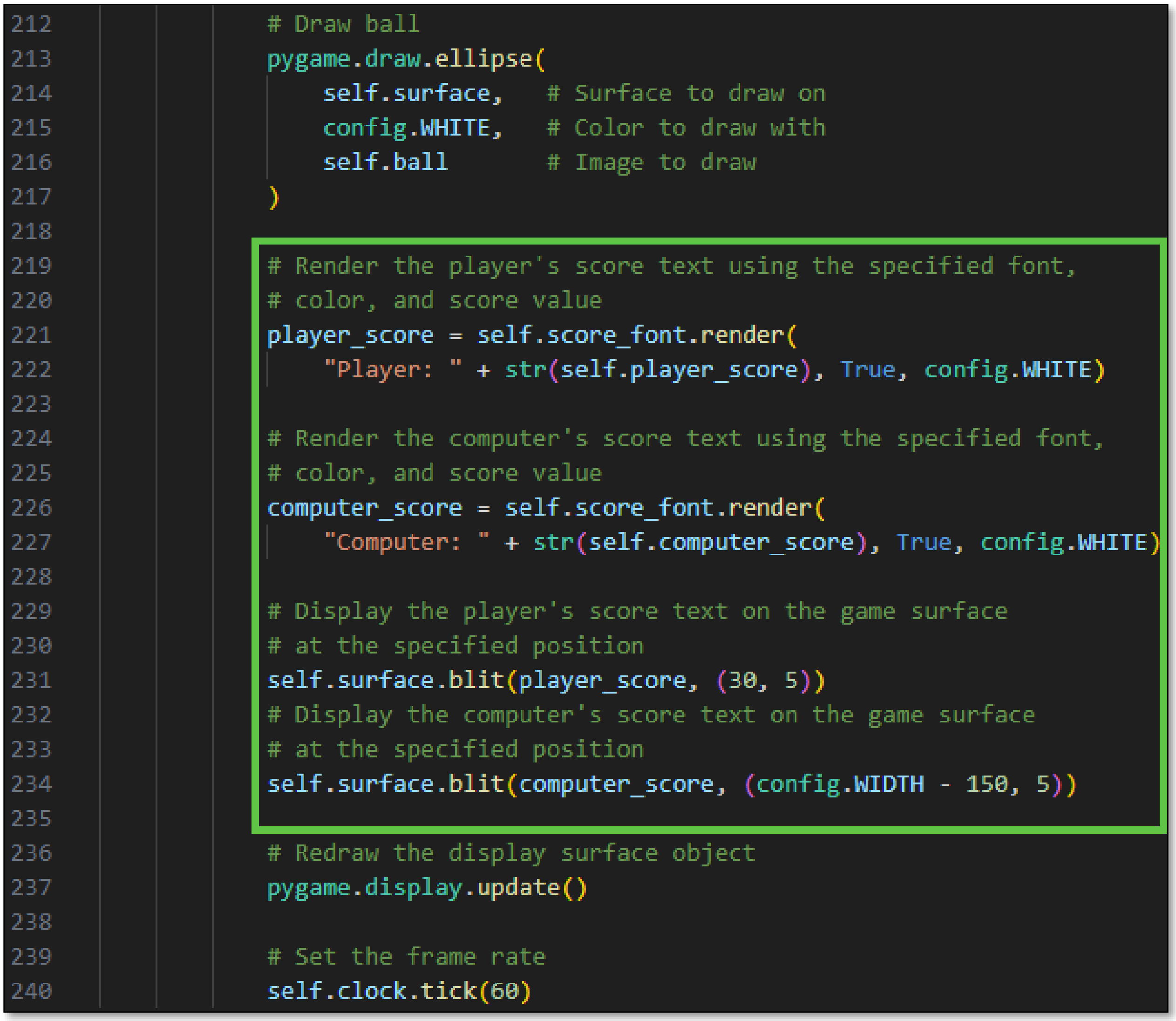
1. Save **pong_5.py** as **pong_6.py**
2. Add the following code to setup the score font and player score tracking.



```
        # Set up player paddles
        self.player = Paddle(
            5,                         # x coordinate
            (config.HEIGHT - 100) // 2  # y coordinate
        )

        self.computer = Paddle(
            config.WIDTH - 15,          # x coordinate
            (config.HEIGHT - 100) // 2  # y coordinate
        )
        self.computer_speed = 3

        self.score_font = pygame.font.SysFont("freesansbold", 18)
        self.player_score = 0
        self.computer_score = 0
```

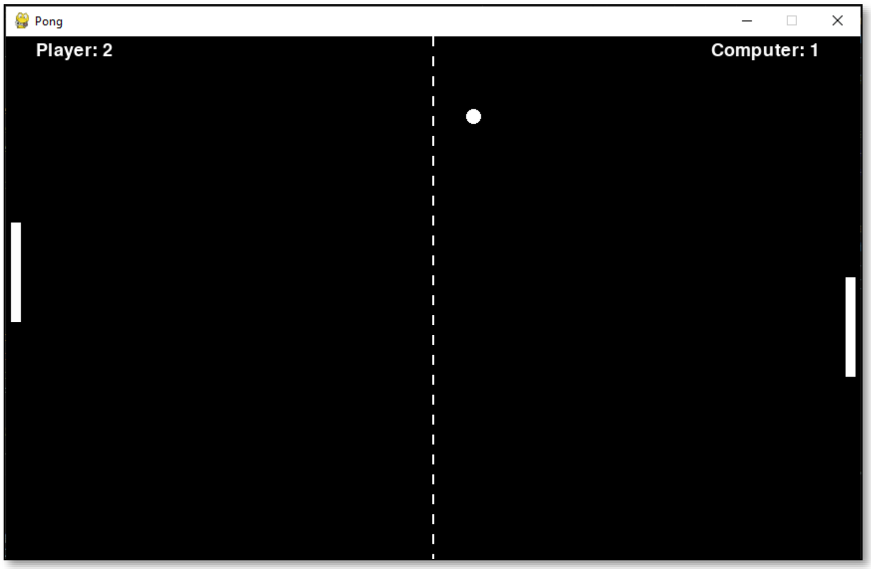


```
# --------------------- CHECK COLLISION -----------------------------------#
    def check_collision(self):
        """Check for all collisions"""
        # Check for collision with left or right wall
        # Subtract ball radius to bounce off the edge of the ball
        if self.ball.left < 0 or self.ball.right >= config.WIDTH:

            # Reverse y direction multiply by -1
            self.ball_speed_x = self.ball_speed_x * -1

        # Check for collision with top or bottom wall
        if self.ball.top < 0 or self.ball.bottom >= config.HEIGHT:

            # Reverse y direction multiply by -1
            self.ball_speed_y = self.ball_speed_y * -1

        # Ball collision with paddles
        if self.ball.colliderect(self.player):
            # Reverse ball direction
            self.ball_speed_x *= -1
            self.player_score += 1

        elif self.ball.colliderect(self.computer):
            # Reverse ball direction
            self.ball_speed_x *= -1
            self.computer_score += 1
```



```python
                # Draw ball
                pygame.draw.ellipse(
                    self.surface,   # Surface to draw on
                    config.WHITE,   # Color to draw with
                    self.ball       # Image to draw
                )

                # Render the player's score text using the specified font,
                # color, and score value
                player_score = self.score_font.render(
                    "Player: " + str(self.player_score), True, config.WHITE)

                # Render the computer's score text using the specified font,
                # color, and score value
                computer_score = self.score_font.render(
                    "Computer: " + str(self.computer_score), True, config.WHITE)

                # Display the player's score text on the game surface
                # at the specified position
                self.surface.blit(player_score, (30, 5))
                # Display the computer's score text on the game surface
                # at the specified position
                self.surface.blit(computer_score, (config.WIDTH - 150, 5))

                # Redraw the display surface object
                pygame.display.update()

                # Set the frame rate
                self.clock.tick(60)
```

Example run:



The game works! We need some sound effects and a game over menu.

Coming up next.

Assignment Submission

Zip up the program files folder and submit in Blackboard.