

## Paddle Ball Game Code Documentation

### Overview

This documentation provides an explanation of the JavaScript code used to create "Paddle Ball", a simple Pong game. The code follows Microsoft .NET Standards and is structured into two main sections: the "Game Code" section for the game logic and the "Start and End Menu Code" section for handling start and end menus.

### Game Code

#### Constants

- **canvas**: A reference to the HTML **canvas** element used for drawing graphics in the game.
- **ctx**: The 2D rendering context of the **canvas**.
- **paddleWidth**: The width of the player's and AI's paddles.
- **paddleHeight**: The height of the player's and AI's paddles.
- **ballSize**: The radius of the ball.
- **startMenu**, **endMenu**, **startButton**, **playAgainButton**, **finalPlayerScore**, **finalAiPlayerScore**: HTML elements used for displaying prompts and scores.
- **playerY** and **aiY**: Initial positions of the player and AI paddles.
- **playerSpeed**: The speed at which the player's paddle moves.
- **ballX** and **ballY**: Initial position of the ball.
- **ballSpeedX** and **ballSpeedY**: Initial speed of the ball in the X and Y directions.
- **aiPaddleOffset**: The distance from the center of the AI's paddle where the ball hits.
- **upPressed** and **downPressed**: Boolean flags for keyboard controls.
- **playerScore** and **aiScore**: Initial player and AI scores.
- **winScore**: The score required to win the game.
- **gameLoopId**: A variable to hold the ID of the game loop interval.

#### Functions

- **randInt(min, max)**: Generates a random integer between **min** and **max**.
- **drawRect(x, y, width, height, color)**: Draws a rectangle on the canvas with the specified **x** and **y** coordinates, **width**, **height**, and **color**.
- **drawCircle(x, y, radius, color)**: Draws a circle on the canvas with the specified **x** and **y** coordinates, **radius**, and **color**.

- **drawText(text, x, y, color, size):** Draws text on the canvas with the specified **text**, **x** and **y** coordinates, **color**, and font **size**.
- **resetBall():** Resets the ball's position and direction.
- **updatePlayerPosition():** Updates the player's paddle position based on keyboard input.
- **gameLoop():** The main game loop responsible for game logic, including rendering paddles, the ball, scores, and handling collisions. It also controls player and AI movement and checks for a win condition.
- Event listeners are used for keyboard controls and AI movement.

### Start and End Menu Code

- The start and end menus are initially hidden (**display: none**) using CSS.
- **startButton** and **playAgainButton** have click event listeners to handle starting a new game and playing again after a game has ended.
- **showEndPrompt():** Displays the end menu, updates the final scores, and hides the game canvas.
- The game loop is started initially to display the start prompt.
- **updatePlayerPosition()** is called to enable keyboard controls for the player.

### How to Play

To start a new game, click the "Start" button on the start menu. The game will begin, and you can control the player's paddle using the 'W' and 'S' keys or the 'Up' and 'Down' arrow keys. The AI controls its paddle autonomously, and the game continues until one player reaches the winning score. After the game ends, you can click the "Play Again" button to start a new game.