

# Documentação do Código HTML

## Sumário

1. [Introdução](#)
2. [Estrutura do Documento](#)
3. [Elementos HTML](#)
4. [Referências Externas](#)

## Introdução

Este documento HTML representa a estrutura básica de um jogo de ping pong. A interface do jogo inclui áreas para jogadores, a bola, e a pontuação. Além disso, há botões para iniciar e reiniciar o jogo. O arquivo style.css é usado para o estilo e o arquivo script.js para a lógica do jogo.

## Estrutura do Documento

O documento é estruturado utilizando tags HTML fundamentais como html, head, e body. A seguir, detalhamos cada parte do documento.

## Elementos HTML

### <html lang="en">

Define a língua padrão do documento como inglês.

### <head>

Contém metadados sobre o documento:

- <meta charset="UTF-8">: Define a codificação de caracteres como UTF-8.
- <meta name="viewport" content="width=device-width, initial-scale=1.0">: Configura a viewport para permitir um design responsivo.
- <title>ping pong</title>: Define o título do documento que aparece na aba do navegador.
- <link rel="stylesheet" href="style.css">: Link para o arquivo CSS externo que estiliza a página.

### <body>

Contém o conteúdo principal da página.

### <div id="game">

Div principal que contém todos os elementos do jogo.

### <div id="players">

Div que contém os elementos representando os jogadores e a bola.

- <div id="player-1"></div>: Representa o jogador 1.
- <div id="player-2"></div>: Representa o jogador 2.
- <div id="ball"></div>: Representa a bola do jogo.

### <div id="score">

Div que contém os elementos relacionados à pontuação e aos controles do jogo.

- <div id="options">
  - <button id="start">iniciar</button>: Botão para iniciar o jogo.

- `<button id="reset">reiniciar</button>`: Botão para reiniciar o jogo.
- `<div id="scores">`
  - `<div id="score-1"></div>`: Exibe a pontuação do jogador 1.
  - `<div id="score-2"></div>`: Exibe a pontuação do jogador 2.

`<script src="script.js"></script>`

Inclui o arquivo JavaScript externo que contém a lógica do jogo.

## Referências Externas

- style.css: Arquivo de estilo CSS para a página.
- script.js: Arquivo de script JavaScript para a lógica do jogo.

Esta documentação fornece uma visão geral da estrutura e dos elementos do documento HTML. Para entender o funcionamento completo, consulte também os arquivos CSS e JavaScript referenciados.

## Documentação do Código JavaScript

### Sumário

1. [Introdução](#)
2. [Elementos HTML](#)
3. [Variáveis Globais](#)
4. [Funções do Jogo](#)
5. [Controle de Eventos](#)
6. [Inicialização](#)

### Introdução

Este documento JavaScript define a lógica de um jogo de ping pong, incluindo a movimentação dos jogadores, da bola, e a atualização das pontuações. Ele interage com elementos HTML e CSS para criar uma experiência de jogo interativa.

### Elementos HTML

javascript

Copiar código

```
// Obtém os elementos do jogo
```

```
const player1 = document.getElementById('player-1');
```

```
const player2 = document.getElementById('player-2');
```

```
const ball = document.getElementById('ball');
```

```
const score1 = document.getElementById('score-1');
```

```
const score2 = document.getElementById('score-2');
```

- player1, player2: Representam os elementos HTML dos jogadores.
- ball: Representa o elemento HTML da bola.
- score1, score2: Representam os elementos HTML das pontuações dos jogadores.

## Variáveis Globais

javascript

Copiar código

```
// Inicializa os scores
```

```
let player1Score = 0;
```

```
let player2Score = 0;
```

```
// Define posições e velocidades iniciais
```

```
let player1Y = 200;
```

```
let player2Y = 200;
```

```
let ballX = 400;
```

```
let ballY = 200;
```

```
let ballXSpeed = 5;
```

```
let ballYSpeed = 5;
```

- player1Score, player2Score: Guardam as pontuações dos jogadores.
- player1Y, player2Y: Guardam as posições verticais dos jogadores.
- ballX, ballY: Guardam as posições da bola.
- ballXSpeed, ballYSpeed: Guardam as velocidades da bola.

## Funções do Jogo

### updatepaddles()

javascript

Copiar código

```
function updatepaddles() {
```

```
    player1.style.top = player1Y + 'px';
```

```
    player2.style.top = player2Y + 'px';
```

```
}
```

Atualiza as posições dos jogadores na tela.

### updateball()

javascript

Copiar código

```
function updateball() {
```

```
    ball.style.left = ballX + 'px';
```

```
    ball.style.top = ballY + 'px';
```

```
}
```

Atualiza a posição da bola na tela.

## **updategame()**

javascript

Copiar código

```
function updategame() {  
    updatepaddles();  
    updateball();  
  
    // Verifica colisões e atualiza velocidade da bola  
    if (ballX < 0 || ballX > 1000) {  
        ballXSpeed = -ballXSpeed;  
    }  
    if (ballY < 0 || ballY > 400) {  
        ballYSpeed = -ballYSpeed;  
    }  
    if (ballY + 10 > player1Y && ballY < player1Y + 100 && ballX < player1.offsetLeft + player1.offsetWidth) {  
        ballXSpeed = -ballXSpeed;  
    }  
    if (ballY + 10 > player2Y && ballY < player2Y + 100 && ballX > player2.offsetLeft - 10) {  
        ballXSpeed = -ballXSpeed;  
    }  
  
    // Atualiza a posição da bola  
    ballX += ballXSpeed;  
    ballY += ballYSpeed;  
  
    // Checa se um jogador marcou  
    if (ballX < 0) {  
        player2Score++;  
        score2.textContent = player2Score;  
        resetGame();  
    } else if (ballX > 1000) {  
        player1Score++;  
        score1.textContent = player1Score;  
    }  
}
```

```
    resetGame();  
}  
  
// Solicita o próximo frame  
requestAnimationFrame(updategame);  
}
```

Atualiza o estado do jogo, incluindo a movimentação dos jogadores e da bola, e verifica colisões.

### **resetGame()**

javascript

Copiar código

```
function resetGame() {  
    player1Y = 200;  
    player2Y = 200;  
    ballX = 400;  
    ballY = 200;  
}
```

Reseta as posições dos jogadores e da bola após um ponto ser marcado.

## **Controle de Eventos**

### **Movimentação dos Jogadores**

javascript

Copiar código

```
document.addEventListener('keydown', function(event) {  
    if (event.key === 'w') {  
        player1Y -= 20;  
    } else if (event.key === 's') {  
        player1Y += 20;  
    }  
});
```

```
document.addEventListener('keydown', function(event) {  
    if (event.key === 'ArrowUp') {  
        player2Y -= 20;  
    } else if (event.key === 'ArrowDown') {  
        player2Y += 20;  
    }  
});
```

```
}  
});
```

Adiciona eventos de pressionamento de teclas para mover os jogadores.

### Inicialização

#### Começar o Jogo

javascript

Copiar código

```
function começar() {  
    document.getElementById('start').addEventListener('click', () => {  
        requestAnimationFrame(updategame);  
    });  
}  
  
começar();
```

Inicia o jogo quando o botão "iniciar" é clicado.

#### Reiniciar o Jogo

javascript

Copiar código

```
function reiniciar() {  
    document.getElementById('reset').addEventListener('click', () => {  
        window.location.reload();  
    });  
}  
  
reiniciar();
```

Reinicia a página quando o botão "reiniciar" é clicado, resetando o jogo.