

Explicação do Jogo Mario Jump - HTML, CSS e JavaScript

1. index.html (estrutura)

```
<!DOCTYPE html> - Inicia o HTML5  
<html lang="pt-br"> - Define idioma (como um comentário no código-fonte)  
<head> - Cabeçalho invisível (como imports no início de Python/Java)  
<meta charset="UTF-8"> - Suporte a acentos (UTF-8 em arquivos .py ou .java)  
<meta name="viewport"...> - Deixa o site responsivo  
<link rel="stylesheet"...> - Como importar uma classe externa  
<link rel="icon"...> - Ícone na aba  
<title>Mario Jump</title> - Nome da janela  
<body> - Início do corpo visível  
<div class="game-board"> - Caixa principal do jogo (como um JPanel)  
<img ...> - Adiciona imagens (como ImageIcon em Java)  
<script src="script.js"> - Conecta lógica do jogo (como importar um script.py)
```

2. style.css (visual)

```
* { margin: 0; padding: 0; } - Remove margens (como limpar variáveis antes de usar)  
width: 100% - largura total (como setWidth em Java)  
height: 500px - altura fixa  
position: relative/absolute - controla posicionamento (como layout managers)  
animation - define movimento com tempo (como timers em Java Swing)  
@keyframes - cria animação quadro a quadro (como Thread que muda posição)  
background: linear-gradient(...) - fundo com gradiente (como definir cor no fundo de um JFrame)
```

3. script.js (funcionamento)

Explicação do Jogo Mario Jump - HTML, CSS e JavaScript

```
const mario = document.querySelector(".mario");

// Em Python: mario = get_elemento("mario")

// Em Java: Element mario = getElementByClass("mario");


const jump = () => { ... }

// Em Python: def jump():

// Em Java: void jump() { ... }


setTimeout(..., 500)

// Em Python: time.sleep(0.5)

// Em Java: Thread.sleep(500);


setInterval(() => { ... }, 10);

// Em Python: while True: ... time.sleep(0.01)

// Em Java: while(true) { Thread.sleep(10); }


getComputedStyle(mario).bottom

// Em Python: estilo = elemento.estilo["bottom"]

// Em Java: getComputedStyle(mario).getBottom()


addEventListener("keydown", jump);

// Em Python com pygame: if event.key == ...

// Em Java: addKeyListener(new KeyAdapter() { public void keyPressed(...) { ... } });



```

4. Conceitos importantes

width - largura (como setSize em Java ou atributo de tamanho em pygame)

height - altura

position - tipo de posicionamento (como usar absolute layout)

Explicação do Jogo Mario Jump - HTML, CSS e JavaScript

animation - movimento automático (como usar Timer ou Thread)

querySelector - busca elemento (como findViewById ou getElement em Selenium)

classList.add/remove - como adicionar/remover classe CSS (similar a adicionar/remover estilo em JavaFX)