



# Como configurar o Player

Danilo Dionisia



# Como configurar o Player

Crie uma pasta chamada player dentro da pasta htdocs do Xampp

Dentro dessa pasta crie um arquivo no Notepad++ chamado index.html

```
<!DOCTYPE html>
<html lang="pt-br">
  <head>
    <meta charset="utf-8">
    <title>Meu primeiro Jogo</title>
    <script src="phaser.js"></script>
    <script src="player.js"></script>
  </head>
  <body>
  </body>
</html>
```

# Como configurar o Player

Lembre-se de copiar o arquivo Phaser.js, que você já utilizou em outras aulas, para a pasta player do Xampp

Agora vamos criar o arquivo que vai personalizar nosso jogo, ou seja, o arquivo que vai criar e configurar o player.

Vá até o Notepad++ e crie o arquivo player.js

```
var game = new Phaser.Game(  
800, 600, Phaser.AUTO, '',  
{  
    preload: preload,  
    create: create,  
    update: update  
});  
  
//Cria as variáveis globais do jogo  
var player;  
var platforms;  
  
//Função que faz o pre carregamento dos sprites  
function preload() {  
  
}  
  
//função que cria os objetos do jogo  
function create() {  
  
}
```

# Como configurar o Player

Agora escreva o código do método preload

//Função que faz o pre carregamento dos sprites

```
function preload() {  
    game.load.image('sky', 'assets/sky.png');  
    game.load.image('ground', 'assets/platform.png');  
    game.load.spritesheet('jogador', 'assets/dude.png', 32, 48);  
}
```

# Como configurar o Player

Comece à escrever o código do método create

```
//habilita a física do jogo
game.physics.startSystem(Phaser.Physics.ARCADE);
//adiciona o sprite do background do jogo
game.add.sprite(0, 0, 'sky');

//cria um grupo para trabalhar com o chão do jogo
platforms = game.add.group();
//habilita o corpo do chao
platforms.enableBody = true;
```



# Como configurar o Player

Ainda dentro do método create

```
//cria a variavel que recebe o grupo do chão
var ground = platforms.create(0, game.world.height - 64, 'ground');
//cria a escala do chão no jogo
ground.scale.setTo(2, 2);
//torna o corpo do chão imóvel
ground.body.immovable = true;

//adiciona o sprite do player e ajusta a altura inicial do player
player = game.add.sprite(32, game.world.height - 500, 'jogador');
//habilita a física do player
game.physics.arcade.enable(player);
```

# Como configurar o Player

Finalizando o código dentro do create

```
//ajusta como o player irá quicar quando cai
player.body.bounce.y = 0.1;
//ajusta a gravidade do player
player.body.gravity.y = 600;
//cria limites para o player não ultrapassar as laterais do jogo
player.body.collideWorldBounds = true;

//cria as animações do player para a direita e esquerda
player.animations.add('left', [0, 1, 2, 3], 10, true);
player.animations.add('right', [5, 6, 7, 8], 10, true);

//captura o teclado para movimentação do player
cursors = game.input.keyboard.createCursorKeys();
```



# Como configurar o Player

Agora vamos desenvolver o código para o método update

```
//cria a variavei que detecta a colisao entre o player e o chao  
var hitPlatform = game.physics.arcade.collide(player, platforms);  
//ajusta a velocidade inicial do player  
player.body.velocity.x = 0;
```

```
if (cursors.left.isDown){  
    //verifica se a tecla para a esquerda foi pressionada  
    player.body.velocity.x = -350;  
    player.animations.play('left');  
}  
else if (cursors.right.isDown){  
    //verifica se a tecla para a direita foi pressionada  
    player.body.velocity.x = 350;  
    player.animations.play('right');  
}  
else{  
    //verifica se o player está parado  
    player.animations.stop();  
    player.frame = 4;  
}  
  
if (cursors.up.isDown && player.body.touching.down && hitPlatform)  
{  
    //verifica se o player está em contato com o chão e ajusta o pulo  
    player.body.velocity.y = -350;  
}
```

# Como configurar o Player

Finalizando o código do update

```
if (cursors.up.isDown && player.body.touching.down && hitPlatform)
{
    //verifica se o player está em contato com o chão e ajusta o pulo
    player.body.velocity.y = -350;
}
```

# Como configurar o Player

Antes de testar, lembre-se de alguns pontos importantes:

- Coloque o script phaser.js dentro da pasta player;
- Crie uma pasta chamada assets dentro da player;
- Insira todas as imagens que você vai utilizar dentro da pasta assets;
- Ajuste o tamanho do sprite do player para que a imagem não seja cortada;
- Para acessar seu jogo lembre-se de “startar” o Xampp;
- Abra o navegador e digite: localhost/player