

Administração Central
Cetec Capacitações

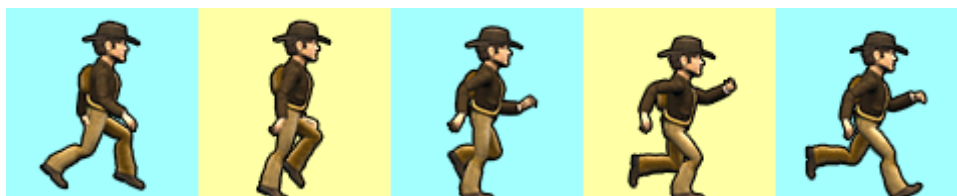
JS

Aula 13 – Animação com Core JS

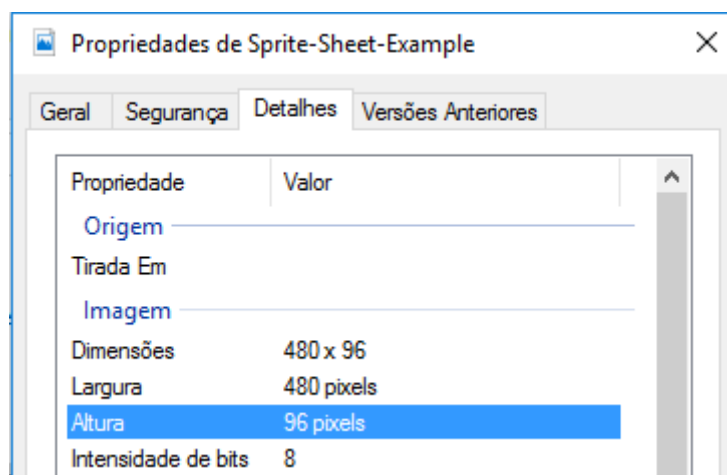
1. Animações

Retomando o uso do **Core.js**, vamos criar um “robot” de animação.

Antes de iniciarmos, vamos verificar a estrutura de um frame de imagens (**spritebatch**)



Vamos analisar as propriedades desta imagem.



A imagem tem **480 x 96**, composta de cinco quadros (**frames**). Desta forma cada quadro (frame) tem exatamente 96 x 96 (**480/5**). Tenha a certeza que todas as imagens fiquem organizadas de forma coerente nos quadros.

Administração Central Cetec Capacitações

Compondo o código CSS:

```
<style type="text/css">
#robo {
    height: 96px;
    width: 96px;
    background-image: url(sua imagem.png);
    background-repeat: no-repeat;
    background-position: 0 0;
}
</style>
```

Na função JavaScript:

```
robo.div = document.getElementById("robo");
    - Recupera o div que compõe a impressão do robô;
robo.frameWidth = 96;
    - Determina a largura da imagem;
robo.frames = 5;
    - Quantidade de quadros (frames) – número de imagens;
robo.offsetX = 0;
    - Determina o valor de X, pois a imagem está na horizontal;
```

Utilize **robo.frameHeight** e **robo.offsetY**, quando a sua imagem estiver disposta na vertical.

```
robo.animar();
    - Executa a função animar();
```

Ok, tudo isso dentro da função **init()**, conforme estrutura:

```
init: function()
{
    robo.div = document.getElementById("robo");
    robo.frameWidth = 96;
    robo.frames = 7;
    robo.offsetX = 0;

    robo.animar();
}
```

Administração Central
Cetec Capacitações

Montando a animação:

robo.offsetX -= robo.frameWidth;

- Determina a posição de X para visualização da imagem;

if (robo.offsetX <= -robo.frameWidth * robo.frames){

robo.offsetX = 0;

}

- Verifica se offsetX não excede o limite da imagem (tamanho x frame), retornando para a posição inicial;

robo.div.style.backgroundColor = robo.offsetX + " 0px";

- Retorna para a **div** a próxima posição da imagem. Lembrando que iniciamos a posição background-position como 0 e 0;

setTimeout(robo.animar, 75);

- Tempo de execução da função animar(), a velocidade (**75**)

Ok, tudo organização na função animar()

```
animar: function(){  
    robo.offsetX -= robo.frameWidth;  
  
    if (robo.offsetX <= -robo.frameWidth * robo.frames){  
        robo.offsetX = 0;  
    }  
    robo.div.style.backgroundColor = robo.offsetX + " 0px";  
    setTimeout(robo.animar, 75);  
}
```

E no HTML?

```
<div id="robo"></div>
```

Não esqueça de adicionar a biblioteca **Core.js** e executá-la **Core.start(robo);**

Verifique os códigos de exemplos, um utiliza imagem na vertical e outro na horizontal.