

## Gabarito - PARTE 1

### Transcrição

Dentro do arquivo `gallows/js/sprite.js` vamos criar a função `createSprite()`. Ela receberá o seletor do elemento que exibirá os frames do nosso sprite. Inclusive, já vamos buscar o elemento e guardar na variável `$el`. Alguns programadores utilizam o prefixo `$` para indicar que a variável guarda um *jQuery Object*, padrão que utilizarei:

```
function createSprite(selector) {  
  
    var $el = $(selector);  
}
```

Utilizaremos o inglês durante a construção do nosso script. Evite misturar inglês com o português dentro de um mesmo script.

Excelente, temos o elemento do DOM encapsulado pelo jQuery que precisaremos iteragir e aplicar cada frame do sprite, alias, vamos aproveitar e criar um array que guardará uma string com o nome de cada frame declarado em `sprite.css`:

```
function createSprite(selector) {  
  
    let $el = $(selector);  
  
    var frames = [  
        'frame1', 'frame2', 'frame3', 'frame4', 'frame5',  
        'frame6', 'frame7', 'frame8', 'frame9'  
    ];  
}
```

É importante sabermos qual é o frame atual, inclusive o último frame, pois não podemos exceder a quantidade de frames do array.

```
function createSprite(selector) {  
  
    var $el = $(selector);  
  
    var frames = [  
        'frame1', 'frame2', 'frame3', 'frame4', 'frame5',  
        'frame6', 'frame7', 'frame8', 'frame9'  
    ];  
  
    var current = 0;  
  
    var last = frames.length - 1;  
}
```

Quando a função `createSprite()` for invocada, logo de cara, precisará adicionar a primeira classe que representa um frame em nosso elemento do DOM. Fazemos isso facilmente através do jQuery com o método `addClass()`:

```
function createSprite(selector) {  
  
    var $el = $(selector);  
  
    var frames = [  
        'frame1', 'frame2', 'frame3', 'frame4', 'frame5',  
        'frame6', 'frame7', 'frame8', 'frame9'  
    ];  
  
    var current = 0;  
  
    var last = frames.length - 1;  
  
    $el.addClass(frames[current]);  
}
```

Agora, vamos definir a função `nextFrame()`, aquela que ao ser chamada mudará o estado de `current` para o próximo frame disponível. Não basta ele adicionar a classe do próximo frame, ele precisa também remover a classe anterior:

```
function createSprite(selector) {  
  
    var $el = $(selector);  
  
    var frames = [  
        'frame1', 'frame2', 'frame3', 'frame4', 'frame5',  
        'frame6', 'frame7', 'frame8', 'frame9'  
    ];  
  
    var current = 0;  
  
    var last = frames.length - 1;  
  
    $el.addClass(frames[current]);  
  
    function nextFrame() {  
  
        $el.removeClass(frames[current])  
            .addClass(frames[++current]);  
    }  
}
```