

≡ 05

## Iterando com jQuery

Temos o seguinte HTML:

```
<ul>
  <li>A</li>
  <li>B</li>
  <li>C</li>
</ul>
```

Qual das opções abaixo seleciona todas as li's com `$('.li')` e itera por cada uma delas imprimindo seu conteúdo no console usando `$.each`?

A

```
$.each(function() {
  var texto = $(this).text();
  console.log(texto);
});
```

B

```
$('.li').each( {
  var texto = $(this).text();
  console.log(texto);
});
```

✓

```
$('.li').each(function() {
  var texto = $(this).text();
  console.log(texto);
});
```

É possível usar o `each` do jQuery para iterarmos em um array padrão do JavaScript:

```
var letras = ['a', 'b', 'c'];

$.each(letras, function() {
  console.log(this);
});
```

No exemplo anterior, passamos dois parâmetro para `$.each`. O primeiro a lista que desejamos iterar e o segundo a função que será chamada para cada item. A cada chamada `this` será `a`, depois, `b` e depois `c`.

Veja que `each` lembra a função `forEach` que todo array possui:

```
letras.forEach(function(letra) {
  console.log(letra);
```

});

Veja que diferente de `$.each`, o tradicional `forEach` recebe como parâmetro o elemento que está iterando e não através de `this` como em `$.each`.

[PRÓXIMA ATIVIDADE](#)