

## Mão na massa: Armazenando a pontuação do usuário

### Começando deste ponto?

Começando deste ponto? Você pode fazer o [DOWNLOAD \(https://s3.amazonaws.com/caelum-online-public/jquery-alura-typewriter/stages/jquery-cap-06.zip\)](https://s3.amazonaws.com/caelum-online-public/jquery-alura-typewriter/stages/jquery-cap-06.zip) completo do projeto do capítulo anterior e continuar seus estudos a partir deste capítulo.

### Salvando a pontuação do usuário

Para salvar a pontuação do usuário no final de cada jogada dele, precisamos primeiramente criar um placar. Para isso, no final da página `principal.html`, vamos adicionar uma tabela:

```
<section class="placar">
  <h3 class="center">Placar</h3>
  <table class="centered bordered">
    <thead>
      <tr>
        <th>Usuário</th>
        <th>No. de palavras</th>
      </tr>
    </thead>
    <tbody>
      <tr>
        <td>Usuario Exemplo</td>
        <td>99</td>
      </tr>
    </tbody>
  </table>
</section>
```

Finalizada a estrutura do placar, podemos partir para o código JavaScript. No `main.js`, vamos criar a função `inserePlacar()` e acessar o nosso placar:

```
function inserePlacar() {
  var placar = $(".placar");
}
```

Mas devemos inserir a pontuação no `tbody`, vamos então obtê-lo. Como `tbody` é um elemento filho do nosso placar, vamos utilizar a função `find` do jQuery, que funciona de modo semelhante a função seletora `$( $ )`, porém fazendo a busca apenas do filho do elemento:

```
function inserePlacar() {
  var placar = $(".placar");
  var corpoTabela = placar.find("tbody");
}
```

Vamos simplificar fazendo tudo em uma linha e como queremos criar uma nova linha na tabela, com o nome do usuário e o número de palavras, vamos pegar essas informações:

```
function inserePlacar(){
  var corpoTabela = $(".placar").find("tbody");
  var usuario = "Seu-nome";
  var numPalavras = $("#contador-palavras").text();
}
```

Precisamos agora inserir uma linha na tabela com esses dados, mas vamos realizar isso em uma função separada:

```
function inserePlacar(){
  var corpoTabela = $(".placar").find("tbody");
  var usuario = "Seu-nome";
  var numPalavras = $("#contador-palavras").text();

  var linha = novaLinha();
}

function novaLinha() {
```

Na função `novaLinha()` precisamos criar um elemento HTML, e para isto vamos utilizar a função jQuery `$( )`, só que ao invés de passarmos um seletor, vamos passar a **tag HTML que queremos criar**:

```
function novaLinha() {
  var linha = $("<tr>");
}
```

Assim a variável `linha` será de fato um `<tr>`, como se estivéssemos acabado de selecionar um no HTML. Vamos montar o resto da tabela com as informações do usuário e o placar, mas como não temos essas informações na função, vamos recebê-las por parâmetro:

```
function inserePlacar(){
  var corpoTabela = $(".placar").find("tbody");
  var usuario = "Seu-nome";
  var numPalavras = $("#contador-palavras").text();

  var linha = novaLinha(usuario, numPalavras);
}

function novaLinha(usuario, numPalavras) {
  var linha = $("<tr>");
  var colunaUsuario = $("<td>").text(usuario);
  var colunaPalavras = $("<td>").text(numPalavras);
}
```

Com tudo pronto, precisamos adicionar os elementos HTML corretamente dentro da linha da tabela, para isso vamos utilizar a função `append`. Ela adiciona a string ou elemento HTML que é passada como parâmetro como último filho do

elemento em qual for chamada:

```
function novaLinha(usuario, palavras) {
  var linha = $("<tr>");
  var colunaUsuario = $("<td>").text(usuario);
  var colunaPalavras = $("<td>").text(palavras);

  // Os dois <td> dentro do <tr>
  linha.append(colunaUsuario);
  linha.append(colunaPalavras);
}
```

E por último, vamos retornar a linha que foi criada pela nossa função:

```
function novaLinha(usuario, palavras) {
  var linha = $("<tr>");
  var colunaUsuario = $("<td>").text(usuario);
  var colunaPalavras = $("<td>").text(palavras);

  // Os dois <td> dentro do <tr>
  linha.append(colunaUsuario);
  linha.append(colunaPalavras);

  return linha;
}
```

Com a linha devidamente criada, vamos adicioná-la no corpo da tabela:

```
function inserePlacar() {
  var corpoTabela = $(".placar").find("tbody");
  var usuario = "Seu nome";
  var numPalavras = $("#contador-palavras").text();

  var linha = novaLinha(usuario, numPalavras);

  corpoTabela.append(linha);
}
```

## Chamando a inserePlacar

A nossa função insere placar deve ser chamada ao fim de cada jogo, e isto ocorre quando tempo chega a zero, ou seja dentro do **if** da função `inicializaCronometro` :

```
function inicializaCronometro() {
  ...
  if (tempoRestante < 1) {
    clearInterval(cronometroID);

    campo.attr("disabled", true);
    campo.toggleClass("campo-desativado");
    // Alteração aqui
  }
}
```

```
        inserePlacar();
    }
    ...
}
```

Mas a função `inicializaCronometro` já está fazendo muitas coisas, vamos separar a responsabilidade de desabilitar o campo, colocar o fundo cinza e chamar o `inserePlacar` em uma outra função, a função `finalizaJogo` :

```
function finalizaJogo() {
    campo.attr("disabled", true);
    campo.toggleClass("campo-desativado");
    inserePlacar();
}
```

E vamos chamar a função `finalizaJogo` dentro do `if` na `inicializaCronometro` substituindo as últimas linhas:

```
function inicializaCronometro() {
    ...
    if (tempoRestante < 1) {
        clearInterval(cronometroID);
        //Alteração aqui
        finalizaJogo();
    }
    ...
}
```

Agora toda ação que deve ser feita ao final de cada jogo pode ser adicionada na função `finalizaJogo` .