

# jQuery UI - Criação de Elementos da Interface

APRESENTAR OS CONCEITOS SOBRE OS RECURSOS DA BIBLIOTECA JQUERY UI, MOSTRANDO SUAS CARACTERÍSTICAS E APLICAÇÃO NO DESENVOLVIMENTO DE INTERFACES DINÂMICAS.

AUTOR(A): PROF. EDSON MELO DE SOUZA

## 1. Introdução

A jQuery é uma biblioteca de funções JavaScript que trabalha diretamente com o HTML, manipulando componentes na página, a fim de proporcionar mais dinamismo. Ela é composta por funções que facilitam o desenvolvimento e a aplicação de recursos, sem a necessidade de muita programação para realizar as tarefas. Seu principal objetivo é otimizar, de maneira fácil, a interpretação dos *scripts* utilizados em uma página.

Neste tópico vamos conhecer a jQuery UI (*Users Interface*), que é um complemento da biblioteca jQuery que contém um conjunto de elementos de tela já prontos para serem usados, como calendários, funções completar automático, botões, entre outros.



## 2. A Biblioteca jQuery UI (*Users Interface*)

jQuery UI é um conjunto de elementos prontos para incrementar a interatividade com o usuário fornecendo, efeitos, *widgets* (componentes da interface gráfica (GUI), incluindo botões, janelas, entre outros) e temas construídos com base na biblioteca jQuery. Esta biblioteca permite criar interfaces altamente interativas, bastando apenas fazer a inclusão de um seletor.

Como se trata de componentes prontos, é importante deixar claro que o uso demasiado pode ser muito prejudicial ao desempenho de uma página, pois os componentes da biblioteca possuem muitos arquivos e são “pesados”, o que impacta diretamente no desempenho durante o carregamento de uma página.

Quando você vai desenvolver um site, as páginas devem seguir diversos princípios de usabilidade e acessibilidade e, portanto, o conjunto de elementos deve ser harmônico, pois um site com páginas totalmente diferentes causa a sensação para o usuário que, ao trocar de página, ele não está no mesmo site. E, neste sentido, ao usar componentes jQuery UI, eles devem seguir o mesmo sistema de cores e formatos desenvolvidos para se adequar ao padrão da sua interface.

## 2.1 Obtendo a jQuery UI

O *download* da biblioteca por ser realizado no endereço <http://jqueryui.com/download/> (<http://jqueryui.com/download/>). Ao acessar a página é necessário selecionar os componentes desejados, pois, como dito anteriormente, há diversas opções. Como você ainda não conhece ainda os componentes disponíveis, faça o *download* de “tudo”, estude e veja qual melhor se adequa aos seus projetos. Isso pode se repetir para projetos diferentes, pois, cada caso é um caso. Para realizar o download, siga os passos seguintes:

### Passo 1 – Versões

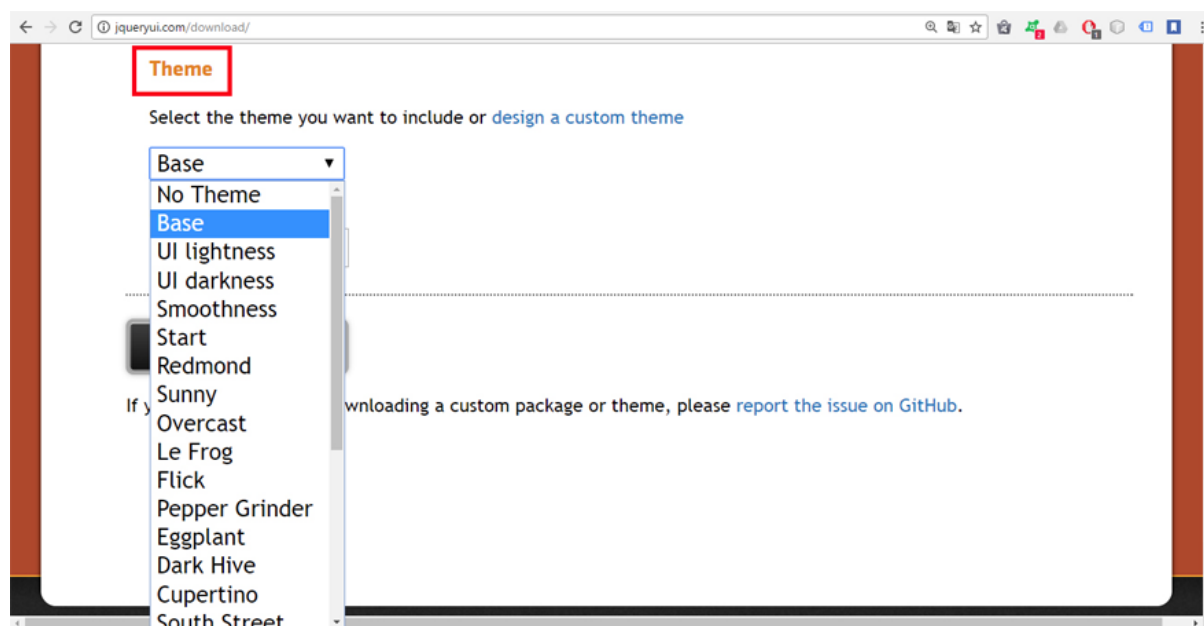
A jQuery possui quatro versões atualmente (maio/2017 – quando este tópico foi escrito), sendo que uma é a versão estável e três são versões anteriores. Dê preferência a utilizar a versão estável, pois é a mais atual.

### Passo 2 - Componentes

Selecione todos os componentes, marcando as *checkbox*. Após selecionar os componentes, será necessário escolher ainda o tema e a versão.

### Passo 3 - Temas

Os temas são os elementos que darão o aspecto visual aos componentes da jQuery UI na página, ou seja, um mesmo componente pode ter aspectos completamente diferentes de acordo com o tema selecionado. A própria equipe que mantém a biblioteca já disponibiliza diversos temas, mas você ainda pode criar seus próprios temas e, para isso, é disponibilizada uma ferramenta para customização dos temas. Na figura a seguir é mostrada a lista *DropDown* com as opções dos temas disponíveis, que está localizada no final da página de *download*.



Para nossos exemplos, vamos utilizar o tema “*UI lightness*”. Após selecionar o tema, clique no botão de *download* para baixar os arquivos. Após a finalização do *download*, descompacte os arquivos em uma pasta de sua escolha.

## 2.2 Incluindo a jQuery UI em Projetos

Para utilizar os componentes da jQuery UI, devemos incluir os seguintes arquivos em nossa página:

1. `<link rel="stylesheet" href="jquery-ui-1.12.1/jquery-ui.min.css">`
2. `<script src="jquery-ui-1.12.1/external/jquery/jquery.js"></script>`
3. `<script src="jquery-ui-1.12.1/jquery-ui.min.js"></script>`

O arquivo padrão para os exemplos deste tópico está listado a seguir:

```
1. <!DOCTYPE html>
2. <html>
3.     <head>
4.         <title>jQuery UI</title>
5.         <meta charset="UTF-8">
6.         <meta name="viewport" content="width=device-width, initial-scale=1">
7.
8.         <!-- inclusão dos arquivos necessários para o funcionamento da jQuery UI -->
9.         <link rel="stylesheet" href="jquery-ui-1.12.1/jquery-ui.min.css">
10.        <script src="jquery-ui-1.12.1/external/jquery/jquery.js"></script>
11.        <script src="jquery-ui-1.12.1/jquery-ui.min.js"></script>
12.
13.        <!-- Aqui deverão ser colocados os script jQuery -->
14.        <script type="text/javascript">
15.
16.        </script>
17.    </head>
18.    <body>
19.
20.        <!-- Aqui deverão ser colocados os elementos HTML -->
21.
22.    </body>
23. </html>
```

## 3. Utilizando os Componentes da jQuery UI

Assim como a jQuery, os componentes da jQuery UI são executados realizando o carregamento por meio da função jQuery “\$()”, podendo estar presentes tanto no “*head*” da página, como diretamente no “*body*”. Os componentes possuem um comportamento padrão, ou seja, já possuem uma configuração inicial, mas também podem ser personalizados, bastando alterar suas propriedades. Cada componente tem seus atributos particulares e, para saber quais são e como alterá-los, é necessário acessar a documentação do componente.

A seguir veremos alguns componentes desta biblioteca:

### 3.1 DatePicker (Calendário)

Em algumas páginas necessitamos que o usuário informe uma data e, para isso, criamos um campo específico. Entretanto, o usuário pode digitar incorretamente a data, o que causará problemas na hora de realizar a gravação em um bando de dados, por exemplo.

Esse componente é vinculado a um campo de texto “*input*” e, ao receber o foco para preenchimento, o componente, ou seja, um calendário é mostrado, permitindo que o usuário selecione com o mouse a data desejada. A seguir é mostrado o código de como inserir este componente em uma página:

```
1. <!DOCTYPE html>
2. <html>
3.     <head>
4.         <title>jQuery UI</title>
5.         <meta charset="UTF-8">
6.         <meta name="viewport" content="width=device-width, initial-scale=1">
7.
8.         <!-- inclusão dos arquivos necessários para o funcionamento da jQuery -->
9.         <link rel="stylesheet" href="jquery-ui-1.12.1/jquery-ui.min.css">
10.        <script src="jquery-ui-1.12.1/external/jquery/jquery.js"></script>
11.        <script src="jquery-ui-1.12.1/jquery-ui.min.js"></script>
12.
13.        <!-- Aqui deverão ser colocados os script jQuery -->
14.        <script type="text/javascript">
15.            $(document).ready(function () {
16.                $("#data").datepicker();
17.            });
18.        </script>
19.    </head>
20.    <body>
21.        <!-- Aqui deverão ser colocados os elementos HTML -->
22.        Clique na caixa para selecionar a data<br>
23.        <input type="text" name="data" id="data">
24.
25.    </body>
26. </html>
```

Nas linhas de 15 a 17 é criado o script jQuery para disponibilizar o componente na página. Na linha 23 é criado um campo texto com o id denominado “data”, mesmo nome atribuído ao seletor na jQuery na linha 16.

Ao executar o código no navegador, é apresentado um campo para preenchimento e, ao clicar nele, aparece o calendário para selecionarmos a data. Por padrão, o calendário já é aberto com a data atual (em inglês), mas podemos configurar para que abra com uma data específica. Veja a seguir a imagem com o resultado após clicar na no campo de texto:



A seguir é apresentada uma modificação no código que transforma o calendário para a língua portuguesa.

```

1. <!DOCTYPE html>
2. <html>
3.     <head>
4.         <title>jQuery UI</title>
5.         <meta charset="UTF-8">
6.         <meta name="viewport" content="width=device-width, initial-scale=1">
7.
8.         <!-- inclusão dos arquivos necessários para o funcionamento da jQuery UI -->
9.         <link rel="stylesheet" href="jquery-ui-1.12.1/jquery-ui.min.css">
10.        <script src="jquery-ui-1.12.1/external/jquery/jquery.js"></script>
11.        <script src="jquery-ui-1.12.1/jquery-ui.min.js"></script>
12.
13.        <!-- Aqui deverão ser colocados os scripts jQuery -->
14.        <script type="text/javascript">
15.            $(document).ready(function () {
16.                $.datepicker.regional['pt-BR'] = {
17.                    closeText: 'Fechar',
18.                    prevText: '<#x3c;Anterior',
19.                    nextText: 'Próximo>#x3e;',
20.                    currentText: 'Hoje',
21.                    monthNames: [
22.                        'Janeiro', 'Fevereiro', 'Março',
23.                        'Abril', 'Maio', 'Junho', 'Julho', 'Agosto',
24.                        'Setembro', 'Outubro', 'Novembro', 'Dezembro'
25.                    ],
26.
27.                    monthNamesShort: [
28.                        'Jan', 'Fev', 'Mar', 'Abr', 'Mai', 'Jun',
29.                        'Jul', 'Ago', 'Set', 'Out', 'Nov', 'Dez'
30.                    ],
31.
32.                    dayNames: [
33.                        'Domingo', 'Segunda-feira', 'Terça-feira',
34.                        'Quarta-feira', 'Quinta-feira', 'Sexta-feira',
35.                        'Sabado'
36.                    ],
37.
38.                    dayNamesShort: [
39.                        'Dom', 'Seg', 'Ter', 'Qua', 'Qui', 'Sex'

```

```
40.         ],
41.
42.         dayNamesMin: [
43.             'Dom', 'Seg', 'Ter', 'Qua', 'Qui', 'Sex'
44.         ],
45.
46.         weekHeader: 'Sm',
47.
48.         dateFormat: 'dd/mm/yy',
49.
50.         firstDay: 0,
51.
52.         isRTL: false,
53.
54.         showMonthAfterYear: false,
55.
56.         yearSuffix: ''};
57.
58.         $.datepicker.setDefaults($.datepicker.regional['pt-BR'])
59.
60.         $("#data").datepicker();
61.     });
62. </script>
63. </head>
64. <body>
65.     <!-- Aqui deverão ser colocados os elementos HTML -->
66.     Clique na caixa para selecionar a data<br>
67.     <input type="text" name="data" id="data">
68.
69. </body>
70. </html>
```

Na linha 18 o componente é preparado para aceitar a língua portuguesa. Na linha 19 inicia-se a configuração do componente com as traduções que se estendem até a linha 36. Perceba que foram utilizados caracteres especiais para evitar erros na apresentação. Na linha 37 o componente é invocado “chamado” e fica disponível para ser utilizado.

Na imagem a seguir é mostrado o resultado após a renderização do código no navegador.



Clique na caixa para selecionar a data

Maio 2017

Dom	Seg	Ter	Qua	Qui	Sex	Sab
	1	2	3	4	5	6
7	8	9	10	11	12	13
14	15	16	17	18	19	20
21	22	23	24	25	26	27
28	29	30	31			

## 3.2 Accordion

Este componente é muito interessante e útil, pois aceita a criação de caixas onde podemos inserir texto em seu interior. Este componente se comporta mostrando os conteúdos de forma classificadas em que, enquanto uma caixa fica aberta, as demais ficam fechadas.

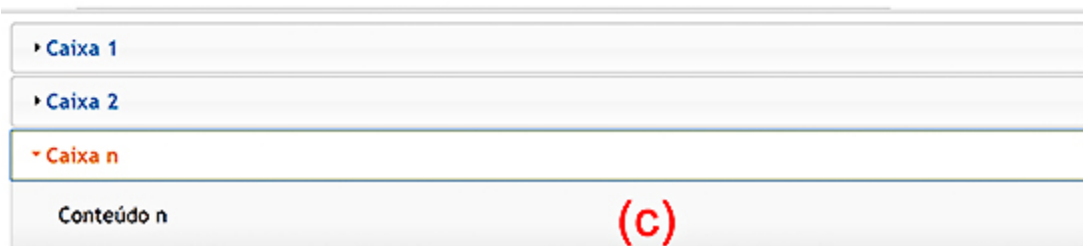
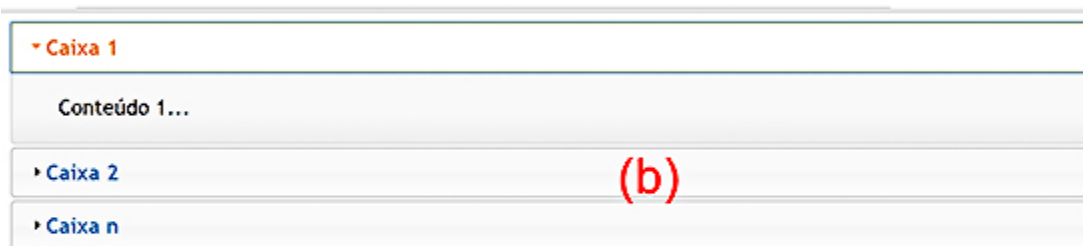
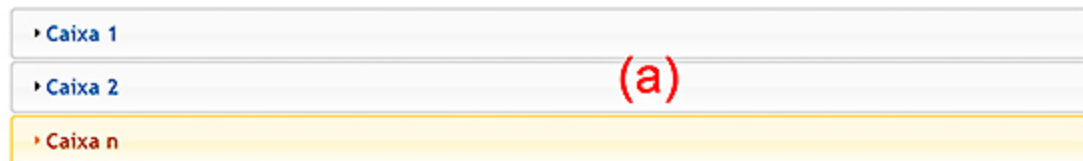
Para criar as caixas, utilizamos uma *div* e, em seu interior, as etiquetas HTML. No exemplo a seguir vamos criar três caixas com textos diferentes para vermos a funcionalidade deste componente.

```
1. <!DOCTYPE html>
2. <html>
3.     <head>
4.         <title>jQuery UI</title>
5.         <meta charset="UTF-8">
6.         <meta name="viewport" content="width=device-width, initial-scale=1">
7.
8.         <!-- inclusão dos arquivos necessários para o funcionamento da j
9.         <link rel="stylesheet" href="jquery-ui-1.12.1/jquery-ui.min.css"
10.        <script src="jquery-ui-1.12.1/external/jquery/jquery.js"></scrip
11.        <script src="jquery-ui-1.12.1/jquery-ui.min.js"></script>
12.
13.        <!-- Aqui deverão ser colocados os script jQuery -->
14.        <script type="text/javascript">
15.            $(document).ready(function () {
16.                $("#accordion").accordion({
17.                    header: 'h3',
18.                    fillSpace: true,
19.                    collapsible: true,
20.                    active: false
21.                });
22.            });
23.        </script>
24.    </head>
25.    <body>
26.        <!-- Div criada para montagem dos painéis -->
27.        <div id="accordion">
28.            <h3><a href="#">Caixa 1</a></h3>
29.            <div>Conteúdo 1...</div>
30.            <h3><a href="#">Caixa 2</a></h3>
31.            <div>Conteúdo 2...</div>
32.            <h3><a href="#">Caixa n</a></h3>
33.            <div>Conteúdo n</div>
34.        </div>
35.    </body>
36. </html>
```

Nas linhas de 16 a 20 configuramos o componente. Na linha 17 é especificado que será usada uma *tag h3* para ser o título da caixa. Na linha 18 é configurada a separação entre as caixas. Na linha 19 é informado que as caixas podem ser inicializadas fechadas e, por fim, na linha 20 é informado que não haverá nenhuma caixa ativa, ou seja, aberta, quando a página carregar.

Nas linhas de 27 a 34 são construídas a *divs* e os elementos *h3* para receberem os textos.

Na imagem a seguir é mostrado o resultado da criação após a renderização, sinalizando três opções (a), (b) e (c).



A imagem acima mostra as três sequências de funcionamento, iniciando com todas as caixas fechadas. Quando clicado na primeira ela é aberta e as demais ficam fechadas. Quando terceira é aberta, a anterior que estava aberta se fecha e assim, sucessivamente.

### 3.3 Tooltip (Dicas Rápidas)

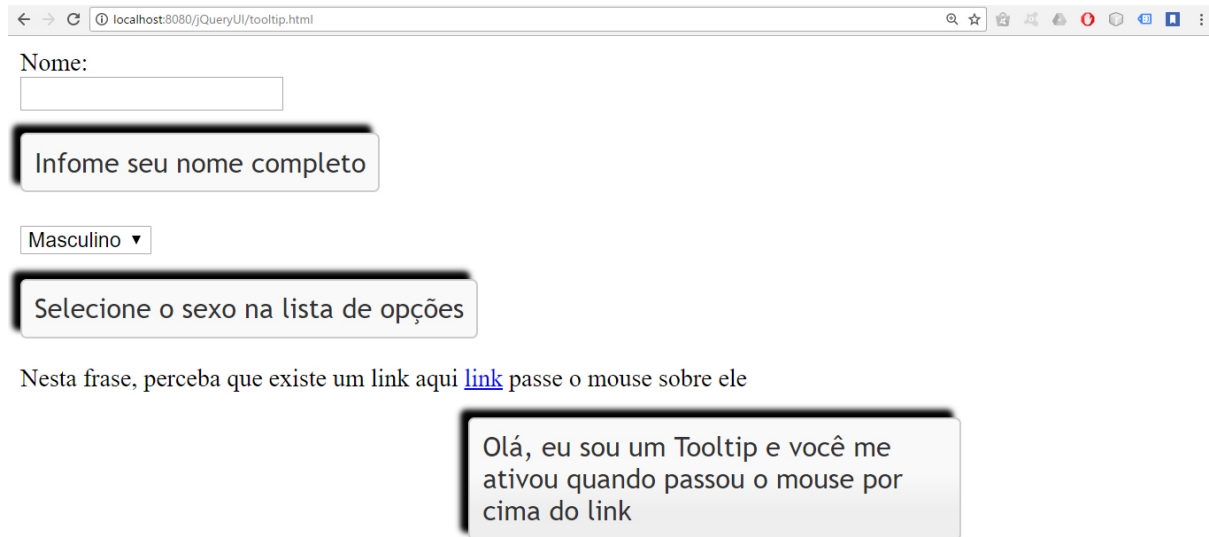
Este recurso é muito útil para explicar um contexto sobre uma palavra quando o mouse for passado sobre uma palavra que tenha o recurso aplicado. É uma explicação adicional que situa melhor o usuário, sendo utilizado juntamente com o cursor do mouse.

Este componente pode ser utilizado em vários elementos HTML como links, caixa de texto, listas *DropDown*, entre outros. Vejamos a seguir o código necessário para implementar essa funcionalidade usando o jQuery UI.

```
1. <!DOCTYPE html>
2. <html>
3.     <head>
4.         <title>jQuery UI</title>
5.         <meta charset="UTF-8">
6.         <meta name="viewport" content="width=device-width, initial-scale=1">
7.
8.         <!-- inclusão dos arquivos necessários para o funcionamento da j
9.         <link rel="stylesheet" href="jquery-ui-1.12.1/jquery-ui.min.css"
10.        <script src="jquery-ui-1.12.1/external/jquery/jquery.js"></scrip
11.        <script src="jquery-ui-1.12.1/jquery-ui.min.js"></script>
12.
13.        <!-- Aqui deverão ser colocados os script jQuery -->
14.        <script type="text/javascript">
15.            $(document).ready(function () {
16.                $(document).tooltip();
17.            });
18.        </script>
19.    </head>
20.    <body>
21.        <!-- Aqui deverão ser colocados os elementos HTML -->
22.        Nome:<br>
23.        <input type="text" name="nome" title="Informe seu nome completo">
24.        <br><br><br><br><br>
25.
26.        <select name="sexo" title="Selecione o sexo na lista de opções">
27.            <option>Masculino</option>
28.            <option>Feminino</option>
29.        </select>
30.
31.        <br><br><br><br>
32.        <p>
33.            Nesta frase, perceba que existe um link
34.            aqui <a href="#" title="Olá, eu sou um Tooltip e você me ati
35.        </p>
36.    </body>
37. </html>
```

Na linha 16 invocamos o método *tooltip()* da jQuery UI. Na linha 23 definimos um campo texto “*input*” e nas linhas de 27 a 30 definimos uma lista *DropDown* “*select*”. Na linha 33 é inserido um *link* com um texto. O funcionamento do recurso está atribuído à propriedade “*title*” dos elementos HTML, ou seja, qualquer elemento da página que tiver um “*title*” definido, será mostrado seu valor quando o mouse passar por cima dele.

Na imagem a seguir é mostrado o recurso *tooltip* em funcionamento no campo texto, lista *DropDown* e no *link*.



Como você pode perceber, os componentes da jQuery UI podem contribuir em muito para que sua interface ganhe dinamismo. Nos *links* em destaque você poderá ver “todos” os componentes que a jQuery UI disponibiliza, inclusive com orientações de como configurar os parâmetros para personalizar as exibições.

## SAIBA MAIS!

jQuery UI Oficial - <http://jqueryui.com/demos/> (<http://jqueryui.com/demos/>)

Tutorials Point: <http://www.tutorialspoint.com/jqueryui/> (<http://www.tutorialspoint.com/jqueryui/>)

## Resumo

Neste tópico foi apresentada a biblioteca jQuery UI, que traz componentes dinâmicos para serem aplicados em páginas HTML. Também foram apresentadas algumas configurações dos componentes para personalização, dependendo da situação desejada.

## Conclusão

Você deve ter percebido como a utilização da jQuery UI pode melhorar ainda mais a dinâmica das páginas HTML, estendendo as funcionalidades da jQuery. Existem dezenas de componentes que podem ser utilizados na construção das suas páginas. Explore os links de apoio, use a imaginação e a criatividade, mas tome cuidado com os exageros.

## ATIVIDADE FINAL

A jQuery UI pode ser definida como:

- A. Conjunto de elementos prontos para incrementar a interatividade com o usuário.
- B. Uma linguagem de programação.
- C. Um tipo de HTML aperfeiçoado.
- D. Uma forma diferente de trabalhar com a jQuery.

Qual a vantagem em utilizar um *datepicker* em uma página?

- A. Permite que não haja erros na digitação de uma data.
- B. Nenhuma, apenas estética.
- C. Criar um calendário com uma estética sofisticada.
- D. Realizar o carregamento da página de forma mais rápida.

Para utilizar o *tooltip()*, qual propriedade deverá ser definida?

- A. title
- B. div
- C. ready
- D. function

## REFERÊNCIA

BALDUÍNO, Plínio. Dominando JavaScript com jQuery. Editora Casa do Código, 2014.

CHAFFER, Jonathan. Learning JQuery 1.3: Better Interaction and Web Development with Simple JavaScript Techniques. Packt Publishing Ltd, 2009.

DA SILVA, Marcelo Soares; FERRARI, Fabiano Cutigi. Integração de frameworks front-end para desenvolvimento de interfaces ricas com JavaServer Faces. Revista TIS, v. 4, n. 1, 2016.

DO AMARAL, Rodrigo Augusto; DE ALMEIDA NERIS, Vânia Paula. Análise comparativa entre frameworks de frontend para aplicações web ricas visando reaproveitamento do back-end. Revista TIS, v. 4, n. 1, 2016.

DUCKETT, Jon. JavaScript & jQuery. Wiley VCH, 2015.

DUCKETT, Jon. Web Design with HTML, CSS, JavaScript and jQuery Set. Wiley Publishing, 2014.

SILVA, Maurício Samy. jQuery UI. Brasil: Novatec, 2012.

TANAKA, Eduardo Hideki. Design de interfaces web com jQuery. In: Proceedings of the Companion Proceedings of the 10th Brazilian Symposium on Human Factors in Computing Systems and the 5th Latin American Conference on Human-Computer Interaction. Brazilian Computer Society, 2011. p. 29-32.