

JavaScript

RICARDO FROHLICH DA SILVA
FABRÍCIO TONETTO LONDERO

Javascript

JavaScript é uma linguagem de programação principalmente tratada no cliente (client side), ou seja, executada localmente.

Caso se pretenda segurança de dados sensíveis, então deve ser tratado no servidor (server side), ou seja, executada remotamente.

Tem diversos usos, tais como validar formulários, alteração de estilo, inserção dinâmica de conteúdo, comunicar-se com servidor (usando o recurso *ajax*)

Javascript

Pode-se inserir código JavaScript ...

... direto no corpo através do conteúdo da **tag script**

... através de arquivos externos através do atributo src da *tag script*

... via eventos, diretamente nos atributos de evento dos elementos por exemplo: *onclick* e *onblur*

```
1 ▼ <!-- Corpo da tag SCRIPT -->
2 ▼ <script>
3     var teste = "01a";
4 </script>
5
6 ▼ <!-- Atributo src -->
7 ▼ <script src="js/exemplo.js"></script>
8
9 ▼ <!-- Atributos de eventos | -->
10 ▼ <button onclick="alert('01a')">Teste</button>
```

JavaScript no Corpo da Página

Basta escrever o código entre as tags `<script>` e `</script>`

```
<script>  
alert( "Oi!" );  
</script>
```

Javascript

Neste caso, quando o navegador do usuário carregar este trecho de código aparecerá uma janela escrita **Oi!**

Deve-se evitar escrever o código diretamente no corpo da página pois ele não poderá ser reutilizado em outras páginas (assim como o código CSS já visto).

JavaScript em Arquivo Externo

É a maneira preferida pela maioria dos desenvolvedores.

Consiste em criar um novo arquivo com a extensão .js e lá escrever o código.

- Assim, como nas folhas de estilo, pode-se reutilizar o código em diversas páginas sem precisar reescrevê-lo, devendo apenas referenciar o arquivo.

```
<script src="javascript.js">  
</script>
```

Variáveis

Declarar variáveis: algumas linguagens de programação exigem que as variáveis sejam declaradas no começo do código, já o JavaScript não exige isso. O nome da variável é *case sensitive*.

```
var teste = "teste";
```

```
teste = "teste";
```


Variáveis

Existem duas maneiras de se declarar uma variável em JavaScript, com e sem o atributo `var`:

- usando `var` a variável é criada no escopo da função (sendo liberada quando a função encerra).
- sem `var`, a variável é criada no escopo global que no browser é o objeto *window* (sendo liberada quando a página é fechada) - não é boa prática de programação!

Variáveis

Uma variável pode assumir diversos valores primitivos

- Number - 10, 10.5, 1.5e3, 0xff (representado 255)
- String - “teste”, ‘teste’
- Boolean - true ou false
- undefined - quando não existe ou não foi inicializada
- null - permite definir explicitamente que não tem valor
- valores não primitivos (Object, Function, Array)

Variáveis

```
1 var n1 = 10;           // Number
2 var n2 = 10.5;
3 var n3 = 1.5e3;
4 var n4 = 0xff;
5
6 var s1 = "Teste";      // String
7 var s2 = 'Teste';
8
9 var b1 = true;         // Boolean
10
11 var u = undefined;
12 var n = null;
13
14 var obj = {           // Object
15     id: 30,
16     nome: "Fulano"
17 };
18 var arr = [ "lista", "de", "elementos" ]; //Array
19 var func = function(x, y) { return x + y }; // Function
```

Condicionais

```
1  var numero = 1;
2  // Comparação normal (compara apenas os valores)
3  if(numero == "1") { // resultado: true
4      alert("== compara apenas os valores");
5  }
6
7  // Comparação restrita (compara valores e tipos das variáveis)
8  var mensagem;
9  if(numero === "1") { // resultado: false
10     mensagem = "Nunca executa";
11 } else {
12     mensagem = "Number é diferente de String com o ===";
13 }
14
15 /*
16     Operador ternário (versão reduzida do IF)
17     condição ? se_verdadeiro : se_falso ;
18 */
19 var x = (numero === "1") ? "Nunca executa" : "Number é diferente de String com o ===" ;
```

```
1  var laranja = 6;
2  switch(laranja) {
3      case 3:
4          alert('Fazer um copo de suco');
5          break;
6      case 6:
7          alert('Fazer dois copos de suco');
8          break;
9      case 9:
10         alert('Fazer uma jarra de suco');
11         break;
12     default:
13         alert('Fazer quanto suco for possível');
14 }
```

Laços de repetição

```
1 // FOR - executa um laço um determinado numero de vezes
2 var pessoas = ["João", "José", "Maria", "Sebastião", "Antônio"];
3
4 for (var i = 0; i < pessoas.length; i++) {
5     alert(pessoas[i]);
6 }
```

```
1 // FOR IN - executa um laço nas propriedades de um objeto
2 var nome = "";
3 var pessoa = { nome:"Iara", sobrenome:"Almeida"};
4 for (x in pessoa) {
5     nome += pessoa[x] + " ";
6 }
7 alert(nome);
```

Laços de repetição

```
1 var resultado="";
2 var numero = 1;
3 while (numero<=10){
4     resultado += numero + " ";
5     numero++;
6 }
7 alert(resultado);
```

```
1 var resultado="";
2 var numero = 1;
3 do{
4     resultado += numero + " ";
5     numero++;
6 }while (numero<=10);
7 alert(resultado);
```

Exercícios

1. Escreva um código que mostre os números ímpares entre 1 e 10.
2. Escreva um código que calcule a soma de 1 até 100 (PS: a resposta é 5050)

Funções

É um recurso que permite efetuar tarefas repetitivas ou funcionalidades que só devem ser executadas após um evento, como o clique de um botão, o uso da função é a solução.

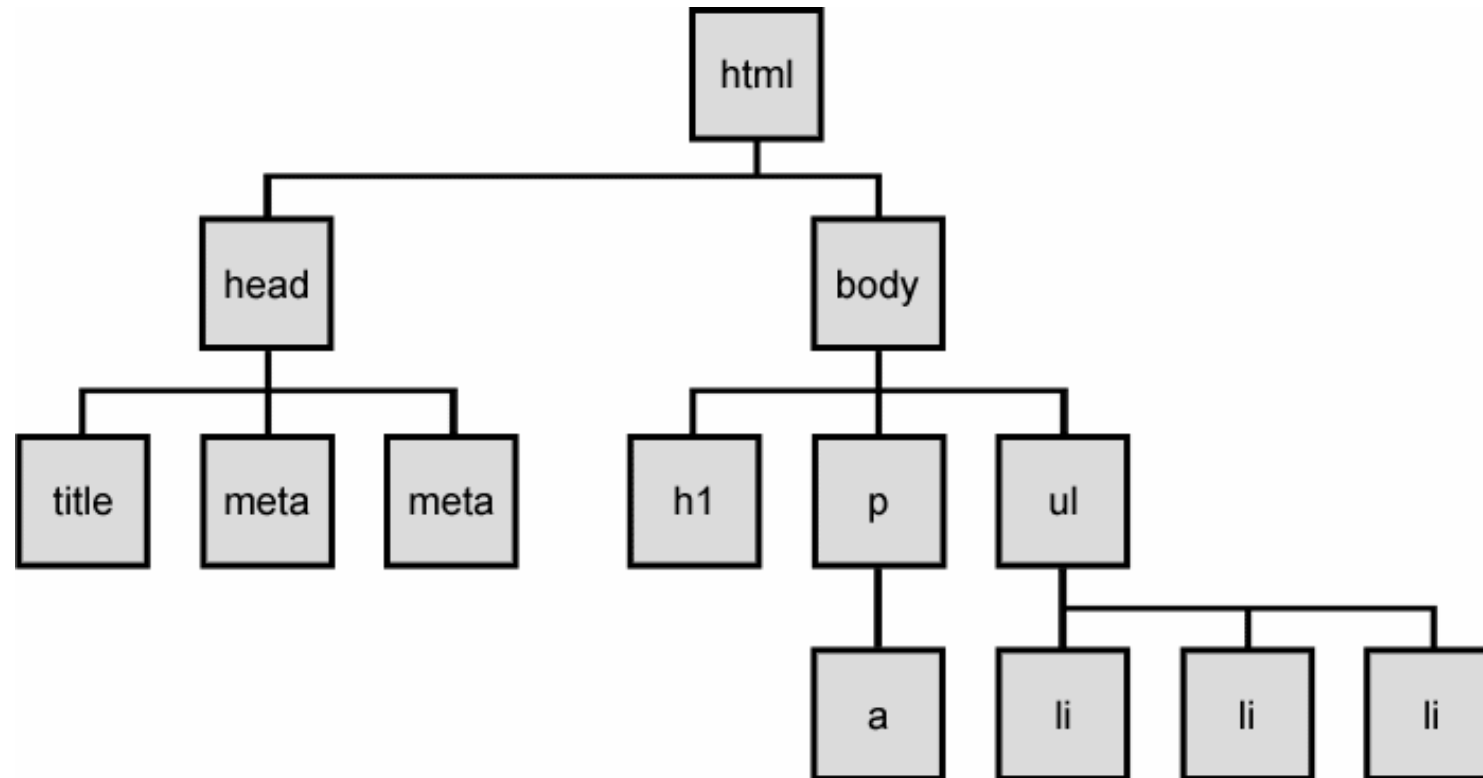
Desafio: “Executar função ao clique do botão”: executar um alerta cada vez que um botão seja acionado até, no máximo, 5 alertas. A cada iteração, apresentar o número do alerta.

DOM

Quando uma página HTML é carregada, o browser cria o DOM da página.

Ou seja, o HTML resultante, o que pode ser visto pressionando Ctrl + U nos browsers!

DOM



Alteração do DOM

Após a página carregada no cliente, existe a possibilidade de alterar o DOM sem efetuar requisições ao servidor.

Para isso, utilizamos Javascript e/ou bibliotecas javascripts, tais como jQuery.

Javascript é executado nativamente pelos browsers.

Eventos

| | |
|--------------------------------------|---------------------------------------------------------------------------|
| <u>onclick</u> | Evento disparado quando se clica com o mouse em algum elemento |
| <u>oncontextmenu</u> | Clique com o direito do mouse |
| <u>ondblclick</u> | Dois cliques do mouse em um elemento |
| <u>onmousedown</u> | Ocorre quando o usuário pressiona o mouse (antes do clique) |
| <u>onmouseenter</u> | Executa quando o mouse entra na área do elemento. |
| <u>onmouseleave</u> | Executa quando o mouse sai da área do elemento. |
| <u>onmousemove</u> | Executado enquanto o usuário esta movendo o mouse em cima de um elemento. |
| <u>onmouseover</u> | Executado quando o mouse entra na área de um elemento ou dos seus filhos. |
| <u>onmouseout</u> | Executado quando o mouse sai da área de um elemento ou de seus filhos. |
| <u>onmouseup</u> | Executado quando o botão do mouse para de ser pressionado. |

Eventos

| | |
|-----------------------------------|---------------------------------------------------------|
| <u>onkeydown</u> | Executado quando o usuário está pressionando uma Tecla. |
| <u>onkeypress</u> | Executado quando o usuário pressiona uma Tecla. |
| <u>onkeyup</u> | Executado quando o usuário solta uma Tecla. |

Eventos

| | |
|-----------------------|------------------------------------------------------------------------------|
| <u>onabort</u> | Executa quando o usuário aborta o carregamento de algum elemento ou recurso. |
| <u>onbeforeunload</u> | Executado antes do documento html ser descarregado |
| <u>onerror</u> | Executado quando ocorre erro no carregar um arquivo externo. |
| <u>onpagehide</u> | Executado quando o usuário “sai” do site. (troca de aba) |
| <u>onscroll</u> | No ato do usuário utilizar a barra de rolagem. |

Listagem completa de eventos

http://www.w3schools.com/jsref/dom_obj_event.asp

Seletores

Para efetuar uma alteração com js, antes precisamos identificar o elemento desejado.

Seletores

```
document.getElementsByTagName("div")
```

```
document.querySelectorAll(".my-class")
```

```
document.getElementsByClassName("my-class")
```

```
document.getElementById("meuID");
```

Exemplo 1

```
<html>
  <body>
    Informe o seu nome:
    <input type="text" id="nome" onfocusout="minhaFuncao()">

    <script>
    function minhaFuncao()
    {
      var x = document.getElementById("nome");
      x.value = x.value.toUpperCase();
    }
    </script>
  </body>
</html>
```

Exemplo 2

```
<html>
  <body>
    Informe o seu nome:
    <input type="text" id="nome" onfocusout="minhaFuncao()">

    <script>
      function minhaFuncao()
      {
        var x = document.getElementById("nome");
        Alert(x.value);
      }
    </script>
  </body>
</html>
```

Exercício 1

Construa uma pagina com dois inputs do tipo Text (um para nome e outro para idade) e um botão. Ao clicar no botão, mostre uma mensagem como esta: “Juarez tem 22 anos”.

Exercício 2

Construa uma pagina com dois inputs do tipo Text (um para nome e outro para idade) e um botão. Ao clicar no botão, mostre uma mensagem como esta: “Juarez é maior de idade”.

Exercício 3

Construa uma pagina com dois inputs do tipo Text e um botão. Ao clicar no botão, mostre o resultado da soma dos valores atribuídos aos inputs.

Exercício 4

Continuando o exercício anterior, mude a cor do botão para azul se o resultado for par e para verde se for ímpar.


```
<html>
  <body>
    v1: <input type="text" id="v1" >
    v2: <input type="text" id="v2" >
    <button id="btn" onclick="minhaFuncao()">Clique!</button>
    <script>
      function minhaFuncao()
      {
        var v1 = parseInt(document.getElementById("v1").value);
        var v2 = parseInt(document.getElementById("v2").value);
        var result = v1 + v2;
        if(result % 2 == 0)
        {
          var btn = document.getElementById("btn");
          btn.style.background='#F00';
        }
        else
        {
          var btn = document.getElementById("btn");
          btn.style.background='#00F';
        }
      }
    </script>
  </body>
</html>
```

Recomendação de leitura

<http://tableless.com.br/tenha-o-dom/>

<http://www.fititnt.org/padrao/seletores.html>

<http://desenvolvimentoparaweb.com/javascript/javascript-jquery-equivalentes-funcoes-nativos/>

<https://www.caelum.com.br/apostila-html-css-javascript/jquery/#12-7-exercicios-jquery-na-home>

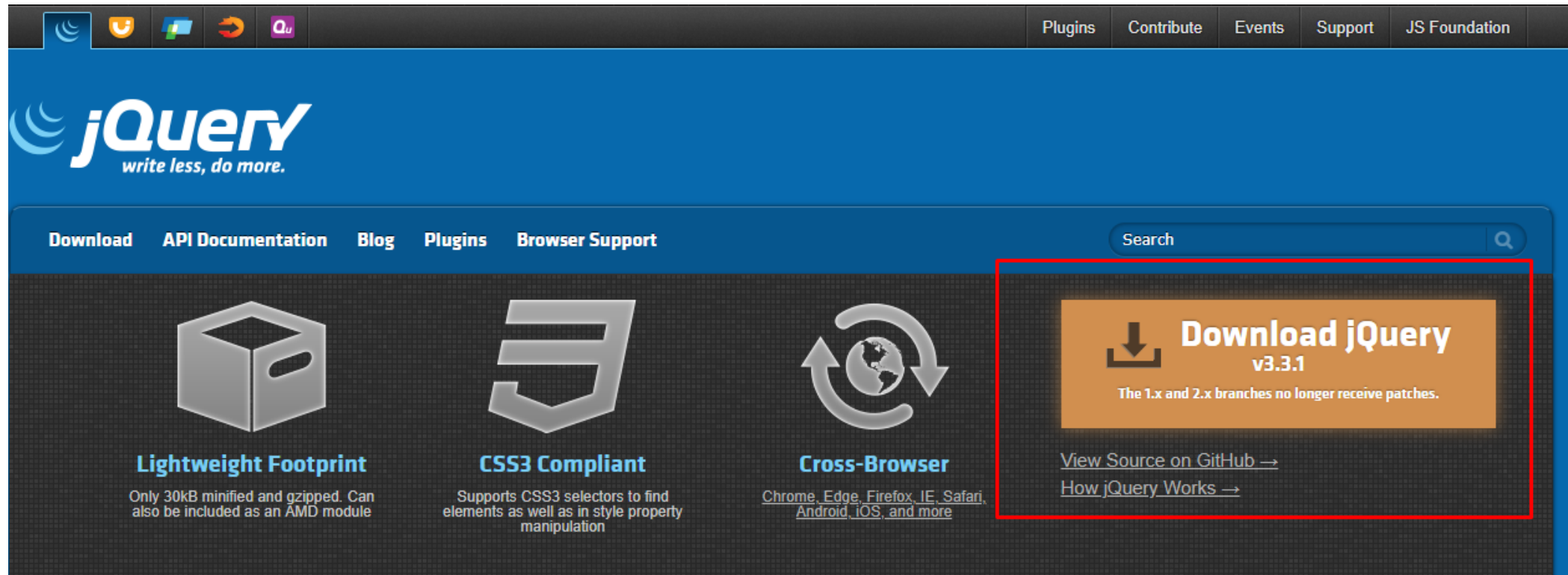
jQuery

O jQuery é uma biblioteca JavaScript rápida, pequena e rica em recursos. Ele torna as coisas como passagem e manipulação de documentos HTML, manipulação de eventos, animação e Ajax muito mais simples, com uma API fácil de usar que funciona em vários navegadores. Com uma combinação de versatilidade e extensibilidade, a jQuery mudou a forma como milhões de pessoas escrevem JavaScript.

jQuery – Como Utilizar

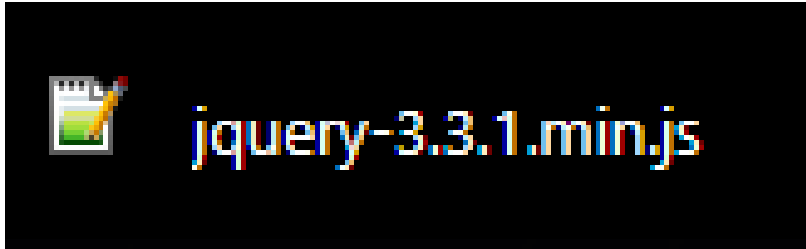
Primeiramente acessamos o site da biblioteca: <https://jquery.com/>

E efetuamos o download:



jQuery – Como Utilizar

Arquivo:



Agora devemos vincula-lo aos nossos arquivos .html

- Adicionando a tag script com o atributo SRC com o diretório do arquivo baixado
- `<script src="jquery-3.3.1.min.js"></script>`
- Isto dentro da tag `<head>` que contem os arquivos e instruções para o bom funcionamento do nosso site

Pronto, jquery já esta pronta para o uso.

jQuery - Exemplo

```
$(document).ready(function(){  
    $("p").click(function(){  
        $(this).hide();  
    });  
});
```

→ esta linha permite que o código interno seja executado apenas quando a página terminar de carregar (o DOM esta pronto)

jQuery - Exemplo

```
$(document).ready(function(){  
    $("p").click(function(){  
        $(this).hide();  
    });  
});
```

→ aqui, para todo e qualquer <p> do nosso HTML (SELETOR), quando for clicado pelo usuário (.click), uma função (function) será disparada.

jQuery - Exemplo

```
$(document).ready(function(){  
    $("p").click(function(){  
        $(this).hide();  
    });  
});
```

→ O Código da função, usa o selector `this` (o que foi selecionado, no caso o `<p>`) e faz a chamada da função `hide()`, responsável por esconder o element `<p>` clicado


```
<html>
  <head>
    <script src="jquery-3.3.1.min.js"></script>
    <script>
      $(document).ready(function() {
        $("p").click(function() {
          $(this).hide();
        });
      });
    </script>
  </head>
  <body>
    <p>
      Lorem ipsum dolor sit amet, consectetur adipiscing elit.
    </p>
  </body>
</html>
```

Sites Úteis

https://www.w3schools.com/jquery/tryit.asp?filename=tryjquery_hide → testar o código apresentado

<https://www.w3schools.com/jquery/default.asp> --> jQuery Tutorial

Exercícios

Escolha 2 exercícios da lista de javascript e refaça-os utilizando jQuery.