



# • Criação de Sites

• Prof. Msc. Christien Lana Rachid

# JavaScript

Podemos incluir as tags de Javascript em vários lugares do código HTML:

- Em um arquivo externo;
- No cabeçalho do documento;
- Dentro de Tags, utilizando tratadores de eventos e
- No corpo do documento.

# JavaScript

No cabeçalho do documento

```
<html>
  <head>
    <meta charset="utf-8">
    <title>JavaScript</title>

    <script type="text/javascript">
      alert("Seja bem-vindo(a)!");
      // JavaScript no Cabeçalho da página HTML.
    </script>
  </head>

  <body>
  </body>
</html>
```

# JavaScript

Em um arquivo externo

```
<html>
  <head>
    <meta charset="utf-8">
    <title>JavaScript</title>

    <script type="text/javascript" src="javascript.js">
      //Chamado para um arquivo externo.
    </script>
  </head>

  <body>
  </body>
</html>
```

# JavaScript

Em uma Tag HTML através de Tratadores de Eventos

```
<html>
  <head>
    <meta charset="utf-8">
    <title>JavaScript</title>
  </head>

  <body>

    <input type="button" value="Mensagem" onclick="alert('Ola!')";>

  </body>
</html>
```

# JavaScript

No corpo de uma página HTML

```
<html>
  <head>
    <meta charset="utf-8">
    <title>JavaScript</title>
  </head>

  <body>
    <script type="text/javascript">
      document.write("Aulas JavaScript");
      //No Corpo de uma página HTML.
    </script>
  </body>
</html>
```

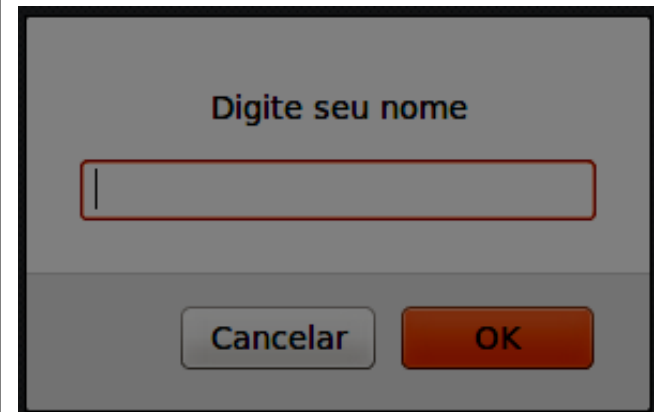
# JavaScript

## Método prompt( )

Abre uma caixa de diálogo para que o usuário possa entrar com dados.

```
<html>
  <head>
    <meta charset="utf-8">
    <title>JavaScript</title>
  </head>

  <body>
    <script type="text/javascript">
      prompt("Digite seu nome");
    </script>
  </body>
</html>
```



# JavaScript

## Método prompt( )

O texto após a vírgula e dentro dos parênteses, será exibido no campo de resposta (resposta padrão), caso prefira deixe em branco, não inclua conteúdo nem espaços entre as aspas.

```
<html>
  <head>
    <meta charset="utf-8">
    <title>JavaScript</title>
  </head>
  <body>
    <script type="test/javascript">
      origem = prompt("Informe a Instituição onde estuda, "CESJF");
    </script>
  </body>
</html>
```



# JavaScript

```
1 <html>
2   <head>
3     <meta charset="utf-8">
4     <title>JavaScript</title>
5     <link rel="stylesheet" type="text/css" href="estilo_exemplo1.css"/>
6   </head>
7   <body>
8     <header>
9       <p id="nome"></p>
10    </header>
11    <script>
12      var usuario = prompt("Informe seu nome", "Nome");
13      document.getElementById("nome").innerHTML = "Olá " + usuario;
14    </script>
15  </body>
16 </html>
```

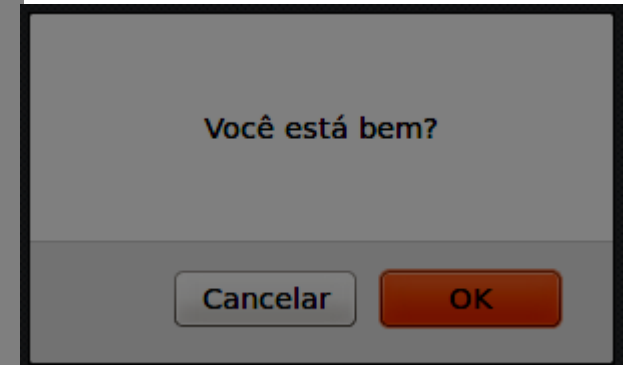
# JavaScript

## Método confirm( )

Exibe uma caixa de confirmação contendo dois botões:  
<ok> (sim) e <cancel> (não).

```
<html>
  <head>
    <meta charset="utf-8">
    <title>JavaScript</title>
  </head>

  <body>
    <script type="text/javascript">
      resp = confirm("Você está bem?");
    </script>
  </body>
</html>
```



# JavaScript

## Método confirm( )

```
1 <html>
2 <head>
3   <meta charset="utf-8">
4   <title>JavaScript</title>
5   <link rel="stylesheet" type="text/css" href="estilo_exemplo1.css"/>
6 </head>
7 <body>
8   <h1>JavaScript</h1>
9
10  <button type="button" onclick="confirm('Confirme a opção! \nEscolha Cancelar ou OK!')">
11    Clique aqui!
12  </button>
13 </body>
14 </html>
```

Quebra de linha

## JavaScript

Clique aqui!

Confirme a opção!  
Escolha Cancelar ou OK

Cancelar

OK

# JavaScript

## Variáveis Globais e Locais:

Para criar variáveis basta dar um nome e atribuir um valor.

**Instituicao** = “CES JF”;

Variáveis também podem ser criadas utilizando a palavra chave **var**.

**var Instituicao** = “CES JF”;

A variável será global se estiver fora de uma função, caso contrário será local, ou seja, só poderá ser acessada dentro da função onde foi criada.

# JavaScript

## Variáveis Globais e Locais:

```
<html>
  <head>
    <meta charset="utf-8">
    <title>JavaScript</title>
  </head>

  <body>
    <script type="text/javascript">
      num = 3858;
      // Criando e atribuindo um valor à uma variável.

      var msg1;
      var msg2; // Criando uma variável por linha.

      var num1, nome1, msg1; // Criando todas as variáveis em uma única linha.

      var nome2 = "João", num2 = "12", msg2 = "telefone do João: 1234-5678";
      // Criando todas as variáveis e atribuindo valores, em uma única linha.

      function cliente(){
        var nome3="João"; // Criando uma variável local dentro de uma função
      }
    </script>
  </body>
</html>
```

# JavaScript

## Nomes de Variáveis

- O primeiro caracter deve ser uma letra ASCII (em maiúscula ou minúscula) ou um caracter de sublinhado (\_). Observe que um número não pode ser usado como o primeiro caracter.
- Os caracteres subsequentes devem ser letras, números ou sublinhados (\_).
- O nome da variável não deve ser uma palavra reservada.

# JavaScript

## Nomes de Variáveis

### Atenção!!!

- Não há limite oficial no comprimento de nomes de variáveis, portanto, vale o bom senso.
- Utilize sempre nomes mnemônicos, ou seja, que traduzam o conteúdo da variável.

# JavaScript

## Nomes de Variáveis

Exemplos de nomes válidos:

- Total\_geral
- nome3
- \_num1
- i (que é diferente de I)



# JavaScript

## Tipos de Dados

Em JavaScript não precisa definir uma variável especificando o tipo de dados que ela armazenará (exceto em casos raros), ela se adaptará ao tipo de dados do conteúdo a ela atribuído.

# JavaScript

## Tipos de Dados: Básicos

- Number - Números Inteiros ou de Ponto Flutuante.
- Ex.: 0; 1; -3; 42; 3,14 ou 3.14.
- String - Trata-se de uma cadeia de caracteres (entre aspas).
- Ex.: “Meu nome é Guilherme”.
- Boolean - Booleanos ou Lógicos; possuem dois valores: Verdadeiro ou Falso.
- Ex.: true; false.

# JavaScript

## Conversões entre tipos de dados Exemplo:

```
total = 40;
```

```
document.write("O total é " + total);
```

**Imprimirá:** "O total é 40".

- **document.write()** trabalha com Strings, portanto, o interpretador automaticamente converte quaisquer não-*strings* na expressão em Strings antes de desempenhar a função.
- A caixa **prompt ()** também trabalha com Strings, portanto, antes de executar qualquer cálculo com valores fornecidos através do prompt, devem ser convertidos do tipo de String para números.

# JavaScript

## Conversões entre tipos de dados

- **Number()** // Converte em um número inteiro, retorna números com ou sem casa decimal.
- **parseInt()** // Converte em um número inteiro.
- **parseFloat()** // Converte em um número com ponto flutuante.

# JavaScript

## Conversões entre tipos de dados

```
1 <html>
2   <head>
3     <meta charset="utf-8">
4     <title>JavaScript</title>
5   </head>
6   <body>
7     <h4>Diferença entre Number(), parseInt() e parseFloat()</h4>
8     <p id="valores1"></p>
9     <p id="valores2"></p>
10    <p id="valores3"></p>
11    <script>
12      document.getElementById("valores1").innerHTML =
13        "Number(true) = " + Number(true) + "<br>" +
14        "Number(false) = " + Number(false) + "<br>" +
15        "Number(5) = " + Number(5) + "<br>" +
16        "Number(5.5) = " + Number(5.5) + "<br>" +
17        "Number(\"5ab\") = " + Number("5ab") + "<br>" +
18        "Number(\"5 6\") = " + Number("5 6") + "<br>";
19
20      document.getElementById("valores2").innerHTML =
21        "parseInt(5) = " + parseInt(5) + "<br>" +
22        "parseInt(5.5) = " + parseInt(5.5) + "<br>" +
23        "parseInt(5ab) = " + parseInt("5ab") + "<br>" +
24        "parseInt(\"5 6\") = " + parseInt("5 6") + "<br>";
25
26      document.getElementById("valores3").innerHTML =
27        "parseFloat(5) = " + parseFloat(5) + "<br>" +
28        "parseFloat(5.5) = " + parseFloat(5.5) + "<br>" +
29        "parseFloat(\"5ab\") = " + parseFloat("5ab") + "<br>" +
30        "parseFloat(\"5 6\") = " + parseFloat("5 6") + "<br>";
31    </script>
32  </body>
33 </html>
34
```

# JavaScript

## Conversões entre tipos de dados

### Diferença entre `Number()`, `parseInt()` e `parseFloat()`

```
Number(true) = 1  
Number(false) = 0  
Number(5) = 5  
Number(5.5) = 5.5  
Number("5ab") = NaN  
Number("5 6") = NaN
```

```
parseInt(5) = 5  
parseInt(5.5) = 5  
parseInt(5ab) = 5  
parseInt("5 6") = 5
```

```
parseFloat(5) = 5  
parseFloat(5.5) = 5.5  
parseFloat("5ab") = 5  
parseFloat("5 6") = 5
```

# JavaScript

## Conversões entre tipos de dados

```
<html>
  <head>
    <meta charset="utf-8">
    <title>JavaScript</title>
  </head>

  <body>
    <script type="text/javascript">
      num_b = parseFloat(prompt("Digite um número",""));
      // num_b será do tipo number.
      alert(num_b + " é do tipo " + typeof(num_b));

      string_a = "2 é um número primo!";
      num_a = parseInt(string_a);
      alert(num_a);
      // A variável num_a armazenará o valor 2,
      //a parte não numérica é ignorada.
    </script>
  </body>
</html>
```

# JavaScript

## Conversões entre tipos de dados - outros

- `String( )` - Converte o tipo de uma variável em String.
- `Boolean( )` - Converte o tipo de uma variável em booleano.
  
- `a = 2;`
- `b = String(a);`
- `alert(typeof(b));`



# JavaScript

Fixando o número de casas “decimais” - **toFixed()** ( )

```
1 <html>
2   <head>
3     <meta charset="utf-8">
4     <title>JavaScript</title>
5   </head>
6   <body>
7     <h4>JavaScript - toFixed()</h4>
8     <script>
9       a = 10/3;
10      document.write(a);
11      a = a.toFixed(2);
12      document.write("<br>" + a);
13    </script>
14  </body>
15 </html>
16
```

## JavaScript - toFixed()

3.3333333333333335  
3.33

# JavaScript

## Operadores - Aritméticos

- **Adição (+)**

```
a = 2 ,    b = 3;  
c = a + b;  
document.write("<br>" + c);           // o valor de "c" será 5
```

- **Subtração (-)**

```
a = 3 ,    b = 1;  
c = a - b;  
document.write("<br>" + c);           // o valor de "c" será 2
```

- **Divisão (/)**

```
a = 8 ,    b = 2;  
c = a / b;  
document.write("<br>" + c);           // o valor de "c" será 4
```

- **Multiplificação (\*)**

```
a = 2 ,    b = 3;  
c = a * b;  
document.write("<br>" + c);           // o valor de "c" será 6
```

# JavaScript

## Operadores - Aritméticos

- **Módulo ou Resto da Divisão(%)**

```
a = 5 , b = 2;
```

```
c = a % b;
```

```
document.write("<br>" + c);
```

*// o valor de "c" será 1*

# JavaScript

## Operadores - Relacionais

- **Menor que (<)**

```
a = 5 , b = 6;  
alert(a < b);
```

- **Menor ou igual a (<=)**

```
a = 3 , b = 5;  
alert(a <= b);
```

- **Maior que (>)**

```
a = 4 , b = 6;  
alert(b > a);
```

- **Maior ou igual a (>=)**

```
a = 5 , b = 7;  
alert(b >= a);
```

- **Igual a (==)**

```
a = 3 , b = 3;  
alert(a == b);
```

- **Não igual a (!=)**

```
a = 3 , b = 4;  
alert(a != b);
```

# JavaScript

## Operadores - Lógicos

- **E (&&)**

```
a = 6 , b = 3, c = 1;
```

```
alert((a > b) && (b > c));
```

*// (a é maior que b) e (b maior que c)*

- **Ou (||)**

```
a = 6 , b = 3, c = 1;
```

```
alert((a > b) || (b == c));
```

*// (a é maior que b) ou (b é igual a c)*

- **Negação (!)**

```
a = 6 , b = 3, c = 1;
```

```
alert(a != b);
```

*// ( a não é igual a b)*

# JavaScript

Incremento de prefixo e sufixo (++)

```
<body>
  <script type="text/javascript">
    a = 5;
    alert(++a + 2);
    // Adiciona 1 à variável "a" antes de executar a expressão.
    // O resultado do alert será 8

    a = 5;
    alert(a++ + 2);
    // Adiciona 1 à variável "a" após a execução da expressão.
    // O resultado do alert será 7
  </script>
</body>
```

# JavaScript

Decremento de prefixo e sufixo (--)

```
<body>
  <script type="text/javascript">
    a = 5;
    alert(--a + 2);
    // Subtrai 1 da variável "a" antes de executar a expressão.
    // O resultado do alert será 6

    a = 5;
    alert(a-- + 2);
    // Subtrai 1 da variável "a" após a execução da expressão.
    // O resultado do alert será 7
  </script>
</body>
```

# JavaScript

## Operadores de Atribuição

- **Atribuição Plena (=)**

`a = 3;`                      *// A Variável "a" recebe o valor 3.*  
`a = b = c = 5;`            *// As variáveis "a", "b" e "c" recebem o valor 5.*

- **Compostos:**

`x =+ 3;`                      *// É o mesmo que `x = x + 3;`*  
`x =- 3;`                      *// É o mesmo que `x = x - 3;`*  
`x =* 3;`                      *// É o mesmo que `x = x * 3;`*  
`x =/ 3;`                      *// É o mesmo que `x = x / 3;`*



# JavaScript

Concatenador de string

```
a = "São" , b = "Paulo";  
c = a + b;
```

O conteúdo de “c” será “São Paulo”.

# JavaScript

## Funções

Funções são grupos de instruções em JavaScript que podem ser tratadas como uma unidade.

São executadas apenas quando o programa as solicita, uma ou mais vezes.

Normalmente definimos as funções dentro do cabeçalho da HTML ou em um arquivo externo e podemos chamá-las em qualquer parte do script.

# JavaScript

## Funções - Exemplo

```
<html>
  <head>
    <meta charset="utf-8">
    <title>JavaScript</title>

    <script type="text/javascript">
      function nome_da_função(){
        sentença1; // Linhas de instruções da função.
        sentença2;
        ...
      }
    </script>

  </head>

  <body>
    <script type="text/javascript">
      nome_da_função();
      //Chamada da função para que ela possa ser executada.
    </script>
  </body>
</html>
```

# JavaScript

## Funções sem Parâmetros

```
<html>
  <head>
    <meta charset="utf-8">
    <title>JavaScript</title>

    <script type="text/javascript">
      function mens(){
        alert("Ola!");
        // Definição da função no cabeçalho da página.
      }
    </script>

  </head>

  <body>
    <script type="text/javascript">
      mens();
      mens();
      mens();
    </script>
  </body>
</html>
```

# JavaScript

## Função recebendo Parâmetros

```
<html>
  <head>
    <meta charset="utf-8">
    <title>JavaScript</title>

    <script type="text/javascript">
      function mens(quem1, quem2){
        alert("Olá " + quem1 + ", Olá " + quem2);
      }
    </script>

  </head>

  <body>
    <script type="text/javascript">
      nome1 = "Carlos";
      nome2 = "Eder";
      mens(nome1, nome2);
    </script>
  </body>
</html>
```

# JavaScript

## Função retornando Valores

```
<html>
  <head>
    <meta charset="utf-8">
    <title>JavaScript</title>

    <script type="text/javascript">
      function media(a, b, c){
        calculo =(a+b+c)/3;
        return calculo;
      }
    </script>

  </head>

  <body>
    <script type="text/javascript">
      result = media(3,5,8);
      alert(result);
    </script>
  </body>
</html>
```

# JavaScript

## Instrução if/else

```
1 <html>
2   <head>
3     <meta charset="utf-8">
4     <title>JavaScript</title>
5   </head>
6   <body>
7     <h4>JavaScript</h4>
8     <p>Digite 2.5 e depois altere para parseFloat()</p>
9     <script>
10      var a = prompt("Digite um valor: ");
11      if (parseInt(a) > 2){
12        alert(parseInt(a) + " maior que 2");
13      }else{
14        alert(parseInt(a) + " menor ou igual a 2");
15      }
16    </script>
17  </body>
18 </html>
```

# JavaScript

## Instrução switch( )

```
<html>
  <head>
    <meta charset="utf-8">
    <title>JavaScript</title>
  </head>

  <body>
    <script type="text/javascript">
      dia_semana = 5;
      switch (dia_semana) {
        case 0 : document.write("Domingo");
        break;
        case 1 : document.write("Segunda-feira");
        break;
        case 2 : document.write("Terça-feira");
        break;
        case 3 : document.write("Quarta-feira");
        break;
        case 4 : document.write("Quinta-feira");
        break;
        case 5 : document.write("Sexta-feira");
        Break;
        case 6 : document.write("Sábado");
        break;
        default : alert("Dia da semana inválido!");
      }
    </script>
  </body>
</html>
```



# JavaScript

## Loop for

```
<html>
  <head>
    <meta charset="utf-8">
    <title>JavaScript</title>
  </head>

  <body>
    <script type="text/javascript">
      for (i = 1; i < 5; i++){
        document.write("Esta é a linha ",i, "<br>");
      }
    </script>
  </body>
</html>
```

# JavaScript

## Loop while

```
<html>
  <head>
    <meta charset="utf-8">
    <title>JavaScript</title>
  </head>

  <body>
    <script type="text/javascript">
      valor = new Array(3,2,4,1,5);
      i = 0;
      total = 0;
      while (total < 10){
        total += valor[i];
        i++;
      }
      document.write("Total = " + total);
      // Resultado: Total = 10
    </script>
  </body>
</html>
```

# JavaScript

## Propriedades e Funções Matemáticas

Propriedades	Descrição
<code>Math.E</code>	Retorna a base dos logaritmos naturais (aproximadamente 2.718).
<code>Math.LN2</code>	Retorna o valor do logaritmo de 2 (aproximadamente 0.693).
<code>Math.LOG2E</code>	Retorna a base do logaritmo de 2 (aproximadamente 1.442).
<code>Math.LN10</code>	Retorna o valor do logaritmo de 10 (aproximadamente 2.302).
<code>Math.LOG10E</code>	Retorna a base do logaritmo de 10 (aproximadamente 0.434).
<code>Math.SQRT2</code>	Retorna a raiz quadrada de 2 (aproximadamente 1.414).
<code>Math.SQRT_2</code>	Retorna a raiz quadrada de 1/2 (aproximadamente 0.707).
<code>Math.PI</code>	retorna o valor de PI (aproximadamente 3.14159).

# JavaScript

## Propriedades e Funções Matemáticas

Métodos	Descrição
<b>Math.abs</b> (número)	<i>Retorna o valor absoluto do número (ponto flutuante).</i>
<b>Math.pow</b> (base, expoente)	<i>Retorna o cálculo do exponencial.</i>
<b>Math.max</b> (número1, número2)	<i>Retorna o maior número entre os fornecidos.</i>
<b>Math.min</b> (número1, número2)	<i>Retorna o menor número entre dos dois fornecidos.</i>
<b>Math.sqrt</b> (número)	<i>Retorna a raiz quadrada do número.</i>
<b>Math.sin</b> (número)	<i>Retorna o seno de um número (ângulo em radianos).</i>
<b>Math.asin</b> (número)	<i>Retorna o arco seno de um número (em radianos).</i>
<b>Math.cos</b> (número)	<i>Retorna o cosseno de um número (ângulo em radianos).</i>
<b>Math.acos</b> (número)	<i>Retorna o arco cosseno de um número (em radianos).</i>
<b>Math.tan</b> (número)	<i>Retorna a tangente de um número (ângulo em radianos).</i>
<b>Math.atan</b> (número)	<i>Retorna o arco tangente de um número (em radianos).</i>
<b>Math.log</b> (número)	<i>Retorna o logaritmo de um número.</i>

# JavaScript

## Arredondando e Truncando Valores

Métodos	Descrição
<code>Math.ceil(número)</code>	<i>Retorna o próximo valor inteiro maior que o número.</i>
<code>Math.floor(número)</code>	<i>Retorna o próximo valor inteiro menor que o número.</i>
<code>Math.round(número)</code>	<i>Retorna o valor inteiro do número, arredondado.</i>

# JavaScript

## Criando Números Pseudo-Aleatórios

Métodos	Descrição
<code>Math.random()</code>	<i>Retorna um número decimal entre 0 e 1 aleatório (não exige nenhum parâmetro).</i>

# JavaScript

## Alterando Valores de Datas

Uma variedade de métodos .set permite configurar componentes de um objeto Date:

Método	Descrição
<code>.setDate( )</code>	<i>Estabelece dia do Mês.</i>
<code>.setMonth( )</code>	<i>Estabelece o Mês. (Valores de 0 à 11; Janeiro = 0).</i>
<code>.SetDay( )</code>	<i>Estabelece o dia da Semana (Valores de 0 à 6; Domingo = 0).</i>
<code>.setYear( )</code>	<i>Estabelece o Ano. (2 dígitos)</i>
<code>.setFullYear( )</code>	<i>Estabelece o Ano. (4 dígitos)</i>
<code>.setTime( )</code>	<i>Estabelece a Hora (e a data) em milésimos de segundos.</i>
<code>.setHours( )</code>	<i>Estabelece a Hora.</i>
<code>.setMinutes( )</code>	<i>Estabelece os Minutos.</i>
<code>.SetSeconds( )</code>	<i>Estabelece os Segundos.</i>

# JavaScript

## Alterando Valores de Datas

```
<html>
  <head>
    <meta charset="utf-8">
    <title>JavaScript</title>
  </head>

  <body>
    <script type="text/javascript">
      hoje = new Date( );
      document.write(hoje);
      hoje.setFullYear(2020);
      alert(hoje);
    </script>
  </body>
</html>
```



# JavaScript

## Obtendo Valores de Datas

O método `.get` obtém valores de um objeto `Date` :

Método	Descrição
<code>.getDate( )</code>	<i>Obtém o dia do Mês.</i>
<code>.getMonth( )</code>	<i>Obtém o Mês. (Valores de 0 à 11; Janeiro = 0).</i>
<code>.getDay( )</code>	<i>Obtém o dia da Semana ( Valores de 0 à 6; Domingo = 0).</i>
<code>.getFullYear( )</code>	<i>Obtém o Ano. (2 dígitos)</i>
<code>.getFullYear( )</code>	<i>Obtém o Ano. (4 dígitos)</i>
<code>.getTime( )</code>	<i>Obtém a Hora (e a data) em milissegundos.</i>
<code>.getHours( )</code>	<i>Obtém a Hora.</i>
<code>.getMinutes( )</code>	<i>Obtém os Minutos.</i>
<code>.getSeconds( )</code>	<i>Obtém os Segundos.</i>

# JavaScript

## Obtendo Valores de Datas

```
<html>
  <head>
    <meta charset="utf-8">
    <title>JavaScript</title>
  </head>

  <body>
    <script type="text/javascript">
      hoje = new Date("Apr 24 2015");
      dia = hoje.getDate( );
      alert(dia);
    </script>
  </body>
</html>
```

# JavaScript

```
1 <html>
2   <head>
3     <meta charset="utf-8">
4     <title>JavaScript</title>
5     <script src="evento.js"></script>
6   </head>
7   <body onload="Tempo(), nData()">
8     <h4>JavaScript</h4>
9     <p id="horario"></p>
10    <p id="n_data"></p>
11  </body>
12 </html>
```

**JavaScript**

13:31:45

04 / 04 / 2017

# JavaScript

```
1 function Tempo(){
2     horario = new Date()
3     hora = horario.getHours()
4     minuto = horario.getMinutes()
5     segundo = horario.getSeconds()
6     if (hora < 10) {
7         hora = "0" + hora
8     }
9     if (minuto < 10) {
10         minuto = "0" + minuto
11     }
12     if (segundo < 10) {
13         segundo = "0" + segundo
14     }
15     document.getElementById("horario").innerHTML = hora + ":" + minuto + ":" + segundo;
16 }
17 function nData() {
18     data = new Date()
19     dia = data.getDate()
20     mes = data.getMonth() + 1
21     ano = data.getFullYear()
22     if (dia < 10) {
23         dia = "0" + dia
24     }
25     if (mes < 10) {
26         mes = "0" + mes
27     }
28     document.getElementById("n_data").innerHTML = dia + " / " + mes + " / " + ano;
29 }
```

# JavaScript

## Atividade

- 1) No código que aparece hora e data altere, o mês deverá aparecer por extenso e também qual é o dia da semana.
- 2) Faça uma função para conversão de temperatura.