

TUTORIAIS JAVASCRIPT

O Objeto Math

Copyright 2013 – Todos os Direitos Reservados
Jorge Eider F. da Silva

Proibida a reprodução deste documento no todo ou em parte por quaisquer meios, seja digital, eletrônico ou impresso sem a expressa autorização do autor por escrito. Os infratores serão punidos de acordo com a Lei.

O Objeto Math

Introdução

O objeto **Math** permite executar tarefas matemáticas. Esse objeto possui várias constantes matemáticas e métodos. Veja alguns exemplos:

```
var x = Math.PI;  
var y = Math.sqrt(25);
```

Ele não é um construtor. Todas as propriedades e métodos de **Math** podem ser chamadas diretamente utilizando **Math** como um objeto sem criá-lo.

Constantes matemáticas

O JavaScript fornece oito constantes matemáticas que podem ser acessadas a partir do objeto **Math**. São as seguintes:

- | | |
|-----------------------|--------------|
| • E | Math.E |
| • PI | Math.PI |
| • Raiz quadrada de 2 | Math.SQRT2 |
| • Raiz quadrada de ½ | Math.SQRT1_2 |
| • Log natural de 2 | Math.LN2 |
| • Log natural de 10 | Math.LN10 |
| • Log de E na base 2 | Math.LOG2E |
| • Log de E na base 10 | Math.LOG10E |

Vejamos algumas dessas constantes em exemplos práticos:

A constante PI

A constante **PI** retorna o raio da área do círculo pelo quadrado de seu raio, aproximadamente 3.1415. Ela é muito utilizada na trigonometria.

Sua sintaxe é a seguinte:

```
Math.PI;
```

Vejamos um exemplo prático:

Exemplo 1

Nesse exemplo vamos utilizar a constante **PI** e imprimir o seu resultado.

Para isso:

1. Digite o código abaixo no seu editor de texto e salve-o como: **js18-exemplo1.html**.

```
<!DOCTYPE html>  
<html>  
Tutorial JavaScript18 - Exemplo 1 - O Objeto Math  
<head>  
  <title>Tutorial JavaScript18 - Exemplo 1 - O Objeto Math</title><p>
```

```
<script language="javascript">
  // Cria uma variável e atribui a constante PI.
  var pi = Math.PI;
  // Imprime o valor de PI.
  document.writeln("Valor de PI: " + pi + "<br>");
</script>
</head>
<body>
</body>
</html>
```

2. O resultado desse código após executado será o seguinte:

```
Tutorial JavaScript18 - Exemplo 1 - O Objeto Math
Valor de PI: 3.141592653589793
```

A constante SQRT1_2

Essa constante retorna a raiz quadrada de $\frac{1}{2}$, aproximadamente 0.707. Muito utilizada em equações matemáticas.

Sua sintaxe é a seguinte:

```
Math.SQRT1_2;
```

Vejamos um exemplo prático:

Exemplo 2

Nesse exemplo vamos utilizar a constante **SQRT1_2** e imprimir o seu resultado.

Para isso:

1. Digite o código abaixo no seu editor de texto e salve-o como: **js18-exemplo2.html**.

```
<!DOCTYPE html>
<html>
  Tutorial JavaScript18 - Exemplo 2 - O Objeto Math
<head>
  <title>Tutorial JavaScript18 - Exemplo 2 - O Objeto Math</title><p>
<script language="javascript">
  // Cria uma variável e atribui a constante SQRT1_2.
  var x = Math.SQRT1_2;
  // Imprime o valor de SQRT1_2.
  document.writeln("Valor de SQRT1_2: " + x + "<br>");
</script>
</head>
<body>
</body>
</html>
```

2. O resultado desse código após executado será o seguinte:

Tutorial JavaScript18 - Exemplo 2 - O Objeto Math

Valor de SQRT1_2: 0.7071067811865476

A constante SQRT2

Essa constante retorna a raiz quadrada de 2, aproximadamente 1.414. Muito utilizada em equações matemáticas.

Sua sintaxe é a seguinte:

```
Math.SQRT2;
```

Vejamos um exemplo prático:

Exemplo 3

Nesse exemplo vamos utilizar a constante **SQRT2** e imprimir o seu resultado.

Para isso:

1. Digite o código abaixo no seu editor de texto e salve-o como: **js18-exemplo3.html**.

```
<!DOCTYPE html>
<html>
Tutorial JavaScript18 - Exemplo 3 - O Objeto Math
<head>
  <title>Tutorial JavaScript18 - Exemplo 3 - O Objeto Math</title><p>
<script language="javascript">
  // Cria uma variável e atribui a constante SQRT2 a ela.
  var x = Math.SQRT2;
  // Imprime o valor de SQRT2.
  document.writeln("Valor de SQRT2: " + x + "<br>");
</script>
</head>
<body>
</body>
</html>
```

2. O resultado desse código após executado será o seguinte:

Tutorial JavaScript18 - Exemplo 3 - O Objeto Math

Valor de SQRT2: 1.4142135623730951

Métodos matemáticos

Além das constantes matemáticas vistas anteriormente, o JavaScript fornece também alguns métodos matemáticos que podem ser utilizados nas suas aplicações. Vejamos alguns deles em detalhes (pelo menos os mais utilizados) através de exemplos práticos:

O método abs()

Esse método retorna o valor absoluto de um número, ou seja, o valor sem sinal. Muito utilizado em expressões matemáticas.

Sua sintaxe é a seguinte:

```
Math.abs(x);
```

Vejamos um exemplo prático:

Exemplo 4

Nesse exemplo vamos utilizar o método **abs(x)** para mostrar o mesmo resultado de dois números, um negativo e outro positivo.

Para isso:

1. Digite o código abaixo no seu editor de texto e salve-o como: **js18-exemplo4.html**.

```
<!DOCTYPE html>
<html>
Tutorial JavaScript18 - Exemplo 4 - O Objeto Math
<head>
  <title>Tutorial JavaScript18 - Exemplo 4 - O Objeto Math</title><p>
<script language="javascript">
  // Cria duas variáveis e atribui valores diferentes a elas.
  var x = Math.abs(25);
  var y = Math.abs(-25);
  // Imprime os valores de x e y.
  document.writeln("Valor de x: " + x + "<br>");
  document.writeln("Valor de y: " + y + "<br>");
</script>
</head>
<body>
</body>
</html>
```

2. O resultado desse código após executado será o seguinte:

```
Tutorial JavaScript18 - Exemplo 4 - O Objeto Math
Valor de x: 25
Valor de y: 25
```

O método ceil()

Esse método arredonda um número para cima para o valor mais próximo, e retorna o resultado. Se o argumento passado for um inteiro, o valor não será arredondado.

Sua sintaxe é a seguinte:

```
Math.ceil(x);
```

Vejamos um exemplo prático:

Exemplo 5

Nesse exemplo vamos utilizar o método **ceil(x)** para mostrar como fica o arredondamento de alguns números.

Para isso:

1. Digite o código abaixo no seu editor de texto e salve-o como: **js18-exemplo5.html**.

```
<!DOCTYPE html>
<html>
Tutorial JavaScript18 - Exemplo 5 - O Objeto Math
<head>
  <title>Tutorial JavaScript18 - Exemplo 5 - O Objeto Math</title><p>
    Sejam:<p>
    x1 = (2.5)<br>
    x2 = (0.925)<br>
    x3 = (-0.25)<br>
    x4 = (-9.9)<p>
    <script language="javascript">
    // Cria quatro variáveis e atribui valores diferentes a elas.
    var x1 = Math.ceil(2.5);
    var x2 = Math.ceil(0.925);
    var x3 = Math.ceil(-0.25);
    var x4 = Math.ceil(-9.9);
    // Imprime os valores de x1, x2, x3 e x4.
    document.writeln("Valor de ceil(x1): " + x1 + "<br>");
    document.writeln("Valor de ceil(x2): " + x2 + "<br>");
    document.writeln("Valor de ceil(x3): " + x3 + "<br>");
    document.writeln("Valor de ceil(x4): " + x4 + "<br>");
  </script>
</head>
<body>
</body>
</html>
```

2. O resultado desse código após executado será o seguinte:

```
Tutorial JavaScript18 - Exemplo 5 - O Objeto Math

Sejam:

x1 = (2.5)
x2 = (0.925)
x3 = (-0.25)
x4 = (-9.9)

Valor de ceil(x1): 3
Valor de ceil(x2): 1
Valor de ceil(x3): 0
Valor de ceil(x4): -9
```

O método cos(x)

Esse método retorna o cosseno de um número, ou seja, um número entre -1 e 1, que representa o cosseno do ângulo representado no parâmetro x. Muito utilizado na trigonometria.

Sua sintaxe é a seguinte:

```
Math.cos(x);
```

Vejamos um exemplo prático:

Exemplo 6

Nesse exemplo vamos utilizar o método **cos(x)** para mostrar o cosseno de alguns ângulos.

Para isso:

1. Digite o código abaixo no seu editor de texto e salve-o como: **js18-exemplo6.html**.

```
<!DOCTYPE html>
<html>
Tutorial JavaScript18 - Exemplo 6 - O Objeto Math
<head>
  <title>Tutorial JavaScript18 - Exemplo 6 - O Objeto Math</title><p>
    Sejam:<p>
    x1 = (0)<br>
    x2 = (45)<br>
    x3 = (90)<br>
    x4 = (180)<p>
    <script type="text/javascript">
      // Cria quatro variáveis e atribui valores diferentes a elas.
      var x1 = Math.cos(0);
      var x2 = Math.cos(45);
      var x3 = Math.cos(90);
      var x4 = Math.cos(180);
      // Imprime os valores dos cossenos de x1, x2, x3 e x4.
      document.writeln("Valor de cos(x1): " + x1 + "<br>");
      document.writeln("Valor de cos(x2): " + x2 + "<br>");
      document.writeln("Valor de cos(x3): " + x3 + "<br>");
      document.writeln("Valor de cos(x4): " + x4 + "<br>");
    </script>
  </head>
<body>
</body>
</html>
```

2. O resultado desse código após executado será o seguinte:

```
Tutorial JavaScript18 - Exemplo 6 - O Objeto Math

Sejam:

x1 = (0)
x2 = (45)
x3 = (90)
x4 = (180)

Valor de cos(x1): 1
Valor de cos(x2): 0.5253219888177297
Valor de cos(x3): -0.4480736161291701
Valor de cos(x4): -0.5984600690578581
```

O método floor()

Esse método arredonda um número para baixo para o inteiro mais próximo, e retorna o resultado. Se o argumento passado for um inteiro, o valor não será arredondado. Muito utilizado em expressões matemáticas.

Sua sintaxe é a seguinte:

```
Math.floor(x);
```

Vejamos um exemplo prático:

Exemplo 7

Nesse exemplo vamos utilizar o método **floor(x)** para mostrar como fica o arredondamento de alguns números.

Para isso:

1. Digite o código abaixo no seu editor de texto e salve-o como: **js18-exemplo7.html**.

```
<!DOCTYPE html>
<html>
Tutorial JavaScript18 - Exemplo 7 - O Objeto Math
<head>
  <title>Tutorial JavaScript18 - Exemplo 7 - O Objeto Math</title><p>
    Sejam:<p>
    x1 = (2.5)<br>
    x2 = (0.925)<br>
    x3 = (-0.25)<br>
    x4 = (-9.9)<p>
    <script type="text/javascript">
    // Cria quatro variáveis e atribui valores diferentes a elas.
    var x1 = Math.floor(2.5);
    var x2 = Math.floor(0.925);
    var x3 = Math.floor(-0.25);
    var x4 = Math.floor(-9.9);
    // Imprime os valores de Math.floor de x1, x2, x3 e x4.
    document.writeln("Valor de floor(x1): " + x1 + "<br>");
    document.writeln("Valor de floor(x2): " + x2 + "<br>");
    document.writeln("Valor de floor(x3): " + x3 + "<br>");
    document.writeln("Valor de floor(x4): " + x4 + "<br>");
  </script>
</head>
<body>
</body>
</html>
```

2. O resultado desse código após executado será o seguinte:

```
Tutorial JavaScript18 - Exemplo 7 - O Objeto Math

Sejam:

x1 = (2.5)
x2 = (0.925)
x3 = (-0.25)
x4 = (-9.9)

Valor de floor(x1): 2
Valor de floor(x2): 0
Valor de floor(x3): -1
Valor de floor(x4): -10
```


O método max()

Esse método retorna o número com maior valor da faixa de números apresentada. Retorna **-Infinity** se nenhum argumento for definido ou **NaN** se um ou mais argumentos não forem números. Muito utilizado em expressões matemáticas.

Sua sintaxe é a seguinte:

```
Math.max(x,y,z,...,n);
```

Vejamos um exemplo prático:

Exemplo 8

Nesse exemplo vamos utilizar o método **max()** para mostrar como se comportam os vários tipos de parâmetros.

Para isso:

1. Digite o código abaixo no seu editor de texto e salve-o como: **js18-exemplo8.html**.

```
<!DOCTYPE html>
<html>
Tutorial JavaScript18 - Exemplo 8 - O Objeto Math
<head>
  <title>Tutorial JavaScript18 - Exemplo 8 - O Objeto Math</title><p>
  Sejam:<p>
  x1 = ()<br>
  x2 = (2,6,90,12,5)<br>
  x3 = ("A")<br>
  x4 = (-9,-6,1,0,3*2)<p>
  <script type="text/javascript">
  // Cria quatro variáveis e atribui valores diferentes a elas.
  var x1 = Math.max();
  var x2 = Math.max(2,6,90,12,5);
  var x3 = Math.max("A");
  var x4 = Math.max(-9,-6,1,0,3*2);
  // Imprime os valores de x1, x2, x3 e x4.
  document.writeln("Valor de x1: " + x1 + "<br>");
  document.writeln("Valor de x2: " + x2 + "<br>");
  document.writeln("Valor de x3: " + x3 + "<br>");
  document.writeln("Valor de x4: " + x4 + "<br>");
</script>
</head>
<body>
</body>
</html>
```

2. O resultado desse código após executado será o seguinte:

Tutorial JavaScript18 - Exemplo 8 - O Objeto Math

Sejam:

```
x1 = 0  
x2 = (2,6,90,12,5)  
x3 = ("A")  
x4 = (-9,-6,1,0,3*2)
```

Valor de x1: -Infinity

Valor de x2: 90

Valor de x3: NaN

Valor de x4: 6

O método min()

Esse método retorna o número com menor valor da faixa de números apresentada. Retorna **-Infinity** se nenhum argumento for definido ou **NaN** se um ou mais argumentos não forem números. Muito utilizado em expressões matemáticas.

Sua sintaxe é a seguinte:

```
Math.min(x,y,z,...,n);
```

Vejamos um exemplo prático:

Exemplo 9

Nesse exemplo vamos utilizar o método **min()** para mostrar como se comportam os vários tipos de parâmetros.

Para isso:

1. Digite o código abaixo no seu editor de texto e salve-o como: **js18-exemplo9.html**.

```
<!DOCTYPE html>  
<html>  
Tutorial JavaScript18 - Exemplo 9 - O Objeto Math  
<head>  
  <title>Tutorial JavaScript18 - Exemplo 9 - O Objeto Math</title><p>  
    Sejam:<p>  
    x1 = ()<br>  
    x2 = (2,6,90,12,5)<br>  
    x3 = ("A")<br>  
    x4 = (-9,-6,1,0,3*2)<p>  
    <script type="text/javascript">  
      // Cria quatro variáveis e atribui valores diferentes a elas.  
      var x1 = Math.min();  
      var x2 = Math.min(2,6,90,12,5);  
      var x3 = Math.min("A");  
      var x4 = Math.min(-9,-6,1,0,3*2);  
      // Imprime os valores de x1, x2, x3 e x4.  
      document.writeln("Valor de x1: " + x1 + "<br>");  
      document.writeln("Valor de x2: " + x2 + "<br>");  
      document.writeln("Valor de x3: " + x3 + "<br>");  
      document.writeln("Valor de x4: " + x4 + "<br>");
```

```
</script>
</head>
<body>
</body>
</html>
```

2. O resultado desse código após executado será o seguinte:

Tutorial JavaScript18 - Exemplo 9 - O Objeto Math

Sejam:

```
x1 = 0
x2 = (2,6,90,12,5)
x3 = ("A")
x4 = (-9,-6,1,0,3*2)
```

```
Valor de x1: Infinity
Valor de x2: 2
Valor de x3: NaN
Valor de x4: -9
```

O método pow()

Esse método retorna o valor de um certo número **x** elevado a potência de um certo número **y** (**x^y**). Muito utilizado em expressões matemáticas.

Sua sintaxe é a seguinte:

```
Math.pow(x, y);
```

Vejamos um exemplo prático:

Exemplo 10

Nesse exemplo vamos utilizar o método **pow()** para mostrar a potência de alguns números.

Para isso:

1. Digite o código abaixo no seu editor de texto e salve-o como: **js18-exemplo10.html**.

```
<!DOCTYPE html>
<html>
Tutorial JavaScript18 - Exemplo 10 - O Objeto Math
<head>
  <title>Tutorial JavaScript18 - Exemplo 10 - O Objeto Math</title><p>
    Sejam:<p>
    x1 = (2,4)<br>
    x2 = (111,2)<br>
    x3 = (3,-3)<br>
    x4 = ((3*2+4),2)<p>
  <script type="text/javascript">
    // Cria quatro variáveis e atribui valores diferentes a elas.
    var x1 = Math.pow(2,4);
    var x2 = Math.pow(111,2);
```

```

var x3 = Math.pow(3,-3);
var x4 = Math.pow((3*2+4),2);
// Imprime os valores de x1, x2, x3 e x4.
document.writeln("Valor de x1: " + x1 + "<br>");
document.writeln("Valor de x2: " + x2 + "<br>");
document.writeln("Valor de x3: " + x3 + "<br>");
document.writeln("Valor de x4: " + x4 + "<br>");
</script>
</head>
<body>
</body>
</html>

```

2. O resultado desse código após executado será o seguinte:

```

Tutorial JavaScript18 - Exemplo 10 - O Objeto Math

Sejam:

x1 = (2,4)
x2 = (111,2)
x3 = (3,-3)
x4 = ((3*2+4),2)

Valor de x1: 16
Valor de x2: 12321
Valor de x3: 0.037037037037037035
Valor de x4: 100

```

O método random()

Esse método retorna um número randômico entre 0 e 1 (exclusive). É muito usado em expressões matemáticas e no desenvolvimento de jogos para sortear números aleatórios, como Bingo, Dados, Loteria, entre outros.

Sua sintaxe é a seguinte:

```
Math.random();
```

Vejamos um exemplo prático:

Exemplo 11

Nesse exemplo vamos utilizar o método **random()** para mostrar algumas faixas de números.

Para isso:

1. Digite o código abaixo no seu editor de texto e salve-o como: **js18-exemplo11.html**.

```

<!DOCTYPE html>
<html>
Tutorial JavaScript18 - Exemplo 11 - O Objeto Math
<head>
  <title>Tutorial JavaScript18 - Exemplo 11 - O Objeto Math</title><p>

```

```

Sejam:<p>
x1 = Math.random()<br>
x2 = (Math.random()*50)+51<br>
x3 = (Math.random()*10)+1<br>
x4 = (Math.random()*100)+1<p>
<script type="text/javascript">
// Cria quatro variáveis e atribui valores diferentes a elas.
var x1 = Math.random();
var x2 = Math.floor((Math.random()*50)+51);
var x3 = Math.floor((Math.random()*10)+1);
var x4 = Math.floor((Math.random()*100)+1);
// Imprime os valores de x1, x2, x3 e x4.
document.writeln("Valor de x1: " + x1 + "<br>");
document.writeln("Número randômico entre 50 e 100: " + x2 + "<br>");
document.writeln("Número randômico entre 1 e 10: " + x3 + "<br>");
document.writeln("Número randômico entre 1 e 100: " + x4 + "<br>");
</script>
</head>
<body>
</body>
</html>

```

- O resultado desse código após executado será o seguinte:

Tutorial JavaScript18 - Exemplo 11 - O Objeto Math

Sejam:

```

x1 = Math.random()
x2 = (Math.random()*50)+51
x3 = (Math.random()*10)+1
x4 = (Math.random()*100)+1

```

Valor de x1: 0.8617306880187243
Número randômico entre 50 e 100: 53
Número randômico entre 1 e 10: 4
Número randômico entre 1 e 100: 24

- Entretanto, os valores serão alterados a cada execução.

O método round()

Esse método tem por finalidade arredondar números, deixando-os somente com a parte inteira. O arredondamento para mais ou para menos depende do primeiro número da parte decimal. É muito usado em expressões matemáticas.

Sua sintaxe é a seguinte:

```
Math.round();
```

Vejamos um exemplo prático:

Exemplo 12

Nesse exemplo vamos utilizar o método **round()** para mostrar como fica o arredondamento de alguns números.

Para isso:

1. Digite o código abaixo no seu editor de texto e salve-o como: **js18-exemplo12.html**.

```
<!DOCTYPE html>
<html>
Tutorial JavaScript18 - Exemplo 12 - O Objeto Math
<head>
  <title>Tutorial JavaScript18 - Exemplo 12 - O Objeto Math</title><p>
    Sejam:<p>
    x1 = 12.45<br>
    x2 = -0.95<br>
    x3 = 10.005<br>
    x4 = -54.89<p>
    <script type="text/javascript">
    // Cria quatro variáveis e atribui valores diferentes a elas.
    var x1 = Math.round(12.45);
    var x2 = Math.round(-0.95);
    var x3 = Math.round(10.005);
    var x4 = Math.round(-54.89);
    // Imprime os valores de x1, x2, x3 e x4.
    document.writeln("Valor de x1: " + x1 + "<br>");
    document.writeln("Valor de x2: " + x2 + "<br>");
    document.writeln("Valor de x3: " + x3 + "<br>");
    document.writeln("Valor de x4: " + x4 + "<br>");
  </script>
</head>
<body>
</body>
</html>
```

2. O resultado desse código após executado será o seguinte:

```
Tutorial JavaScript18 - Exemplo 12 - O Objeto Math

Sejam:

x1 = 12.45
x2 = -0.95
x3 = 10.005
x4 = -54.89

Valor de x1: 12
Valor de x2: -1
Valor de x3: 10
Valor de x4: -55
```

O método sin(x)

Esse método retorna o seno de um número, ou seja, retorna um número entre -1 e 1, que representa o seno do ângulo representado no parâmetro **x**. Muito utilizado na trigonometria.

Sua sintaxe é a seguinte:

```
Math.sin(x);
```

Vejamos um exemplo prático:

Exemplo 13

Nesse exemplo vamos utilizar o método **sin(x)** para mostrar o cosseno de alguns ângulos.

Para isso:

1. Digite o código abaixo no seu editor de texto e salve-o como: **js18-exemplo13.html**.

```
<!DOCTYPE html>
<html>
Tutorial JavaScript18 - Exemplo 13 - O Objeto Math
<head>
  <title>Tutorial JavaScript18 - Exemplo 13 - O Objeto Math</title><p>
    Sejam:<p>
    x1 = (0)<br>
    x2 = (45)<br>
    x3 = (90)<br>
    x4 = (180)<p>
    <script type="text/javascript">
    // Cria quatro variáveis e atribui valores diferentes a elas.
    var x1 = Math.sin(0);
    var x2 = Math.sin(45);
    var x3 = Math.sin(90);
    var x4 = Math.sin(180);
    // Imprime os senos de x1, x2, x3 e x4.
    document.writeln("Os senos dos ângulos acima são:<p>");
    document.writeln("Seno de x1 => " + x1 + "<br>");
    document.writeln("Seno de x2 => " + x2 + "<br>");
    document.writeln("Seno de x3 => " + x3 + "<br>");
    document.writeln("Seno de x4 => " + x4 + "<br>");
  </script>
</head>
<body>
</body>
</html>
```

2. O resultado desse código após executado será o seguinte:

```
Tutorial JavaScript18 - Exemplo 13 - O Objeto Math

Sejam:

x1 = (0)
x2 = (45)
x3 = (90)
x4 = (180)

Os senos dos ângulos acima são:

Seno de x1 => 0
Seno de x2 => 0.8509035245341184
Seno de x3 => 0.8939966636005579
Seno de x4 => -0.8011526357338304
```

O método sqrt(x)

Esse método retorna a raiz quadrada de um número qualquer representado no parâmetro **x**. Se **x** for um número negativo, **NaN** é retornado. Muito utilizado em expressões matemáticas.

Sua sintaxe é a seguinte:

```
Math.sqrt(x);
```

Vejamos um exemplo prático:

Exemplo 14

Nesse exemplo vamos utilizar o método **sqrt(x)** para extrair a raiz quadrada de alguns números.

Para isso:

1. Digite o código abaixo no seu editor de texto e salve-o como: **js18-exemplo14.html**.

```
<!DOCTYPE html>
<html>
Tutorial JavaScript18 - Exemplo 14 - O Objeto Math
<head>
  <title>Tutorial JavaScript18 - Exemplo 14 - O Objeto Math</title><p>
    Sejam:<p>
    x1 = 2<br>
    x2 = 3<br>
    x3 = 12321<br>
    x4 = -25<p>
    <script type="text/javascript">
    // Cria quatro variáveis e atribui valores diferentes a elas.
    var x1 = Math.sqrt(2);
    var x2 = Math.sqrt(3);
    var x3 = Math.sqrt(12321);
    var x4 = Math.sqrt(-25);
    // Imprime as raízes quadradas de x1, x2, x3 e x4.
    document.writeln("As raízes quadradas dos números dados são:<p>");
    document.writeln("Raiz de x1 => " + x1 + "<br>");
    document.writeln("Raiz de x2 => " + x2 + "<br>");
    document.writeln("Raiz de x3 => " + x3 + "<br>");
    document.writeln("Raiz de x4 => " + x4 + "<br>");
    </script>
  </head>
<body>
</body>
</html>
```

2. O resultado desse código após executado será o seguinte:

Tutorial JavaScript18 - Exemplo 14 - O Objeto Math

Sejam:

$x_1 = 2$

$x_2 = 3$

$x_3 = 12321$

$x_4 = -25$

As raízes quadradas dos números dados são:

Raiz de $x_1 \Rightarrow 1.4142135623730951$

Raiz de $x_2 \Rightarrow 1.7320508075688772$

Raiz de $x_3 \Rightarrow 111$

Raiz de $x_4 \Rightarrow \text{NaN}$

Exercícios de fixação

- 1) Esse método retorna a raiz quadrada de um número qualquer representado no parâmetro **x**.
 - a) `sqrt2(x)`
 - b) `sqrt(x)`
 - c) `sqrt1_2(x)`
 - d) `sqrt_x`
- 2) Que método devemos utilizar para nos informar o maior número de uma faixa de valores?
 - a) `max()`
 - b) `maximum()`
 - c) `major()`
 - d) `max{}`
- 3) Se quiséssemos arredondar um número qualquer, que método deveríamos utilizar?
 - a) `ceil()`
 - b) `floor()`
 - c) `int()`
 - d) `round()`
- 4) Esse método retorna o valor de um certo número **x** elevado a potência de um certo número **y** (**x^y**).
 - a) `pot()`
 - b) `ceil()`
 - c) `pow()`
 - d) `max{}`
- 5) Esse método retorna **-Infinity** se nenhum argumento for definido ou **NaN** se um ou mais argumentos não forem números.
 - a) `min()`
 - b) `max ()`
 - c) `sin()`
 - d) `cos()`
- 6) Qual o resultado das duas linhas de código a seguir após a sua execução?

```
var x1 = Math.round(12.945);
var x2 = Math.round(-0.195);
```

 - a) 13 e 1
 - b) 13 e 0
 - c) 12 e 0
 - d) 12 e 1
- 7) Qual o resultado do código a seguir após a sua execução?

```
var x1 = Math.pow(max(3,2),min(5,2));
var x2 = Math.round(19.5);
document.write(x1 * x2);
```

 - a) 120

- b) 140
- c) 160
- d) 180

8) O método _____ retorna o valor absoluto de um número, ou seja, o valor sem sinal.

- a) ceil()
- b) module()
- c) absolute()
- d) abs()

9) Qual o resultado da expressão após a sua execução?

```
var x1 = Math.floor(Math.cos(90) + Math.min(Math.pow(5,2)));  
var x2 = Math.sqrt(Math.pow(4,2));  
document.write(x1 * x2 + x1);
```

- a) 120
- b) 130
- c) 140
- d) 150

10) Esse método arredonda um número para baixo para o inteiro mais próximo, e retorna o resultado.

- a) round()
- b) ceil()
- c) abs()
- d) floor()

Exercícios propostos

- 1) Crie uma rotina para gerar 20 números randômicos inteiros entre 10 e 20, e preencha uma array somente com os números pares, depois mostre o resultado dessa array em uma janela de diálogo.
- 2) Crie uma rotina para gerar 10 números randômicos inteiros entre 1 e 20, e em seguida extraia a raiz quadrada somente dos números ímpares, depois mostre o resultado em uma janela de diálogo.
- 3) Crie uma rotina para calcular o quadrado de 10 números pares entre 0 e 20, e em seguida mostre o resultado em uma janela de diálogo, linha por linha.
- 4) Crie uma rotina para imprimir em uma janela de diálogo a raiz quadrada de 30 números randômicos quaisquer e inteiros somente os divisíveis por 3.
