



# PROGRAMAÇÃO FRONT-END

GIOVANNI RIBEIRO



# Funções Javascript



Uma função JavaScript é um bloco de código projetado para executar uma tarefa específica.

Apesar de ser interessante a possibilidade de alterar o documento todo por meio do JavaScript, é muito comum que as alterações sejam feitas quando o usuário executa alguma ação, como por exemplo, mudar o conteúdo de um botão ao clicar nele e não quando a página carrega. Porém, por padrão, qualquer código colocado no <script>, como fizemos anteriormente, é executado assim que o navegador lê ele.

Para guardarmos um código para ser executado em algum outro momento, por exemplo, quando o usuário clicar num botão, é necessário utilizar alguns recursos do JavaScript no navegador.

Primeiro vamos criar uma **função**:

# Funções Javascript



Para chamar a função mostraAlerta só precisamos utilizar o nome da função e logo depois abrir e fechar parênteses.

```
function mostraAlerta() {  
    alert("Alerta...");  
}  
  
mostraAlerta();
```

Uma função JavaScript é definida com a **function** palavra-chave, seguida por um **nome**, seguido de parênteses **()**.

```
function minhaFunction() {  
    console.log("Minha Função");  
}  
  
minhaFunction();
```

# Funções Javascript



Os parênteses podem incluir nomes de parâmetros separados por vírgulas:  
( **parâmetro1**, **parâmetro2**, ... )

```
function minhaFunction(parametro1, parametro2) {  
    return parametro1 * parametro2;  
}  
  
console.log(minhaFunction(4, 4));
```

16

> |

```
function minhaFunction(parametro1, parametro2) {  
    return parametro1 + parametro2;  
}  
  
console.log(minhaFunction(4, 4));
```

8

> |

# Funções Javascript



Como funciona exatamente o **return** Javascript

Define o valor retornado por uma função. Quando não há um valor especificado, **undefined** será retornado.

O código acima imprime **1** no console.

```
1
```

Retorno sem valor

```
function retornaVazio() {  
  return;  
}  
console.log(retornaVazio());
```

O código acima imprime **undefined** no console.

```
undefined
```

# Funções Javascript



Calcule o produto de dois números e retorne o resultado:

```
<div id="exemplo">
</div>

<script>
  var x = minhaFunction(4, 3);

  function minhaFunction(a, b) {
    return a * b;
  }

  document.getElementById("exemplo").innerHTML = x;
```

# Programa situação do aluno



Desenvolvendo HTML

```
<div class="container">
  <h1>Situação do Aluno</h1>
  <div>
    <label for="nota">Digite sua Nota: </label>
    <input class="form-control" type="text" id="nota" />
  </div>
  <br />
  <button onclick="situacao()">Verificar</button>
  <button onclick="limpar()">Limpar</button>
  <br /><br />
  <div id="resposta">
  </div>
</div>

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

# Programa situação do aluno



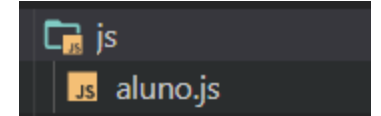
Criando a Função

```
function situacao() {  
  
}
```

Criando variável

```
var nota = document.getElementById("nota").value;
```

Chamada JS



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



# Programa situação do aluno



Criando condição

```
if(nota >= 6) {  
    document.getElementById("resposta").innerHTML = "Aluno aprovado";  
} else {  
    document.getElementById("resposta").innerHTML = "Aluno Reprovado";  
}
```

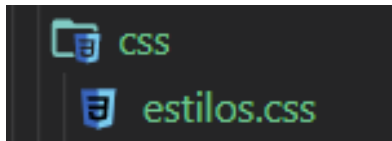
Criando função para Limpar campos

```
function limpar() {  
    document.getElementById("nota").value = "";  
}
```

# Melhorando o visual



Criando o **CSS** e chamando no **HTML**



```
<link rel="stylesheet" href="css/estilos.css" />
```

```
body {  
  background: #D3D3D3;  
}
```

# Melhorando o visual



Tratando elementos HTML

```
body {  
  background: #D3D3D3;  
  font-family: Arial, Helvetica, sans-serif;  
}  
  
.container {  
  width: 500px;  
  margin: 0 auto;  
  background: #fff;  
  padding: 15px;  
  text-align: center;  
  border: 2px solid #191979;  
}  
  
h1 {  
  color: #191970;  
  text-align: center;  
  margin-bottom: 20px;  
}
```

# Melhorando o visual



Aplicando estilo em botão

```
<button class="btn-verificar" onclick="situacao()">Verificar</button>  
<button class="btn-limpar" onclick="limpar()">Limpar</button>
```

```
.btn-verificar {  
  background: #191979;  
  padding: 10px;  
  color: #fff;  
  margin-right: 10px;  
  cursor: pointer;  
  font-weight: bold;  
  border: 0;  
}
```


```
.btn-limpar {  
  background: #FF0000;  
  padding: 10px;  
  color: #fff;  
  margin-right: 10px;  
  cursor: pointer;
```

# Melhorando o visual



Aplicando estilo na mensagem

```
<div id="resposta">  
</div>
```

```
#resposta {  
  color:  #8B0000;  
}
```

Aplicando função **window location**

```
window.location.reload();
```