



INTRODUÇÃO AO DESENVOLVIMENTO WEB

PROF^a. M.Sc. JULIANA H Q BENACCHIO

JavaScript

- JavaScript é uma linguagem de programação.
- Usada para criar pequenos “programas” para realizar ações em uma página web.
- Apesar do nome, JavaScript não está relacionada ao Java.

JavaScript

- O código JavaScript não é compilado.
- Deve ser colocado junto com os comandos HTML.
- O código fonte é chamado de *script*.

JavaScript

- Por que estudar JavaScript?
- Camadas:
 - **conteúdo** (HTML)
 - **formatação** (CSS)
 - **comportamento** (JavaScript)

- O que pode ser feito no JavaScript:
 - Animações
 - Validações
 - Manipulação de eventos do usuário
 - Efetuar cálculos
 - Componentes gráficos avançados
 - etc...

Criando um script

- Pode ser colocado em qualquer lugar dentro de uma página web.
- O mais comum é encontrarmos o código dentro da tag **<head>**.
- Para colocar código JavaScript, utilizamos a tag **<script>**
- Vejamos um exemplo:

Exemplo

- Uso da tag **<script>** para inserir código JavaScript numa página HTML:

```
<!DOCTYPE html>
<html>
<head>
  <title>Exemplo Javascript</title>
  <script>
    alert("Hello World!");
  </script>
</head>
<body>
</body>
</html>
```

Caixas de diálogos

- JavaScript disponibiliza 3 tipos de caixas de diálogos:
 - `alert()`
 - `prompt()`
 - `confirm()`

alert()

- Função do javascript que permite exibir uma mensagem ao usuário

```
<script>  
    alert("Hello World!");  
</script>
```

prompt ()

- Permite solicitar uma informação do usuário

```
<script>
```

```
    nome = window.prompt("Digite seu nome:");
```

```
    document.write(nome);
```

```
</script>
```

confirm()

- Esta caixa de diálogo possui dois botões. Um de OK e outro Cancelar. Pode-se usar esta função para solicitar uma decisão do usuário.

```
<script>
```

```
    opcao = confirm("Deseja prosseguir?");
```

```
    if(opcao)
```

```
        alert("Você clicou em OK");
```

```
    else
```

```
        alert("Você clicou em CANCELAR");
```

```
</script>
```

Variáveis

- Recurso do javascript que possibilita armazenar valores.
- Não é necessário declarar qual o tipo da variável.
- Exemplos

```
<script>  
    var numero = 100;  
    var fator = 5;  
    var result = numero * fator;  
</script>
```

Operadores aritméticos

- Operadores aritméticos:
 - + → Soma
 - - → Subtração
 - * → Multiplicação
 - / → Divisão
 - % → Resto da divisão

Operadores relacionais

- Os operadores relacionais são utilizados em expressões condicionais para a comparação do valor de duas expressões:
 - $>$ → Maior que
 - \geq → Maior ou igual à
 - $<$ → Menor que
 - \leq → Menor ou igual à
 - $==$ → Igual à
 - $!=$ → Diferente de

Operadores lógicos

- Os operadores lógicos são utilizados para conectar expressões lógicas sendo geralmente utilizados em expressões condicionais:
 - **&&** → AND (E lógico)
 - **||** → OR (OU lógico)
 - **!** → NOT (Operador de negação)

Estrutura condicional `if-else`

- No JavaScript uma decisão pode ser representada através da estrutura `if-else`. Por exemplo:

```
<script>
  var idade = eval(prompt("Qual a sua idade?"));
  if(idade>=18)
    document.write("<h2>Inscrição autorizada</h2>");
  else
    document.write("<h2>Não autorizado</h2>");
</script>
```

Condicional múltiplo `if-else-if`



- Em algumas situações pode-se ter um condicional múltiplo, onde várias condições precisam ser testadas, por exemplo:

```
<script>
  var idade = eval(prompt("Qual a sua idade?"));
  if(idade<16)
    document.write("<h2>Não eleitor</h2>");
  else if(idade>=16 && idade<18 || idade > 65)
    document.write("<h2>Eleitor facultativo</h2>");
  else
    document.write("<h2>Eleitor obrigatório</h2>");
</script>
```

Estrutura Condicional `switch-case`



- O comando **`switch-case`** é útil quando temos uma sequência longa de **`if-else`**'s.

```
<script>
    switch(n1) {
        case 1:
            document.write("Janeiro");
            break;
        case 2:
            document.write("Fevereiro");
            break;
        default:
            document.write("Mês inválido");
    }
</script>
```

Estrutura de Repetição for

- O comando **for** representa um mecanismo de repetição, onde é necessário definir 3 condições: (1) início da contagem; (2) condição de parada; (3) incremento/decremento

```
<script>
  for (i=0; i<10; i++){
    document.write("<h2>" + i + "</h2>");
  }
</script>
```

Estrutura de Repetição `while`

- O comando **`while`** representa um mecanismo de repetição, interpretado da seguinte maneira: Enquanto (condição) for verdadeira (faça):

```
<script>
  var i = 5;
  document.write("<b>iniciando while</b>");
  while(i>0){
    document.write("<br>em loop, na posição: " + i);
    i--;
  }
  document.write("<br><b>while finalizado</b>");
</script>
```