

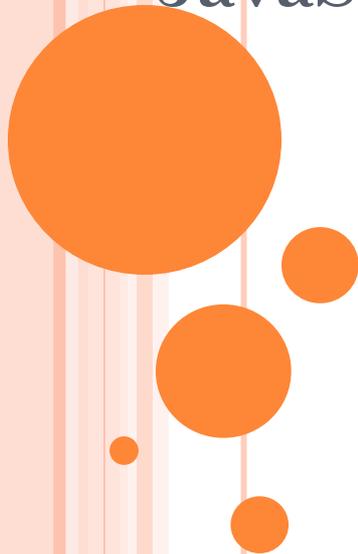
INSTITUTO FEDERAL DE EDUCAÇÃO CIÊNCIA E TECNOLOGIA
CURSO TÉCNICO INTEGRADO EM INFORMÁTICA
CAMPUS CURRAIS NOVOS

Desenvolvimento Web

JavaScript – aula I – *Introdução, inserção e variáveis*

Professor: Bruno Gomes

2010



INTRODUÇÃO

- Vimos anteriormente que a parte cliente de uma página *Web* é composta de basicamente três componentes
 - Conteúdo (informação)
 - Apresentação (visual, formatação e design – apresentação)
 - Comportamento (mudanças dinâmicas)
- Podemos dividir esses componentes em três tecnologias básicas:
 - *XHTML* (conteúdo)
 - *CSS* (apresentação)
 - *JavaScript* (comportamento)



INTRODUÇÃO

- Idealmente, esses três componentes devem estar separados
 - XHTML em um arquivo, fazendo referência a CSS e *JavaScript* externos
- Separação é importante para:
 - Reuso de código
 - Atender a uma gama maior de plataformas de exibição e políticas de empresa
 - Navegador pode não suportar CSS ou *JavaScript*
 - Usuários podem preferir desativar CSS ou *JavaScript*
 - Facilitar a manutenção da página *Web*
 - Designer cuida de CSS, imagens, etc.
 - Programador mantém o *JavaScript*



INTRODUÇÃO - JAVASCRIPT

- *JavaScript* é uma linguagem interpretada
 - Não necessita de compilação para executar programas
 - Navegador precisa conter apenas um programa interpretador de *JavaScript*
- Desenvolvida por Brendan Eich, em 1995, para o *Netscape Navigator (2.0)*
 - Posteriormente incorporado ao *Internet Explorer* com o nome de *JScript*
- *JavaScript* não é *Java*!
 - *Sintaxe* parecida, vinda de C/C++, mas com origens distintas
 - *Java* é uma linguagem bem mais poderosa e complexa



INTRODUÇÃO – JAVASCRIPT

- Atualmente, a chamada “Web 2.0” faz uso intenso da linguagem através da tecnologia AJAX (*Asynchronous Javascript and XML*)
- Algumas características
 - Interpretada
 - Sintaxe / semântica semelhante à C
 - Baseada em objetos
 - Sensível à caixa
 - Palavras escritas com letra minúscula são diferentes de palavras escritas com letras maiúsculas
 - Palavras-chave da linguagem, como *if* e *switch*, são sempre escritas com letras minúsculas
- **Atenção!** Navegadores mais antigos ou de modo texto podem não suportar *JavaScript*



INTRODUÇÃO - JAVASCRIPT

- Devemos ter o cuidado de escrever código *JavaScript*:
 - Sem “congestionar” o documento XHTML
 - Sem deixar de fora usuários que estão com *JavaScript* desativado
 - Sem interferir em outros códigos *JavaScript* que estejam sendo aplicados na mesma página



O QUE PODEMOS FAZER COM JAVASCRIPT ?

- **JavaScript oferece uma linguagem de programação para interagir com XHTML**
 - Linguagem simples e de fácil aprendizado
- **Colocar texto dinâmico em uma página**
 - Exemplo: `document.write("<h1>" + name + "</h1>")` pode escrever o valor de uma variável em uma página HTML.
- **Reagir a “eventos”**
 - Executa quando algo está para acontecer:
 - uma página termina de carregar,
 - um usuário clica em algum elemento da página, etc.
- **Ler e escrever elementos XHTML**
 - Por exemplo, para mudar o conteúdo de um elemento



O QUE PODEMOS FAZER COM JAVASCRIPT ?

○ Validar dados de formulário

- Antes de enviar os dados do formulário para o servidor, *JavaScript* pode ser utilizado para validar os campos do formulário, livrando o servidor de processamento extra.

○ Detectar o navegador do usuário

- Permite corrigir apresentação específica para aquele navegador ou carregar outra página que funcione melhor com o navegador.

○ Criar *cookies*

- Armazena e recupera informações salvas no computador do usuário.



JAVASCRIPT – PRIMEIRO EXEMPLO

```
<!DOCTYPE html PUBLIC "-//W3C//DTD XHTML 1.0 Strict//EN"
  "http://www.w3.org/TR/xhtml1/DTD/xhtml1-strict.dtd">
<html xmlns="http://www.w3.org/1999/xhtml">
  <head>
    <title>Escrevendo em XHTML com JavaScript</title>
    <script type="text/javascript">
      <!--
        document.writeln("<h1>Bem vindo à programação com
                          JavaScript!</h1>");
      // -->
    </script>
  </head>
  <body> ...
  </body>
</html>
```

JAVASCRIPT

- No caso do exemplo anterior, introduzimos o *JavaScript* na seção *<head>* do documento, utilizando a tag *<script>*
 - Neste caso, o conteúdo *JavaScript* é posto no *body*, mas antes do conteúdo do *body*
 - Objeto *document* representa o documento XHTML que o navegador está atualmente exibindo
 - Método *writeln* escreve a string recebida como argumento no documento XHTML
 - Comentários (*<!-- // -->*) foram adicionados para que navegadores antigos, que não tem suporte a *JavaScript*, ignorem o código
 - Do contrário, o código *JavaScript* apareceria na página
- 

JAVASCRIPT – INSTRUÇÕES

```
<script type="text/javascript">  
  <!--  
    document.writeln("<h1>Bem-vindo à programação com  
                          JavaScript!</h1>");  
  // -->  
</script>
```

A parte:

```
document.writeln("<h1>Bem vindo à programação com  
JavaScript!</h1>");
```

é uma instrução *JavaScript*

- *Uma instrução deve terminar (embora não seja obrigatório) com um “;” (ponto-e-vírgula)*

MODOS DE INSERÇÃO DE JAVASCRIPT

- Assim como fizemos para CSS, temos 3 formas de inserir código *JavaScript* nas nossas páginas
 - Como vimos anteriormente, na seção *<head>* do documento, utilizando a tag *<script>*
 - Em um arquivo externo ao XHTML
 - De modo *inline*, dentro de alguma *tag* do *XHTML*

- É preferível incluí-lo em um arquivo externo
 - Separação clara entre conteúdo e comportamento
 - Facilidade de manutenção do *Web site*
 - Reutilização do(s) mesmo(s) programas *JavaScript* em diferentes páginas do seu *Web site*



INSERINDO JAVASCRIPT EM UM ARQUIVO EXTERNO

```
<head>  
  <title>Java Script externo</title>  
  
  <script type="text/javascript" src="exemplo.js">  
  </script>  
  
</head>
```

- Em **src** colocamos um nome ou caminho de arquivo
- Pode ser até mesmo uma “url” completa para um arquivo *JavaScript* armazenado em um servidor externo

COMENTÁRIOS

- São partes do código que são desconsideradas pelo interpretador
 - Utilizados normalmente para fins de documentação
- Comentário de uma única linha
 - //
- Comentário de múltiplas linhas
 - Começa com `/*` e termina com `*/`



VARIÁVEIS

- Variáveis são declaradas utilizando-se a palavra-chave **var**
 - Ex.: **var** numero = 0;
var cor = "red";
- Uso de **var** não é obrigatório, mas quando não usamos estamos declarando uma *variável global*
 - numero = 0 //número é uma variável global



TIPAGEM DAS VARIÁVEIS É DINÂMICA

- Tipagem das variáveis é dinâmica (linguagem fracamente tipada)
 - Você não precisa declarar o tipo da variável explicitamente
 - A atribuição de um valor à variável define seu tipo
- Assim, embora não seja uma boa prática, uma mesma variável pode receber diferentes tipos no decorrer da execução do programa

- Exemplo:

//JavaScript

```
var a;
```

```
a = "Bruno"; // "a" é "String"
```

```
a = 10; /*OK, "a" agora é  
inteiro*/
```

//Java

```
String a;
```

```
a = "Bruno";
```

```
a = 10 //Erro!
```



TIPOS DE VARIÁVEIS EM JAVASCRIPT

- Números (Inteiros e ponto-flutuante)
- Strings
- Booleanos
- *Arrays*
- Objetos



TIPOS DE VARIÁVEIS – NÚMEROS

- Números (Inteiros e ponto flutuante)
 - **var** num = 0;
 - **var** nota1, nota2, nota3 = 0;
 - **var** total = 245.89;
 - **var** res = 4 - 12 + a;
- Valores máximos suportados do tipo numérico:
 - +/-1.7976931348623157e+308
- Valores podem ser obtidos com:
 - *Number.MAX_VALUE*
 - *Number.MIN_VALUE*



ALGUMAS FUNÇÕES DA BIBLIOTECA MATEMÁTICA (MATH)

- Para usar: ***Math.<nome_da_função>***
- **abs(x)** *Valor absoluto de "x"*
- **ceil(x)** *Arredonda o valor de "x" para o inteiro mais próximo acima de "x"*
- **exp(x)** *exponencial de "x"*
- **floor(x)** *Arredonda "x" para o inteiro mais próximo acima de "x"*
- **log(x)** *Logaritmo natural de "x"*
- **max(x,y,z,...,n)** *Retorna o maior número da lista*
- **min(x,y,z,...,n)** *Retorna o menor número da lista*
- **pow(x,y)** *Potência de "x" na base "y"*
- **random()** *Gera número aleatório entre 0 e 1*
- **round(x)** *Arredonda x para o inteiro mais próximo*
- **sqrt(x)** *Raiz quadrada de "x"*

