

Introdução à Tecnologia Web

JavaScript Histórico e Características

Índice

1 – JAVASCRIPT.....	2
1.1 – Histórico.....	2
1.2 – Aplicações de JavaScript.....	2
a) Interatividade.....	2
b) Validação de formulários.....	2
c) Controle de comportamento da página.....	2
d) Personalização de aparência da página pelo usuário.....	2
e) Dinamismo de conteúdo.....	3
1.3 – Características da linguagem.....	3
a) Client-side.....	3
b) Interpretada.....	3
c) Case-sensitive.....	3
d) Fracamente tipada.....	3
e) Dinamicamente tipada.....	3
f) Estruturada ou Orientada à Objetos.....	3
g) Acionamento.....	4
Notas deste capítulo.....	4
Referências Bibliográficas.....	4

1 - JAVASCRIPT

JavaScript é uma linguagem de programação utilizada para criar pequenos programas (scripts) encarregados de realizar ações dentro de uma página Web.

A linguagem base para criação de *websites* seguindo os padrões W3C é sempre a HTML/XHTML (Hypertext Markup Language/ eXtensible Hypertext Markup Language). Contudo, esta linguagem define apenas a estrutura e os elementos que compõem cada página de um site, sendo uma linguagem limitada. Para configurar uma aparência diferenciada a um site, complementamos o código com a linguagem de estilos CSS (Cascading Style Sheets). Da mesma forma, podemos enriquecer o site com “funcionalidades” que não podem ser criadas com HTML e nem com CSS. Para tal, necessitamos outras linguagens classificadas como linguagens de programação, e é neste caso que entra JavaScript.

1.1 - Histórico

JavaScript foi criada pela Netscape em 1995 com o nome de LiveScript, em uma sequência de soluções criadas para suprir as limitações da HTML:

1ª Solução: Applets Java da Sun Microsystems (mini aplicativos escritos em Linguagem Java)

2ª Solução: LiveScript da Netscape

3ª Solução: JavaScript parceria da Netscape com a Sun, rebatizando o LiveScript em uma jogada comercial para aproveitar o sucesso da então linguagem Java da Sun. Mas não teria dado certo se JavaScript não fosse uma ótima linguagem script para páginas web.

1.2 - Aplicações de JavaScript

O objetivo dos scripts é oferecer funcionalidades que adicionem às páginas:

a) Interatividade

Interatividade é processo de interação entre pessoas e sistemas. A interação ocorre quando o usuário ou o sistema dispara uma ação e a outra parte apresenta uma reação/resposta que modifica o estado atual do sistema ou que pode desencadear uma sequência de novas interações. Esse processo também se dá em *websites*. Exemplos são: calculadoras, agendas, jogos, personalização da aparência das páginas pelos usuários, etc.

b) Validação de formulários

É o processo de verificação dos dados fornecidos pelos usuários através de formulários. A verificação ocorre ainda no navegador, antes do envio dos dados ao servidor para processamento/armazenamento. Esse processo, quando efetuado através de scripts client-side tornam a navegação mais eficiente, pois o usuário corrige os dados não aprovados no mesmo momento sem recarregar o formulário vazio.

c) Controle de comportamento da página

O comportamento dos elementos da página podem ser controlados com scripts. Pode-se definir definem para diversos elementos qual comportamento apresentar de acordo com alguma ação do usuário (clique, passar o mouse, digitar, selecionar, etc). Este controle pode ser feito pelo uso de DHTML (Dynamic HTML), que é um padrão de desenvolvimento com a combinação de HTML, Javascript, CSS e DOM (Document Object Model).

d) Personalização de aparência da página pelo usuário

Permite que um usuário possa mudar características de formatação como a cor e tamanho dos elementos textuais, de modo que fique mais agradável a ele. Também feita pelo uso de DHTML.

e) Dinamismo de conteúdo

Consiste em atualizar o conteúdo de partes específicas da página sem o recarregamento da página toda (*reload*) pelo navegador. A atualização é feita quase sem que o usuário perceba, e torna a navegação mais eficiente. Consegue-se esta característica com a técnica de programação AJAX (Asynchronous JavaScript and XML), que une JavaScript, HTML/XHTML, CSS, XML/JSON em conjunto com outra linguagem/tecnologia server-side (PHP, JSP, ASP, etc) e banco de dados.

1.3 – Características da linguagem

a) Client-side

Linguagem que roda na máquina cliente (lado cliente). O próprio navegador do cliente é o encarregado de interpretar as instruções JavaScript e executá-las para realizar as operações programadas.

b) Interpretada

JavaScript não necessita ser compilada, pois é uma linguagem interpretada diretamente pelo navegador.

c) Case-sensitive

Em JavaScript se deve respeitar as maiúsculas e as minúsculas. Ex:

Nome é diferente de **nome**.

numero é diferente de **Numero**.

For é diferente de **for**.

d) Fracamente tipada

JavaScript é chamada fracamente tipada pois ao declarar as variáveis não definimos qual o tipo da variável (inteira, ponto flutuante, cadeia, booleana, etc). Assim, podemos armazenar em uma variável a informação que desejarmos, pois não tem tipo “definido” na declaração.

Ex:

```
string nome = "Maria";  
var nome;           // apenas declaração  
var nome = "Maria"; // declaração com atribuição  
nome = "Maria";    // declaração com atribuição
```

e) Dinamicamente tipada

Toda linguagem fracamente tipada é também dinamicamente tipada. Isso significa que a variável vai se tornar de um determinado tipo no momento da atribuição de conteúdo. Se o tipo de conteúdo mudar durante a execução do programa a variável vai mudar de tipo dinamicamente.

Ex:

```
nome = "João";    (variável nome não tem tipo definido, mas recebeu conteúdo do tipo string)  
nome = 10;       (agora a variável nome possui conteúdo do tipo numérico)
```

f) Estruturada ou Orientada à Objetos

Originalmente a linguagem foi criada seguindo o paradigma de programação Estruturada, mas também pode ser programada no paradigma Orientado à Objetos (OO).

g) Acionamento

Scripts podem ser acionados por eventos (passar o mouse, clicar, tempo, etc) ou automaticamente na sequência em que o navegador interpreta o conteúdo da página e localiza a linha de comando.

Notas deste capítulo

1 - JavaScript não pode ser confundida com Java, apesar da semelhança no nome. Javascript tem nome e sintaxe semelhante à linguagem Java, mas é totalmente diferente no conceito e no uso.

2 - Procure padronizar os nomes das variáveis e nomes de funções. Utilize sempre letras minúsculas, iniciando com letras minúsculas apenas a segunda palavra que compõe o nome. Observe o padrão:

nome; nomeCompleto; numero; primeiroNumero;
soma() somaNum() validaEmail()

Outras Informações

Além de uso em navegadores processando páginas HTML, o JavaScript é hoje usado também na construção do navegador Mozilla Firefox.

Referências Bibliográficas

1. CARDOSO, Márdel. *Desenvolvimento web para o ensino superior*. Rio de Janeiro: Axcel Books, 2004.
2. DAMIANI, Edgard. *Javascript - Guia de Consulta Rápida*. São Paulo: Novatec, 2001.
3. FREEMAN, Eric, FREEMAN, Elisabeth. *Use a Cabeça: HTML com CSS e XHTML*. Rio de Janeiro: Alta Books, 2008.
4. GOODMAN, Danny. *Javascript : a Bíblia*. Rio de Janeiro: Campus, 2001.
5. GOODMAN, Danny. *Javascript e DHTML: Guia Prático*. Rio de Janeiro: Campus, 2008.
6. HTML. *W3C Recommendation: HTML 4.01 Specification*. Disponível online em: [<http://www.w3.org/TR/html4/>]
7. MACEDO, Marcelo da Silva. *Construindo sites adotando padrões Web*. Rio de Janeiro: Ciência Moderna, 2004.
8. MARCONDES, Christian Alfim. *HTML fundamental 4.0*. São Paulo: Érica, 2005.
9. MORRISON, Michael. *Use a Cabeça: Javascript*. Rio de Janeiro: Alta Books, 2008.
10. NEGRINO, Tom; SMITH, Dori. *Javascript para a World Wide Web*. Rio de Janeiro: Campus, 2001.
11. SILVA, Maurício Samy. *JavaScript - Guia do programador*. São Paulo: Novatec, 2010.
12. W3C. *JavaScript-Based Style Sheets*. Disponível online em: [<http://www.w3.org/Submission/1996/1/WD-jsss-960822>]
13. W3C. *Scripts*. Disponível online em: [<http://www.w3.org/TR/REC-html40/interact/scripts.html>]
14. W3C. *World Wide Web Consortium*. Disponível online em: [<http://www.w3.org/>]