Programação Cliente-servidor

Aula 5: XML e AJAX

XML

XML não é uma linguagem, mas um conjunto de regras de escrita, ou de maneira formal. seu objetivo primordial é facilitar o compartilhamento de informações na Internet

A interoperabilidade pode ser entendida como uma característica que se refere à capacidade de diversos sistemas e organizações trabalharem em conjunto

Ao criarmos um documento XML, devemos seguir algumas regras:

* Sempre deve existir um e apenas um nó raiz
* Toda etiqueta precisa ter fechamento
* Os comandos, ou etiquetas, devem estar corretamente aninhados
* Os atributos de cada etiqueta não podem apresentar repetições
* Ocorre a diferenciação entre elementos em maiúsculo ou minúsculo (case-sensitive)

Exemplo:

<?xml version="1.0" encoding="iso-8859-1"?>

<turma codigo="1001">

<professor>Tom Sawyer</professor>

<alunos>

<aluno matricula="1968001">Gleedy Lee</aluno>

<aluno matricula="1968002">Alex Lifeson</aluno>

<aluno matricula="1968003">Neil Peart</aluno>

</alunos>

</turma>

Adição de um arquivo css ao xml

<?xml-stylesheet type="text/css" href="turma.css"?>

Esquemas

Em termos de XML, os esquemas podem ser definidos com o uso de dois tipos de tecnologia:

Document Type Definition (DTD)2

O modelo de gramática proporcionado pelo DTD é bastante simples, mas não utiliza XML em sua definição e não permite o uso de namespaces, o que significa dizer que não podemos ter gramáticas múltiplas com uso de DTD.

nós já utilizamos uma DTD ao colocar <!DOCTYPE html> no início de nossas páginas HTML

Exemplo:

<!DOCTYPE colaborador[

<!ELEMENT colaborador(nome, matricula)>

<!ELEMENT nome (#PCDATA)>

<!ELEMENT matricula (#PCDATA)>

<!ATTLIST matricula uf CDATA #REQUIRED>

]>

Note que são definidos elementos com o uso de ELEMENT e atributos com o uso de ATTLIST.

linguagem de transformação

s documentos XML permitem a organização da informação, mas não estão preocupados com o aspecto visual desta informação, pois é papel do HTML, junto ao CSS. É neste ponto que surgem linguagens específicas para este fim, as chamadas “linguagens de transformação”.

A linguagem de transformação mais utilizada para o ambiente que envolve XML é denominada eXtensible Stylesheet Language (XSLT). Esta é uma linguagem com sintaxe XML para a transformação de arquivos XML em outros formatos, como SVG, VRML, HTML, ou até outros arquivos XML.

Podemos observar, a seguir, um arquivo que será nomeado “**contatos.xsl**”.

Neste código podemos observar várias das características do XSLT, a começar pela utilização do namespace correto para o reconhecimento dos comandos da linguagem.

<xsl:stylesheet xmlns:xsl="//www.w3.org/1999/XSL/Transform" version="1.0">

A partir deste ponto podemos definir templates aplicáveis a determinadas tags do XML, ou à raiz do documento.

<xsl:template match="/">

Neste caso, ao encontrar a raiz, será iniciada a construção do HTML contendo uma tabela no corpo do documento, sendo criada uma linha para cada contato encontrado no documento XML com o uso de xsl:for-each.

<xsl:for-each select="contatos/contato">

Os valores a serem utilizados na saída são obtidos com o comando xsl:value-of. Este comando aceita apresentar o valor de texto de um elemento interno diretamente, de atributos com uso de arroba, ou o texto do elemento corrente com uso de ponto.

<xsl:value-of select="@nome"/>

Note que o arquivo XML referencia o XSL como uma folha de estilo.

<?xml-stylesheet type="text/xsl" href="contatos.xsl"?>

Leitura com DOM

Com o uso de JavaScript é muito simples tratar os arquivos XML, pois o uso do **parser** segue os mesmos princípios de acesso aos elementos do HTML.

Em seguida instanciamos um objeto do tipo DOMParser. Este objeto será responsável pela criação da árvore DOM a partir do conteúdo XML.

parser = new DOMParser();

A árvore é montada em memória e colocada na variável xmlDoc com o uso do parser.

<xsl:for-each select="contatos/contato">

A partir daí é possível acessar os elementos do XML como qualquer árvore DOM, e com o uso de getElementsByTagName recebemos todas as tags com o nome especificado, sendo recebidas em um vetor com índice inicial zero.

var noInfo = xmlDoc.getElementsByTagName("nome")[0]

Porém, este nó será do tipo elemento; para acessar o texto, precisamos acessar o primeiro filho dele na árvore, correspondendo a um nó do tipo texto, com a propriedade nodeValue informando o valor deste texto.

noInfo.childNodes[0].nodeValue;

Tecnologia AJAX

Uso de XML na Web

O formato XML é amplamente utilizado na Web em comunicações entre empresas (B2B), principalmente devido ao formalismo proporcionado pelos esquemas XSD e DTD, o que garante a recepção de dados com formatos e tipos previamente estipulados.

A definição de esquemas pode ser feita com o uso de DTD, que não segue a sintaxe XML e não permite namespaces nem a definição de tipos, ou XSD, que utiliza XML e namespaces, além de permitir a definição de tipos.

|  |  |  |
| --- | --- | --- |
| |  | | --- | | Analise as seguintes afirmativas acerca da sintaxe JSON e assinale a opção correta.  I - Valores numéricos devem ser colocados entre aspas.  II - Valores nulos são representados por null.  III - Vetores são definidos com o uso de chaves. | | |
|  | | |

|  |  |  |
| --- | --- | --- |
|  |  | Todas as afirmativas são falsas. |
|  |  | Apenas as afirmativas II e III são verdadeiras. |
|  |  | Todas as afirmativas são verdadeiras. |
|  |  | Apenas as afirmativas I e III são verdadeiras. |
| Certo |  | Apenas a afirmativa II é verdadeira. |

|  |  |  |
| --- | --- | --- |
| |  | | --- | | No que tange ao JSON (Java Script Object Notation), leia as asserções abaixo e, a seguir, assinale a alternativa correta.  I. Várias tecnologias atuais utilizam o JSON como formato texto para troca de informações;  II. O Web Services REST passou a ter uso restrito em função das plataformas móveis, como Android e iOS.  III. A definição do objeto JSON é iniciada com o uso de chaves, e os pares chave-valor são separados por vírgula. | | |
|  | | |

|  |  |  |
| --- | --- | --- |
| Certo |  | Somente as asserções I e III estão corretas |
|  |  | Somente a asserção I está correta |
|  |  | Somente a asserção II está correta |
|  |  | Somente as asserções II e III estão corretas |
|  |  | Somente as asserções I e II estão corretas |

|  |  |  |
| --- | --- | --- |
| |  | | --- | | No uso do JQuery, o seletor $("li:odd") serviria para selecionar: | | |
|  | | |

|  |  |  |
| --- | --- | --- |
| Certo |  | Todas as ocorrências ímpares de itens de lista. |
|  |  | O item central de cada lista da página. |
|  |  | O último item de cada lista. |
|  |  | O primeiro item de cada lista. |
|  |  | Todas as ocorrências pares de itens de lista. |

|  |  |  |
| --- | --- | --- |
| |  | | --- | | Ao implementar uma página, com uso de JQuery, deseja-se que os campos de texto fiquem com o fundo amarelo quando  ganharem o foco, branco quando perderem o foco e laranja quando o conteúdo for modificado. Quais métodos devem ser  utilizados, respectivamente? | | |
|  | | |

|  |  |  |
| --- | --- | --- |
|  |  | mouseenter, mouseleave e change |
|  |  | keydown, keyup e keypress |
|  |  | mouseenter, mouseleave e click |
| Certo |  | focus, blur e change |
|  |  | focus, blur e click |

|  |  |  |
| --- | --- | --- |
| |  | | --- | | Diversos efeitos são disponibilizados com o uso de JQuery. Qual deles faria com que uma DIV desaparecesse "lentamente"? | | |
|  | | |

|  |  |  |
| --- | --- | --- |
|  |  | shake |
|  |  | show |
| Certo |  | fadeOut |
|  |  | hide |
|  |  | fadeIn |

|  |  |
| --- | --- |
| Explicação:  O efeito fadeOut apaga lentamente o componente, enquanto fadeIn exibe lentamente, show exibe instantaneamente, hide apaga instantaneamente, e shake faz ele "tremer". | |

|  |  |  |
| --- | --- | --- |
| |  | | --- | | De modo geral, o XML parser nada mais é do que: | | |
|  | | |

|  |  |  |
| --- | --- | --- |
|  |  | Uma técnica utilizada para a manipulação de metadados. |
|  |  | Um objeto DOM implementado dentro do XML. |
|  |  | Uma classe de documentos criados em JavaScript. |
|  |  | Um método utilizado exclusivamente na validação de documento JavaScript. |
| Certo |  | Um analisador de documentos XML, que tem como objetivo carregar ou criar um documento utilizando o JavaScript. |

|  |  |  |
| --- | --- | --- |
| |  | | --- | | Um esquema define uma gramática para determinado domínio do problema, e a sintaxe XML permite duas formas para a definição de esquemas, que são: | | |
|  | | |

|  |  |  |
| --- | --- | --- |
|  |  | XSD e XSL-FO |
|  |  | XSLT e XSL-FO |
| Certo |  | DTD e XSD |
|  |  | DTD e XSLT |
|  |  | XSD e XSLT |

|  |  |  |
| --- | --- | --- |
| |  | | --- | | "Em aplicativos Android, um recurso extremamente utilizado é o de (I). Basta você baixar alguns aplicativos na Google Play Store que os encontrará facilmente. Embora sejam de fácil implementação, devemos reforçar os cuidados em relação à usabilidade e aparência dos mesmos."  A sentença acima refere-se a: (assinale a alternativa que preencher corretamente a (I)). | | |
|  | | |

|  |  |  |
| --- | --- | --- |
|  |  | Botão |
|  |  | Link |
|  |  | Ícone |
| Certo |  | Menu |
|  |  | Caixa de texto |

|  |  |  |
| --- | --- | --- |
| |  | | --- | | Em aplicativos Android, um recurso extremamente utilizado é o de menu. Basta você baixar alguns  aplicativos na Google Play Store que os encontrará facilmente.  Podemos trabalhar com três tipos de menus em Android, os quais:  I. Menu de opção e barra de opção  II. Menu de texto  III. Menu pop-up  Após a leitura, assinale a alternativa correta. | | |
|  | | |

|  |  |  |
| --- | --- | --- |
|  |  | Apenas a afirmação I está correta |
|  |  | Apenas a afirmação II está correta |
| Certo |  | Apenas as afirmações I e III estão corretas |
|  |  | Apenas as afirmações II e III estão corretas |
|  |  | Apenas as afirmações I e II estão corretas |

|  |  |  |
| --- | --- | --- |
| |  | | --- | | No que tange ao conceito de fragmento, analise as asserções abaixo e, a seguir, assinale a alternativa correta:  I. Bastante similar a uma Activity, um fragmento (*fragment*) consiste em uma pequena porção de Activty, que permite um projeto mais modular.  II. Pode-se afirmar que um fragmento é uma espécie de *activity*.  III. O conceito de fragmento surgiu com o Android 3.0 (*Honeycomb*) devido à necessidade de customizá-lo para as interfaces dos aplicativos, em função da pluralidade de tipos e tamanhos de dispositivos, em especial os *tablets*. | | |
|  | | |

|  |  |  |
| --- | --- | --- |
|  |  | Somente a asserção I está correta |
|  |  | Somente as asserções I e II estão corretas |
| Certo |  | Somente as asserções I e III estão corretas |
|  |  | Somente a asserção II está correta |
|  |  | Somente as asserções II e III estão corretas |

|  |  |  |
| --- | --- | --- |
| |  | | --- | | Ajax não é meramente uma tecnologia. É uma abordagem moderna para desenvolvimento de sites interativos. A abordagem de desenvolvimento tradicional tem semelhanças e diferenças em relação ao Ajax. Uma característica exclusiva de Ajax em relação à abordagem tradicional é que: | | |
|  | | |

|  |  |  |
| --- | --- | --- |
|  |  | Usa javascript como linguagem para desenvolver código no lado do cliente. |
|  |  | Executa as requisições através do protocolo HTTP. |
|  |  | Usa (x)html / css para definir o aspecto visual da página. |
|  |  | Representa os objetos no lado cliente com DOM. |
| Certo |  | Permite recuperação assíncrona de dados usando XMLHttpRequest. |