

# **Conceitos introdutórios**

## **DOM - XML**

# DOM

O DOM define um padrão de acesso e manipulação de documentos:

De acordo com o [W3C](#), Document Object Model (DOM) é uma interface de plataforma e linguagem neutra que permite que programas e scripts acessar e atualizar dinamicamente o conteúdo, estrutura e estilo de um documento."

O HTML DOM define um padrão para acessar e manipular documentos HTML. Ele apresenta um documento HTML como uma estrutura de árvore.

O DOM XML define um padrão de acesso e manipulação de documentos XML. Ele apresenta um documento XML como uma estrutura de árvore.

# JSON

Além da terminação .json em todos os arquivos que utilizam esse formato, os dados armazenados devem seguir uma notação específica, ou seja, precisam ser organizados com os seguintes elementos básicos:

chaves { } para delimitar os objetos e obrigatórias para iniciar e encerrar o conteúdo;

colchetes [ ] para indicar um array;

dois pontos : para separar a chave e seu valor correspondente;

vírgula , para indicar a separação entre os elementos.

Exemplos de como os dados devem ser relacionados em um arquivo .json.

String

```
{ "distrito": "Lisboa" }
```

Array

```
{  
  "distritos": ["Lisboa", "Setubal", "Evora"]  
}
```

# JSON

## Objeto

```
{ "distrito": {  
    "distrito": "Lisboa",  
    "sigla": "LIS"  
}  
}
```

## Lista de objetos

Confira como fazer a notação para indicar uma lista de objetos:

```
{ "distritos":[  
    {"distrito": "Lisboa", "sigla": "LIS"},  
    {"distrito": "Setubal", "sigla": "STB"},  
    {"distrito": "Evora", "sigla": "EVA"}  
]  
}
```

# JSON

Quais as diferenças entre .json e .xml?

Outro formato utilizado para a troca de dados entre aplicações é o XML — eXtensible Markup Language. Apesar de também ser um arquivo de texto, existem algumas diferenças entre os dois modelos. Confira as principais.

## **Notação**

A primeira diferença entre os dois modelos é a forma de fazer a notação dos dados. Conforme mencionamos, o JSON utiliza uma notação simples, enquanto o XML utiliza uma estrutura de tags personalizadas para representar os objetos. Além disso, elas devem conter o par, ou seja, a tag de abertura e a de fechamento.

Outra característica da notação XML é que o seu conteúdo não precisa ser delimitado com aspas, como acontece com os textos no formato JSON. Nele, o que indica o início e o fim das informações são as tags de abertura e fechamento.

# O QUE É XML?

- Tecnologia desenvolvida pelo W3C

<http://www.w3c.org>

- *W3C: World Wide Web Consortium*
  - consórcio formado por acadêmicos e empresários
  - definição de padrões para a Web



<http://www.w3.org/standards/xml/>

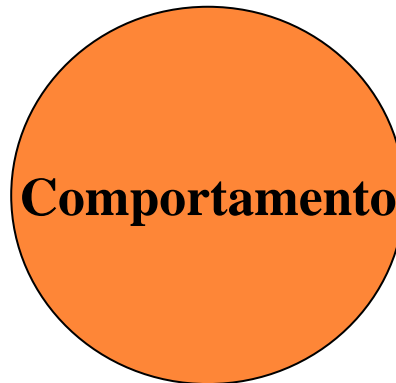
# Trio dos Padrões *Web*



**HTML**  
**XHTML**  
**XML**



**CSS 1**  
**CSS 2**  
**CSS 3**



**ECMAScript**  
**DOM**

# Trio dos Padrões *Web*

## ○ XML

- Foi projetado para **transportar** e **armazenar** dados

## ○ HTML

- Foi projetado para **mostrar** os dados
- Trabalha em conjunto com o CSS



# O QUE É XML?

- XML significa *eXtensible Markup Language*.
- XML é uma linguagem de marcação muito parecido com HTML.
- XML foi projetado para transportar dados, não para exibir dados.
- ***Tags XML não são predefinidas. Você deve definir suas próprias tags.***
- **XML é projetado para ser autodescritivo.**
- XML é uma recomendação da W3C.

# O QUE É XML?

## ○ A diferença entre XML e HTML:

- XML não é um substituto para HTML.
- XML e HTML foram projetados com objetivos diferentes:
  - XML foi projetado para transportar e armazenar dados, com foco no que os dados são;
  - HTML foi projetado para exibir dados, com foco em como os dados aparecem.
- HTML é para a exibição de informações, enquanto o XML é para carregar /transportar informações.

# O QUE É XML?

- XML é uma **meta linguagem** de **marcação**
  - **meta linguagem**
    - XML é um **padrão aberto**
      - cada aplicação define o protocolo (linguagem) para a representação dos seus dados
  - **linguagem de marcação**
    - semelhante ao que ocorre com a linguagem HTML
      - utiliza *tags* para descrição dos dados
  - HTML x XML
    - **tags em HTML**: são predefinidas e voltadas para a formatação da apresentação de dados em *browsers Web*
    - **tags em XML**: não são predefinidas (dependem da intenção da aplicação); indicam a **intenção** do dado e delimitam o seu **conteúdo**

# O QUE É XML?

## ○ XML não faz nada

- Talvez seja um pouco difícil de entender, mas XML não faz nada.
- XML foi criado para **estruturar**, **armazenar** e **transportar** a informação.

# O QUE É XML?

- O exemplo abaixo é uma mensagem **para Sérgio**, escrita **por Juliano**, **armazenada** como XML:

```
<note>
  <to>Joao</to>
  <from>Jose</from>
  <heading>Lembre-se</heading>
  <body>Não esqueça o exercício da semana!</body>
</note>
```

- A mensagem é bastante autodescritiva.
  - Tem emissor e receptor de informações, um título e um corpo de mensagem.
- Mas, ainda assim, este documento XML não faz nada.
  - É apenas informação envolta em *tags*.
  - Alguém deve escrever um trecho de *software* para enviar, receber ou exibi-la.

# EXEMPLO DE DADO XML

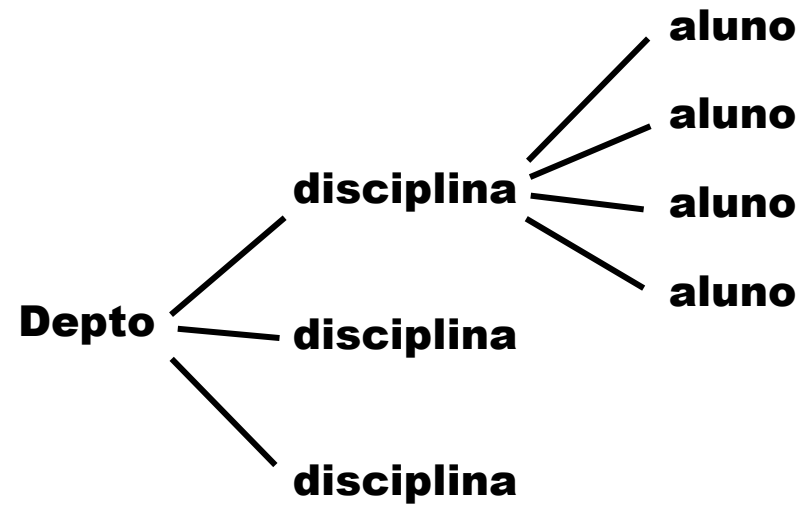
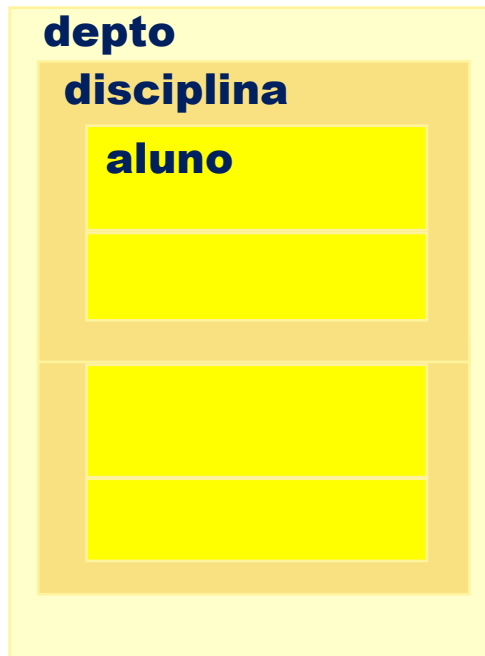
```
<livro>
  <titulo>Tecnologia XML</titulo>
  <autor>
    <nome>João da Silva</nome>
    <eMail>js@hotmail.com</eMail>
    <endereco>
      <comercial>rua A, 34 - Lisbon - PT</comercial>
      <residencial>rua B, 5 - Lisbon - PT</residencial>
    </endereco>
  </autor>
  ...
  <capitulo nome="Introdução">Este capítulo apresenta
  ...
    <secao>
      <nome>Linguagens de Marcação</nome>
      ...
    </secao>
  </capitulo>
  ...
</livro>
```

← **tag** (intenção do dado)

← **conteúdo do dado**

→ **estrutura hierárquica, ordenada e complexa**

# ESTRUTURA HIERÁRQUICA



# UTILIZAÇÃO DE XML

## ○ Transferência de dados

- entre BDs
- entre dispositivos móveis conectados à Internet
  - exemplo: celulares
- entre aplicações distribuídas de modo geral

## ○ Aplicações *Web*

- exemplo: aplicações B2B
  - "protocolos" XML para dados e transações de negócio

## ○ Integração de dados

- representação padrão para fontes heterogêneas de dados
  - tabelas de BDs, documentos, arquivos, ...



# Uso EXTENSIVO DE XML...

- Problemas a serem resolvidos
  - projeto de uma estrutura para dados em XML
  - métodos de acesso a dados XML
  - estratégias para armazenamento e indexação de dados XML
  - ...
- A tecnologia de BD é útil neste contexto
  - nova linha de pesquisa: BD XML Nativo
    - definem um modelo lógico para dados XML
    - definem um esquema de armazenamento proprietário ou utilizam um esquema de armazenamento de um BD convencional

# INTRODUÇÃO - SINTAXE XML

- Dados XML são definidos em um **documento XML**
- Um documento XML contém
  - **cabeçalho**
  - **dados**
    - elementos simples ou compostos
    - atributos de elementos
    - referências a entidades
  - **comentários**
  - **instruções de processamento**

# EXEMPLO DE DOCUMENTO XML

```
<?xml version ="1.0" encoding ="UTF-8" ?>
<!-- documento XML sobre livros -->
<!DOCTYPE livros [
    <!ENTITY xml "eXtensible Markup Language">
    ...
]>
<livros>
    <livro ISBN="112">
        <título>Tecnologia &xml;</título>
        <autor>
            <nome>João da Silva</nome>
            <eMail>js@hotmail.com</eMail>
        </autor>
        ...
        <capítulo nome="Introdução">A &xml; foi ...
            <seção>
                <nome>Linguagens de Marcação</nome> ...
            </seção>
        </capítulo> ...
    </livro> ...
</livros>
```

# EXEMPLO DE DOCUMENTO XML

```
<?xml version="1.0" encoding="UTF-8" ?>
<!-- documento XML sobre livros -->
<!DOCTYPE livros [
    <!ENTITY xml "eXtensible Markup Language">
    ...
]>
<livros>
    <livro ISBN="112">
        <título>Tecnologia &xml;</título>
        <autor>
            <nome>João da Silva</nome>
            <eMail>js@hotmail.com</eMail>
        </autor>
        ...
        <capítulo nome="Introdução">A &xml; foi ...
            <seção>
                <nome>Linguagens de Marcação</nome> ...
            </seção>
        </capítulo> ...
    </livro> ...
</livros>
```

cabeçalho

# CABEÇALHO

- Instruções para o processador de XML

`<?xml version="1.0" encoding="UTF-8" ?>`

**1.1**

**UTF-8**

- Veja mais detalhes sobre o encoding na especificação do XML:  
<http://www.w3.org/TR/2006/REC-xml11-20060816/#NT-EncodingDecl>

# EXEMPLO DE DOCUMENTO XML

```
<?xml version ="1.0" encoding ="UTF-8" ?>
<!-- documento XML sobre livros --> ← comentário
<!DOCTYPE livros [
    <!ENTITY xml "eXtensible Markup Language">
    ...
]>
<livros>
    <livro ISBN="112">
        <título>Tecnologia &xml;</título>
        <autor>
            <nome>João da Silva</nome>
            <eMail>js@hotmail.com</eMail>
        </autor>
        ...
        <capítulo nome="Introdução">A &xml; foi ...
            <seção>
                <nome>Linguagens de Marcação</nome> ...
            </seção>
        </capítulo> ...
    </livro> ...
</livros>
```

# EXEMPLO DE DOCUMENTO XML


```
<?xml version ="1.0" encoding ="UTF-8" ?>
<!-- documento XML sobre livros -->
<!DOCTYPE livros [
    <!ENTITY xml "eXtensible Markup Language">
    ...
]>
<livros>
    <livro ISBN="112">
        <título>Tecnologia &xml;</título>
        <autor>
            <nome>João da Silva</nome>
            <eMail>js@hotmail.com</eMail>
        </autor>
        ...
        <capítulo nome="Introdução">A &xml; foi ...
            <seção>
                <nome>Linguagens de Marcação</nome> ...
            </seção>
        </capítulo> ...
    </livro> ...
</livros>
```



instrução de processamento

# EXEMPLO DE DOCUMENTO XML

```
<?xml version ="1.0" encoding ="UTF-8" ?>
<!-- documento XML sobre livros -->
<!DOCTYPE livros [
    <!ENTITY xml "eXtensible Markup Language">
    ...
]>
<livros>
    <livro ISBN="112">
        <título>Tecnologia &xml;</título>
        <autor>
            <nome>João da Silva</nome>
            <eMail>js@hotmail.com</eMail>
        </autor>
        ...
        <capítulo nome="Introdução">A &xml; foi ...
            <seção>
                <nome>Linguagens de Marcação</nome> ...
            </seção>
        </capítulo> ...
    </livro> ...
</livros>
```



A blue arrow points from the text "elemento raiz" to the opening tag of the root element, `<livros>`.



# EXEMPLO DE DOCUMENTO XML

```
<?xml version ="1.0" encoding ="UTF-8" ?>
<!-- documento XML sobre livros -->
<!DOCTYPE livros [
    <!ENTITY xml "eXtensible Markup Language">
    ...
]>
<livros>
    <livro ISBN="112">
        <título>Tecnologia &xml;</título>
        <autor>
            <nome>João da Silva</nome>
            <eMail>js@hotmail.com</eMail>
        </autor>
        ...
        <capítulo nome="Introdução">A &xml; foi ...
            <seção>
                <nome>Linguagens de Marcação</nome> ...
            </seção>
        </capítulo> ...
    </livro> ...
</livros>
```

elemento simples  
(#PCDATA)



# EXEMPLO DE DOCUMENTO XML

```
<?xml version ="1.0" encoding ="UTF-8" ?>
<!-- documento XML sobre livros -->
<!DOCTYPE livros [
    <!ENTITY xml "eXtensible Markup Language">
    ...
]>
<livros>
    <livro ISBN="112">
        <título>Tecnologia &xml;</título>
        <autor>
            <nome>João da Silva</nome>
            <eMail>js@hotmail.com</eMail>
        </autor>
        ...
        <capítulo nome="Introdução">A &xml; foi ...
            <seção>
                <nome>Linguagens de Marcação</nome> ...
            </seção>
        </capítulo> ...
    </livro> ...
</livros>
```

elemento simples  
(#PCDATA)

elemento complexo

# EXEMPLO DE DOCUMENTO XML

```
<?xml version ="1.0" encoding ="UTF-8" ?>
<!-- documento XML sobre livros -->
<!DOCTYPE livros [
    <!ENTITY xml "eXtensible Markup Language">
    ...
]>
<livros>
    <livro ISBN="112">
        <título>Tecnologia &xml;</título>
        <autor>
            <nome>João da Silva</nome>
            <eMail>js@hotmail.com</eMail>
        </autor>
        ...
        <capítulo nome="Introdução">A &xml; foi ...
            <seção>
                <nome>Linguagens de Marcação</nome> ...
            </seção>
        </capítulo> ...
    </livro> ...
</livros>
```

elemento simples  
(#PCDATA)


elemento complexo

elemento misto

# EXEMPLO DE DOCUMENTO XML

```
<?xml version ="1.0" encoding ="UTF-8" ?>
<!-- documento XML sobre livros -->
<!DOCTYPE livros [
    <!ENTITY xml "eXtensible Markup Language">
    ...
]>
<livros>
    <livro ISBN="112">
        <título>Tecnologia &xml;</título>
        <autor>
            <nome>João da Silva</nome>
            <eMail>js@hotmail.com</eMail>
        </autor>
        ...
        <capítulo nome="Introdução">A &xml; foi ...
            <seção>
                <nome>Linguagens de Marcação</nome> ...
            </seção>
        </capítulo> ...
    </livro> ...
</livros>
```

**atributo**



# EXEMPLO DE DOCUMENTO XML

```
<?xml version ="1.0" encoding ="UTF-8" ?>
<!-- documento XML sobre livros -->
<!DOCTYPE livros [
    <!ENTITY xml "eXtensible Markup Language">
    ...
]>
<livros>
    <livro ISBN="112">
        <título>Tecnologia &xml;</título>
        <autor>
            <nome>João da Silva</nome>
            <eMail>js@hotmail.com</eMail>
        </autor>
        ...
        <capítulo nome="Introdução">A &xml; foi ...
            <seção>
                <nome>Linguagens de Marcação</nome> ...
            </seção>
        </capítulo> ...
    </livro> ...
</livros>
```

**declaração de entidade**

**referência a uma entidade**

# DOCUMENTO XML BEM FORMADO

- Toda *tag* de início deve possuir a *tag* de final
  - `<nome> . . . </nome>`
  - `<nome />`
- Todas *tags* são aninhadas (ordem não pode ser misturada)
- Começam com letra ou sublinhado
- Após primeiros caracteres são aceitos:
  - Números, "-" e "."

# DOCUMENTO XML BEM FORMADO

- Não podem conter espaços em branco
- Não podem começar com a *string* "XML"
- XML é ***case sensitive***

# DOCUMENTO XML BEM FORMADO

- *Tags* estão bem formadas ?

```
<nome Aluno>Vitor</nome Aluno>
```

```
<nomeAluno>Vitor</nomeAluno>
```

```
<nomeAluno>Vitor</nomealuno>
```



# DOCUMENTO XML BEM FORMADO

- *Tags* estão bem formadas?

```
<nome>Vitor</nome>
```

```
<nome />
```

```
<aluno>  
    <nome>Vitor  
</aluno>
```

# DOCUMENTO XML BEM FORMADO

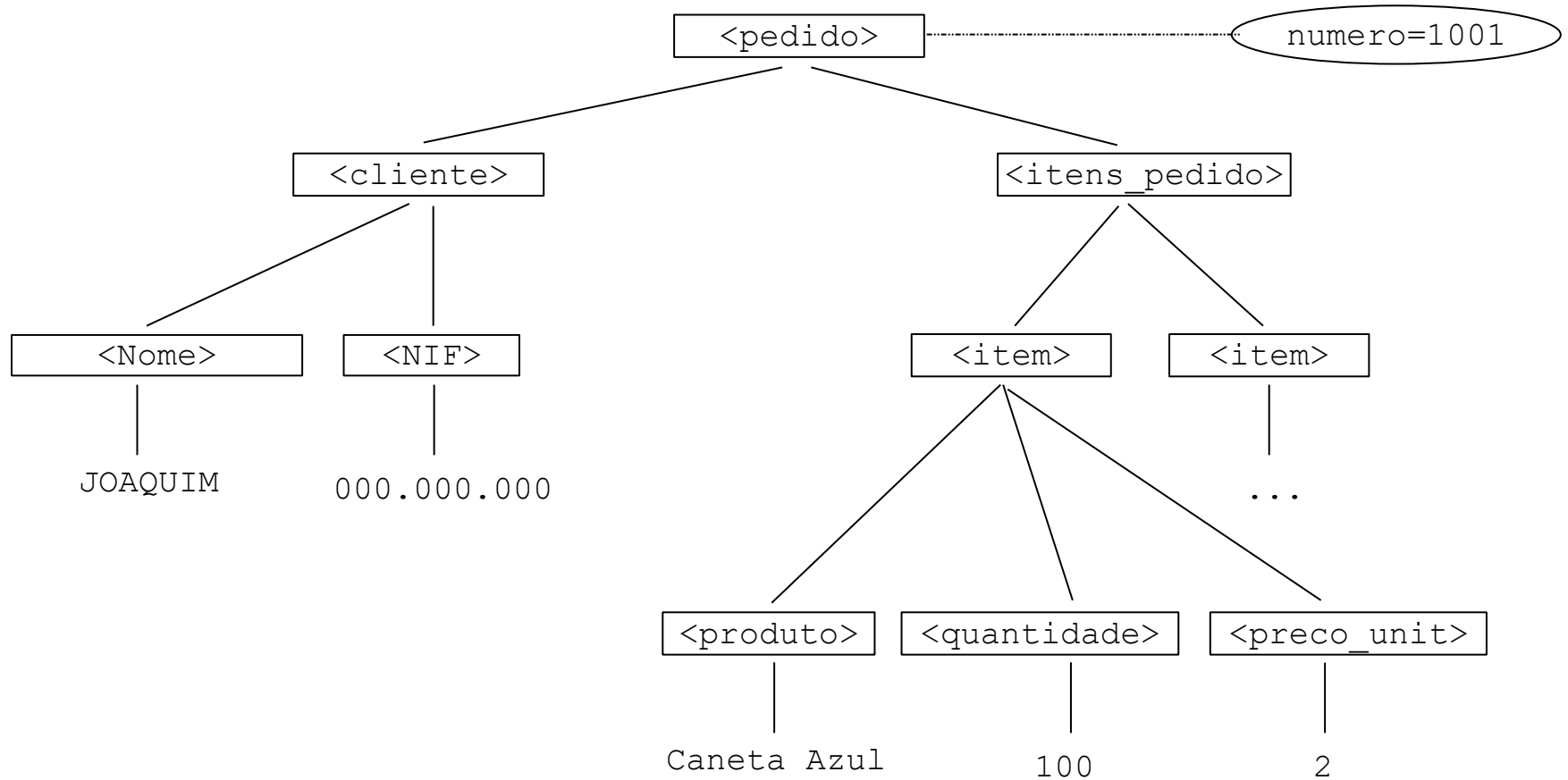
- *Tags* estão bem formadas?

```
<aluno>  
    <nome>Vitor </nome>  
</aluno>
```

```
<aluno>  
    <nome>Vitor</aluno>  
<nome>
```

# Exercício em sala

- Crie o documento XML para representar os dados de pedidos:



# EXERCÍCIO EM SALA

- Crie um documento XML para representar a estrutura do banco de dados para uma EMPRESA DE AGENCIAMENTO DE SERVIÇOS.



- Crie pelo menos DOIS registros de cada tabela
- Abra o documento XML no *Browser* para verificar se está BEM FORMADO

# DICA

- O *Notepad++* possui um *plug-in* que possui suporte à edição de documentos XML

- XML Tools

(<http://blog.irvingduran.com/2011/08/installing-xml-plugin-in-notepad/>)



# NATUREZA DE UM DADO XML

- Um dado XML é um dado não convencional
  - é um dado semiestruturado
- Dado semiestruturado
  - estrutura heterogênea
  - estrutura auto descritiva
  - estrutura parcial

# ESTRUTURA HETEROGÊNEA

- Cada instância com um esquema particular
  - consequência
    - esquemas extensos para suportar representações alternativas

```
<autor>  
  <nome>Joao da Silva</nome>  
  <endereço>rua B, 23</endereço>  
  <eMail>xyz@abc....</eMail>  
</autor>
```

```
<autor>  
  <nome>Antonio Sauro</nome>  
  <endereço>  
    <rua>Rua A</rua>  
    <numero>767</numero>  
    <cidade>Lisboa</cidade>  
  </endereço>  
  <fone>33313333</fone>  
  <fone>33313332</fone>  
</autor>
```

# ESTRUTURA AUTO DESCRITIVA

- Cada instância carrega o seu esquema

```
<autor>  
  <nome>Antonio Sauro</nome>  
  <endereco>  
    <rua>Rua A</rua>  
    <numero>767</numero>  
    <cidade>Lisboa</cidade>  
  </endereco>  
  <fone>33313333</fone>  
  <fone>33313332</fone>  
</autor>
```



# ESTRUTURA PARCIAL

- Apenas parte da descrição da instância precisa ser estruturada

```
<capitulo numero = "2" titulo = "Tecnologia XML">  
  Este capítulo descreve ... XML  
  <ref>(Mel03)</ref>. XML é um padrão ...  
  <seção número = 1>  
    <título>DTD</título>  
    Esta seção descreve ...  
  </seção>  
  ...  
</capitulo>
```

# DADO XML & DADO CONVENCIONAL

- Dados XML não são naturalmente adequados para armazenamento em BDs

<b>Dado Convencional</b>	<b>Dado XML</b>
representação homogênea	representação heterogênea
esquema independente dos dados	representação auto descritiva
totalmente estruturado	estrutura parcial
esquema enxuto	esquema extenso

# TECNOLOGIA XML

- Muitas semelhanças com a tecnologia de SGBD
  - DTD e XSD
    - definição do esquema de um documento XML
  - *XPath e XQuery*
    - linguagens de navegação/consulta a documentos XML
  - DOM e SAX
    - APIs para acesso a dados XML
  - XSLT...

# SOFTWARES PARA OS EXERCÍCIOS

- DICA:

- Use os *softwares* abaixo indicados para validar os exercícios realizados em sala de aula:

- *Exchanger XML Editor*

- <http://www.exchangerxml.com>

- *BaseX*

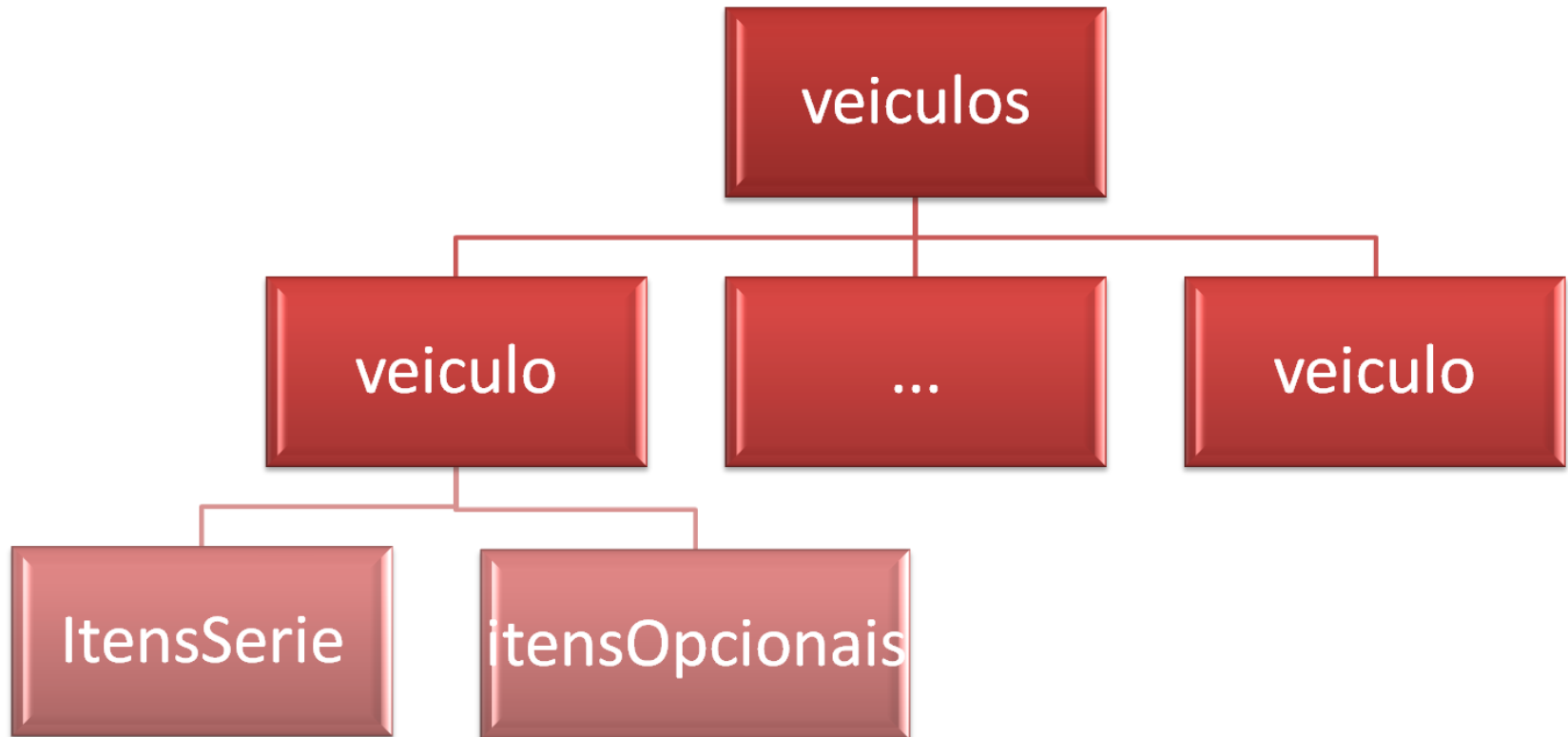
- <http://basex.org>

# EXERCÍCIO 01

- Crie um documento XML para representar a estrutura de um banco de dados com informações sobre automóveis.
  - Crie um documento o mais detalhado possível (informações sobre carros e quantidade de modelos (instâncias))
  - Abra o documento XML no *Browser* para verificar se está BEM FORMADO
  - Cada aluno fará o exercício considerando os modelos de uma marca distinta (apresentadas pelo professor)
  - Procure separar itens de série e opcionais, pensando em um futuro sistema de comparação de veículos

# EXERCÍCIO 01

- Crie um documento XML para representar a estrutura de um banco de dados com informações sobre automóveis.



# EXTRAS

- Validador XML

<http://www.xmlvalidation.com/?L=0>

- Validador XHTML/XML

<http://validator.aborla.net/>

- Validador HTML/XHTML/WML/XML

<http://www.validome.org/>

# DEFININDO UM DOCTYPE

## ○ HTML 4.01 *Strict, Transitional, Frameset*

- **Transitional:** permite que um documento utilize alguns recursos depreciados (obsoletos) na nova versão.
- **Strict:** DTD que contém a definição formal da especificação, tal como ela deve ser usada.
- **Frameset:** inclui os elementos necessários para o uso de *frames*, que nem sempre são necessários em um *site*.

48

```
<!DOCTYPE HTML PUBLIC "-//W3C//DTD HTML 4.01//EN"  
    "http://www.w3.org/TR/html4/strict.dtd">
```

```
<!DOCTYPE HTML PUBLIC "-//W3C//DTD HTML 4.01  
    Transitional//EN" "http://www.w3.org/TR/html4/loose.dtd">
```

```
<!DOCTYPE HTML PUBLIC "-//W3C//DTD HTML 4.01 Frameset//EN"  
    "http://www.w3.org/TR/html4/frameset.dtd">
```