



Rogério Araújo | rogerioaraujo.com.br

Série Rações Semanais

# Namespaces e DOM/SAX

Rogério Araújo

<http://rogerioaraujo.wordpress.com>

**Série Rações Semanais**

# **Namespaces e DOM/SAX**

**Rogério Araújo**

## Questões

---

*O insucesso é apenas uma oportunidade para recomeçar de novo com mais inteligência.*

*Henry Ford*

### CESPE 2010 TCU

**139** Considere que o referido líder da equipe solicite a um programador que comente o trecho de código de uma página de programação server-side apresentado a seguir.

```
<html xmlns="http://www.w3.org/1999/xhtml"
      xmlns:h="http://java.sun.com/jsf/html"
      xmlns:f="http://java.sun.com/jsf/core"
      xmlns:ui="http://java.sun.com/jsf/facelets"
      xmlns:ez="http://java.sun.com/jsf/composite/ezcomp">
```

Nessa situação, se o programador disser que a página importa exatamente cinco namespaces XML, sendo que o namespace default possui o URI (universal resource identifier) <http://www.w3.org/1999/xhtml>, esse comentário estará correto.

### CESPE 2010 INMETRO – Cargo 16

**46 Letra E** O documento apresentado usa <http://java.sun.com/jsf/html> como namespace default.

```
<!DOCTYPE html PUBLIC "-//W3C//DTD XHTML 1.0 Transitional//EN"
"http://www.w3.org/TR/xhtml1/DTD/xhtml1-transitional.dtd">
<html xmlns="http://www.w3.org/1999/xhtml"
      xmlns:ui="http://java.sun.com/jsf/facelets"
      xmlns:h="http://java.sun.com/jsf/html">
</html>
```

### FGV 2008 Senado Federal – Analista de Sistemas

Para as questões 45 a 47 utilize o documento XML em (A) e seu correspondente XML Schema em (B):

**A**

```

<?xml version="1.0"?>
<OrdemdeCompra xmlns="http://xyz.org/oc.xsd"
  xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
  xsi:schemaLocation="http://xyz.org/oc.xsd"
  datacompra="20-12-2000">
  <enddestino pais="BRASIL">
    <nome>Luis Potata</nome>
    <rua>Rua Torta 423</rua>
    <cidade>Cintra</cidade>
    <estado>SP</estado>
    <cep>90952-023</cep>
  </enddestino>
  <endpagamento pais=" BRASIL">
    <nome>Julia Pombal</ nome>
    <rua>Rua Silvano 30</rua>
    <cidade>Pirara</cidade>
    <estado>PA</estado>
    <cep>76889-043</cep>
  </endpagamento>
  <comentario>Esta compra é urgente!</comentario>
  (...)</OrdemdeCompra>

```

**B**

```

<xs:schema xmlns:xs="http://www.w3.org/2001/XMLSchema"
  targetNamespace="http://xyz.org/oc.xsd"
  xmlns="http://xyz.org/oc.xsd"
  elementFormDefault="qualified">
  <xs:element name="OrdemdeCompra"
    type="TipoOrdemdeCompra"/>
  <xs:element name="comentario" type="xs:string"/>
  <xs:complexType name=" TipoOrdemdeCompra ">
    <xs:sequence>
      <xs:element name="enddestino" type="endereco"/>
      <xs:element name="endpagamento" type="endereco"/>
      <xs:element ref="comentario" minOccurs="0"/>
      <xs:element name="itens" type="Itens"/>
    </xs:sequence>
    <xs:attribute name="datacompra" type="xs:date"/>
  </xs:complexType>
  <xs:complexType name="endereco">
    <xs:sequence>
      <xs:element name="nome" type="xs:string"/>
      <xs:element name="rua" type="xs:string"/>
      <xs:element name="cidade" type="xs:string"/>
      <xs:element name="estado" type="xs:string"/>
      <xs:element name="cep" type="xs:decimal"/>
    </xs:sequence>
    <xs:attribute name="pais" type="xs:NMTOKEN"
      fixed="BRASIL"/>
  </xs:complexType>
  (...)</xs:schema>

```

Onde (...) representa outros elementos aqui não apresentados.

**45** É correto afirmar que, no documento XML:

- a) o elemento rua poderia anteceder o elemento nome em enddestino.
- b) o elemento comentario pode aparecer mais de uma vez.
- c) http://xyz.org/oc.xsd é o namespace padrão.
- d) o atributo datacompra não deveria estar dentro da tag de abertura do elemento OrdemdeCompra.
- e) os elementos enddestino e endpagamento não podem ter um atributo com mesmo nome.

## RGA 2011 (questões de minha autoria)

**1** Há duas formas de classificação de APIs para XML: uma baseada em árvores, por exemplo SAX, e outra baseada em eventos, por exemplo DOM.

**2** DOM, por criar uma árvore de nós a partir de um documento XML, é mais simples que o SAX que trabalha disponibilizando métodos para resposta a eventos produzidos durante a leitura do mesmo documento.

**3** DOM é um padrão W3C, SAX não é.

**4** DOM é separado em apenas HTML DOM e XML DOM.

**5** DOM define os [...] e as [...] de todos os elementos do documento e os [...] para acessá-los.

**6** No DOM, uma árvore de nós mostra apenas o conjunto de nós de um documento XML.

**7** SAX é apenas disponível em Java.

**8** O acesso aos dados de um documento XML é feita por um parser SAX de forma sequencial.

**9** O SAX consome menos memória do que o DOM.

**10** É obrigatório o uso de DTD ou XML Schema ao usar um parser SAX ou DOM para leitura de documentos XML.

## CESGRANRIO 2008 Petrobrás – Técnico em Informática

Ao ler e processar arquivos XML, é interessante considerar o desempenho em duas dimensões: tempo e memória. Examine, a seguir, as afirmativas sobre os padrões de consumo de memória, tanto no SAX quanto no DOM.

- I. O consumo de memória do SAX é uma constante, independente do tamanho do documento XML.
- II. O consumo de memória de uma implementação DOM depende do tamanho do documento XML.
- III. SAX sempre consome mais menos memória do que DOM.

Está(ão) correta(s) a(s) afirmativa(s)

- a) I, apenas.
- b) I e II, apenas.
- c) I e III, apenas.
- d) II e III, apenas.
- e) I, II e III.

## CESPE 2009 INMETRO – Cargo 5

**75** Na linguagem XML, todo atributo é parte de um elemento, todo elemento é raiz ou filho de uma raiz, a construção de uma árvore pode empregar o modelo DOM, uma transformação pode ser direcionada por um documento XSLT, e quando se deseja consumir pouca memória no processamento de XML pode-se empregar um parser do tipo SAX.

## CESPE 2009 INMETRO – Cargo 10

**113** Arquivos XML permitem o processamento automatizado de seus dados por meio de tecnologias como DOM e SAX. Os arquivos HTML, por sua vez, só podem ser processados automaticamente por meio da tecnologia SAX.

## Gabarito

---

*Só se pode alcançar um grande êxito quando nos mantemos fiéis a nós mesmos.*

*Friedrich Nietzsche*

### CESPE 2010 TCU

**139** Considere que o referido líder da equipe solicite a um programador que comente o trecho de código de uma página de programação server-side apresentado a seguir.

```
<html xmlns="http://www.w3.org/1999/xhtml"
      xmlns:h="http://java.sun.com/jsf/html"
      xmlns:f="http://java.sun.com/jsf/core"
      xmlns:ui="http://java.sun.com/jsf/facelets"
      xmlns:ez="http://java.sun.com/jsf/composite/ezcomp">
```

Nessa situação, se o programador disser que a página importa exatamente cinco namespaces XML, sendo que o namespace default possui o URI (universal resource identifier) <http://www.w3.org/1999/xhtml>, esse comentário estará correto.

**Gabarito: CERTO.**

### CESPE 2010 INMETRO – Cargo 16

**46 Letra E** O documento apresentado usa <http://java.sun.com/jsf/html> como namespace default.

```
<!DOCTYPE html PUBLIC "-//W3C//DTD XHTML 1.0 Transitional//EN"
"http://www.w3.org/TR/xhtml1/DTD/xhtml1-transitional.dtd">
<html xmlns="http://www.w3.org/1999/xhtml"
      xmlns:ui="http://java.sun.com/jsf/facelets"
      xmlns:h="http://java.sun.com/jsf/html">
</html>
```

**Gabarito: ERRADO.**

### FGV 2008 Senado Federal – Analista de Sistemas

Para as questões 45 a 47 utilize o documento XML em (A) e seu correspondente XML Schema em (B):

**A**

```

<?xml version="1.0"?>
<OrdemdeCompra xmlns="http://xyz.org/oc.xsd"
  xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
  xsi:schemaLocation="http://xyz.org/oc.xsd"
  datacompra="20-12-2000">
  <enddestino pais="BRASIL">
    <nome>Luis Potata</nome>
    <rua>Rua Torta 423</rua>
    <cidade>Cintra</cidade>
    <estado>SP</estado>
    <cep>90952-023</cep>
  </enddestino>
  <endpagamento pais=" BRASIL">
    <nome>Julia Pombal</ nome>
    <rua>Rua Silvano 30</rua>
    <cidade>Pirara</cidade>
    <estado>PA</estado>
    <cep>76889-043</cep>
  </endpagamento>
  <comentario>Esta compra é urgente!</comentario>
  (...)</OrdemdeCompra>

```

**B**

```

<xs:schema xmlns:xs="http://www.w3.org/2001/XMLSchema"
  targetNamespace="http://xyz.org/oc.xsd"
  xmlns="http://xyz.org/oc.xsd"
  elementFormDefault="qualified">
  <xs:element name="OrdemdeCompra"
    type="TipoOrdemdeCompra"/>
  <xs:element name="comentario" type="xs:string"/>
  <xs:complexType name=" TipoOrdemdeCompra ">
    <xs:sequence>
      <xs:element name="enddestino" type="endereco"/>
      <xs:element name="endpagamento" type="endereco"/>
      <xs:element ref="comentario" minOccurs="0"/>
      <xs:element name="itens" type="Itens"/>
    </xs:sequence>
    <xs:attribute name="datacompra" type="xs:date"/>
  </xs:complexType>
  <xs:complexType name="endereco">
    <xs:sequence>
      <xs:element name="nome" type="xs:string"/>
      <xs:element name="rua" type="xs:string"/>
      <xs:element name="cidade" type="xs:string"/>
      <xs:element name="estado" type="xs:string"/>
      <xs:element name="cep" type="xs:decimal"/>
    </xs:sequence>
    <xs:attribute name="pais" type="xs:NMTOKEN"
      fixed="BRASIL"/>
  </xs:complexType>
  (...)</xs:schema>

```

Onde (...) representa outros elementos aqui não apresentados.

**45** É correto afirmar que, no documento XML:

- a) o elemento rua poderia anteceder o elemento nome em enddestino.
- b) o elemento comentario pode aparecer mais de uma vez.
- c) http://xyz.org/oc.xsd é o namespace padrão.
- d) o atributo datacompra não deveria estar dentro da tag de abertura do elemento OrdemdeCompra.
- e) os elementos enddestino e endpagamento não podem ter um atributo com mesmo nome.

**Gabarito:** letra C.

## RGA 2011 (questões de minha autoria)

**1** Há duas formas de classificação de APIs para XML: uma baseada em árvores, por exemplo SAX, e outra baseada em eventos, por exemplo DOM.

**Gabarito:** **ERRADO.**

**2** DOM, por criar uma árvore de nós a partir de um documento XML, é mais simples que o SAX que trabalha disponibilizando métodos para resposta a eventos produzidos durante a leitura do mesmo documento.

**Gabarito:** **ERRADO.**

**3** DOM é um padrão W3C, SAX não é.

**Gabarito:** **CERTO.**

**4** DOM é separado em apenas HTML DOM e XML DOM.

**Gabarito: ERRADO.**

**5** DOM define os **objetos** e as **propriedades** de todos os elementos do documento e os **métodos (interfaces)** para acessá-los.

**6** No DOM, uma árvore de nós mostra apenas o conjunto de nós de um documento XML.

**Gabarito: ERRADO.**

**7** SAX é apenas disponível em Java.

**Gabarito: ERRADO.**

**8** O acesso aos dados de um documento XML é feita por um parser SAX de forma sequencial.

**Gabarito: CERTO.**

**9** O SAX consome menos memória do que o DOM.

**Gabarito: CERTO.**

**10** É obrigatório o uso de DTD ou XML Schema ao usar um parser SAX ou DOM para leitura de documentos XML.

**Gabarito: ERRADO.**

## CESGRANRIO 2008 Petrobrás – Técnico em Informática

Ao ler e processar arquivos XML, é interessante considerar o desempenho em duas dimensões: tempo e memória. Examine, a seguir, as afirmativas sobre os padrões de consumo de memória, tanto no SAX quanto no DOM.

- I. O consumo de memória do SAX é uma constante, independente do tamanho do documento XML.
- II. O consumo de memória de uma implementação DOM depende do tamanho do documento XML.
- III. SAX sempre consome mais memória do que DOM.

Está(ão) correta(s) a(s) afirmativa(s)

- a) I, apenas.
- b) I e II, apenas.
- c) I e III, apenas.
- d) II e III, apenas.
- e) I, II e III.

**Gabarito: letra C.**

## CESPE 2009 INMETRO – Cargo 5

**75** Na linguagem XML, todo atributo é parte de um elemento, todo elemento é raiz ou filho de uma raiz, a construção de uma árvore pode empregar o modelo DOM, uma transformação pode ser direcionada por um documento XSLT, e quando se deseja consumir pouca memória no processamento de XML pode-se empregar um parser do tipo SAX.

**Gabarito:** CERTO.

## CESPE 2009 INMETRO – Cargo 10

**113** Arquivos XML permitem o processamento automatizado de seus dados por meio de tecnologias como DOM e SAX. Os arquivos HTML, por sua vez, só podem ser processados automaticamente por meio da tecnologia SAX.

**Gabarito:** ERRADO.



<http://rogerioaraujo.wordpress.com>