

Auto-évaluation XML cours de base

Document: f1006test.fm

24 octobre 2019

ABIS Training & Consulting
Diestsevest 32 / 4b
B-3000 Leuven
Belgium

The logo for ABIS Training & Consulting. The word "abis" is written in a bold, blue, lowercase sans-serif font. A solid black circle is positioned above the letter 'i'. Below the text is a thick, horizontal grey bar.

TRAINING & CONSULTING

INTRODUCTION AUTO-ÉVALUATION XML COURS DE BASE

Ce test est une auto-évaluation qui vous permettra de déterminer si le cours XML: cours de base est encore utile pour vous.

Ce test comporte des questions à choix multiples. Pour la plupart des questions, plusieurs réponses correctes sont possibles. Cochez les solutions qui vous semblent bonnes, et comparez vos réponses avec les solutions.

Si vous n'avez aucune idée, ne cochez aucune réponse. Votre réponse sera considérée comme incorrecte mais vous n'améliorerez pas votre score par un coup de chance. La durée de ce test devrait être de plus ou moins 10 minutes.

Vous trouverez les réponses aux questions ainsi que la grille d'évaluation à la fin de ce document.

Bonne chance!

QUESTIONS AUTO-ÉVALUATION XML COURS DE BASE

1. Observons le document xml suivant.

```
<Question subject='Math'>
  <Remark>please, read <b> the question </b> carefully </Remark>
  <Calculation>
    <Number>2</Number>
    <Operator>plus</Operator>
    <Number>3</Number>
  </Calculation>
  <Result>5 is the right answer</Result>
</Question>
```

Quel élément a un contenu mixte (mixed content) ?

- (a) Remark.
 - (b) Result.
 - (c) Calculation.
 - (d) Question.
2. Quel noeud est considéré par l'analyseur (parser) comme un noeud textuel ?
- (a) <PCDATA>1 > 2</PCDATA>
 - (b) <![CDATA[1 > 2]]>
 - (c) <CDATA>1 > 2</CDATA>
 - (d) <![CDATA[1 > 2]>
3. Quels sont les éléments xml corrects ? [2 réponses]
- [a] <Calc>1 > 2</Calc>
 - [b] <Calc>2 < 3</Calc>
 - [c] <Calc>2 > 3</Calc>
 - [d] <Calc><![CDATA[1 > 2]>

4. Considérons l'élément xml suivant.

```
<name>
  <firstname>John</firstname>
  <lastname>Lennon</lastname>
</name>
```

Quelle description DTD est juste?

(a)

```
<!ELEMENT name (firstname, lastname)+>
<!ELEMENT firstname (#CDATA)>
<!ELEMENT lastname (#CDATA)>
```

(b)

```
<!ELEMENT name (firstname | lastname)>
<!ELEMENT firstname (#CDATA)>
<!ELEMENT lastname (#CDATA)>
```

(c)

```
<!ELEMENT name (lastname | firstname)+>
<!ELEMENT firstname (#PCDATA)>
<!ELEMENT lastname (#PCDATA)>
```

(d)

```
<!ELEMENT name (firstname | lastname)>
<!ELEMENT firstname (#PCDATA)>
<!ELEMENT lastname (#PCDATA)>
```

5. Laquelle est une bonne façon de définir en DTD des attributs 'nr' et 'lang' pour l'élément 'Article'?

(a) <!ATTLIST Article nr #CDATA REQUIRED lang #CDATA OPTIONAL>

(b) <!ATTRIBUTES Article nr #PCDATA lang #PCDATA>

(c) <!ELEMENT Article <!ATTRIBUTES nr #REQUIRED lang #REQUIRED>>

(d) <!ATTLIST Article lang CDATA #IMPLIED nr CDATA #REQUIRED>

6. Considérons le document xml suivant.

```
<name>
  <firstname>John</firstname><lastname>Lennon</lastname>
</name>
```

Lequel est un bon schema?

O (a)

```
<xs:schema xmlns:xs="http://www.w3.org/2001/XMLSchema" >
  <xs:element name="FirstName" type="xs:string"/>
  <xs:element name="LastName" type="xs:string"/>
  <xs:element name="name">
    <xs:complexType>
      <xs:sequence>
        <xs:element ref="FirstName"/>
        <xs:element ref="LastName"/>
      </xs:sequence>
    </xs:complexType>
  </xs:element>
</xs:schema>
```

O (b)

```
<xsd:schema xmlns:xsd="http://www.w3.org/2001/XMLSchema" >
  <xsd:element name="name">
    <xsd:complexType>
      <xsd:sequence>
        <xsd:element name="firstname" type="xsd:string"/>
        <xsd:element name="lastname" type="xsd:string"/>
      </xsd:sequence>
    </xsd:complexType>
  </xsd:element>
</xsd:schema>
```

O (c)

```
<xs:schema xmlns:xs="http://www.w3.org/2001/XMLSchema" >
  <xs:element name="firstname" type="xs:String"/>
  <xs:element name="lastname" type="xs:String"/>
  <xs:element name="name" type="nametype"/>
  <xs:complexType type="nametype">
    <xs:sequence>
      <xs:element ref="firstname"/>
      <xs:element ref="lastname"/>
    </xs:sequence>
  </xs:complexType>
</xs:schema>
```

O (d)

```
<xs:schema xmlns:xsd="http://www.w3.org/2001/XMLSchema" >
  <xs:element name="name">
    <xs:complexType>
      <xs:sequence>
        <xs:element ref="firstname" />
        <xs:element ref="lastname" />
      </xs:sequence>
    </xs:complexType>
  </xs:element>
</xs:schema>
```

7. Quel élément ne peut pas être utilisé comme enfant de `<xs:simpleType>` ?
- (a) `<xs:extension>`
 - (b) `<xs:restriction>`
 - (c) `<xs:list>`
 - (d) `<xs:annotation>`
8. De quel attribut pour l'élément racine `xs:schema` a-t-on besoin si l'on veut mettre les éléments dans un namespace ?
- (a) `elementFormDefault`
 - (b) `targetNamespace`
 - (c) `defaultNamespace`
 - (d) `attributeFormDefault`

9. Quels documents XML sont well-formed ? [2 réponses]

[a]

```
<?xml version="1.0" encoding="UTF-8"?>
<x xmlns="https://abis.be" xmlns:abis="www.abis.be"
  xmlns:a="https://abis.be">
  <y a:a="1" abis:a="2"></y>
</x>
```

[b]

```
<?xml version="1.0" encoding="UTF-8"?>
<x xmlns="www.abis.be" xmlns:abis="www.abis.be" >
  <y a="1" abis:a="2"></y>
</x>
```

[c]

```
<?xml version="1.0" encoding="UTF-8"?>
<x xmlns="www.abis.be">
  <y a="1" a="2"></y>
</x>
```

[d]

```
<?xml version="1.0" encoding="UTF-8"?>
<x xmlns="www.abis.be" xmlns:abis="www.abis.be" xmlns:a="www.abis.be">
  <y a:a="1" abis:a="2"></y>
</x>
```

10. Quel type de noeud existe en XPath?

- (a) entity
- (b) cdata
- (c) processing-instruction
- (d) document-fragment

11. Quel axe est un 'reverse axis'?

- (a) descendant
- (b) ancestor
- (c) following
- (d) attribute

12. D'après les xml et xsl suivants. Quel est le résultat?

```
<Company>
  <Name>Abis</Name>
  <City>Leuven</City>
  <Telephone>016/245610</Telephone>
</Company>

<xsl:stylesheet version="1.0" xmlns:xsl="http://www.w3.org/1999/XSL/Transform">
  <xsl:template match="/">
    <xsl:apply-templates select="Company/Telephone" />
  </xsl:template>
  <xsl:template match="Telephone">
    <xsl:apply-templates select="preceding-sibling:*" />
  </xsl:template>
</xsl:stylesheet>
```

- (a) AbisLeuven
- (b) LeuvenAbis
- (c) LeuvenAbis016/245610
- (d) Aucune des précédentes propositions

13. Avec quel type de 'nodetype-test' peut-on sélectionner une section CDATA? [2 réponses]

- [a] child::text()
- [b] child::cdata()
- [c] child::processing-instruction()
- [d] child::node()

14. Comment peut-on créer en xsl un élément vide nommé 'PartNumber'?

(a)

```
<xsl:element name="PartNumber">
</xsl:element>
```

(b)

```
<xsl:element name="PartNumber" type="empty"/>
```

(c)

```
<xsl:text disable-output-escaping="yes">&lt;PartNumber/&gt;</xsl:text>
```

(d)

```
<PartNumber> </PartNumber>
```

15. Quel élément xsl ne peut pas avoir un attribut 'select'?

(a) xsl:copy

(b) xsl:apply-template

(c) xsl:value-of

(d) xsl:for-each

16. Quelle 'output-method' affiche uniquement des noeuds textuels?

(a) string

(b) text

(c) html

(d) xml

17. Quelle expression xpath peut-on utiliser pour sélectionner un élément Person qui a le même parent que le noeud contextuel et qui se trouve le plus proche de ce noeud contextuel?

(a) /descendant-or-self::Person

(b) preceding-sibling::Person

(c) preceding-sibling::Person[1]

(d) parent::*[1]

18. Quelle est la méthode correcte pour appeler ce template?

```
<xsl:template name="test">
  <xsl:param name="abc" />
</xsl:template>
```

(a)

```
<xsl:call-template name="test">
  <xsl:param name="abc" select="firstname" />
</xsl:call-template>
```

(b)

```
<xsl:apply-template select="test" />
```

(c)

```
<xsl:call-template name="test" />
```

19. On désire imprimer une espace entre le firstname et le lastname. Comment faire en xsl?

(a)

```
<xsl:value-of select="firstname" />
<xsl:text> </xsl:text>
<xsl:value-of select="lastname" />
```

(b)

```
<xsl:value-of select="firstname" /> <xsl:value-of select="lastname" />
```

(c)

```
<xsl:value-of select="firstname" />
&nbsp;
<xsl:value-of select="lastname" />
```

(d)

```
<xsl:preserve-space>
  <xsl:value-of select="firstname" />
  <xsl:value-of select="lastname" />
</xsl:preserve-space>
```

20. Quels types de noeud peut-on sélectionner avec * ? [2 réponses]

[a] Noeuds de commentaire

[b] Attributs

[c] Element nodes

[d] Text nodes

EVALUATION.

Ici sont les réponses correctes:

1. a
2. b
3. a c
4. c
5. d
6. b
7. a
8. b
9. a b
10. c
11. b
12. a
13. a d
14. c
15. a
16. b
17. c
18. c
19. a
20. b c

Comptez 1 point par bonne réponse. Pour les questions avec plusieurs bonnes réponses, comptez 1 point seulement si vous avez coché toutes les bonnes alternatives.

Si votre score atteint 80% ou plus, il ne vous est pas nécessaire de suivre ce cours. Dans ce cas vous pouvez considérer suivre un des [cours de suite](#).

Si votre score est entre 50% et 75%, le cours [XML cours de base](#) vous permettra de compléter vos connaissances.

Si votre score est inférieur à 50%, il vous est vivement conseillé de suivre le cours [XML cours de base](#). N'oubliez cependant pas de vérifier si vous disposez des connaissances préalables suffisantes pour suivre efficacement ce cours (auto-évaluation [XML concepts: fichier PDF](#))