



I10-002^{Q&As}

XML Master: Professional V2

Pass XML Master I10-002 Exam with 100% Guarantee

Free Download Real Questions & Answers **PDF** and **VCE** file from:

<https://www.geekcert.com/i10-002.html>

100% Passing Guarantee
100% Money Back Assurance

Following Questions and Answers are all new published by XML Master
Official Exam Center

- ⚙️ **Instant Download** After Purchase
- ⚙️ **100% Money Back** Guarantee
- ⚙️ **365 Days** Free Update
- ⚙️ **800,000+** Satisfied Customers





QUESTION 1

When receiving an XML document following the DTD defined by NewsML (portions modified for this question), the document is processed via SAX programming.

```
[DTD]
<!ENTITY % localid " Duid ID      #IMPLIED
                  Euid CDATA    #IMPLIED">

<!ELEMENT NewsIdentifier (NewsItemId ,
RevisionId)>
<!ELEMENT NewsItemId    (#PCDATA)>
<!ELEMENT RevisionId    (#PCDATA)>
<!ATTLIST RevisionId    Update   CDATA   #REQUIRED >

<!ELEMENT Identification (NewsIdentifier)>
<!ATTLIST Identification %localid; >

<!ELEMENT NewsItem      (Identification)>

<!ELEMENT NewsML        (NewsItem+)>
```

Push the Exhibit Button to load the referenced "DTD". When processing the following "XML Document" according to the method shown by "SAX Processing," which of the following correctly describes the output results (print method output)? Note that the attribute value is not output for this question.

[XML Document]
The root element name (document element name) is "NewsML".
The XML document follows the "DTD" definition.
The XML Document has no document type declaration.
The XML Document is comprised of elements, attributes and character data; comments and CDATA sections, etc. are not included.
The XML Document is indented, and includes ignorable white spaces such as line feeds and tabs before the start-tags.

[SAX Processing]
Use the following "ContentHandlerImpl" class, and parse the XML Document using SAX.
Assume no execution errors.

```
class ContentHandlerImpl extends DefaultHandler {
    private String elementName = "";

    public void startElement(
        String namespaceURI, String localName, String qName, Attributes atts)
        throws SAXException {
        elementName = qName;
    }

    public void endElement(
        String namespaceURI, String localName, String qName)
        throws SAXException {
        elementName = "";
    }

    public void characters(char[] ch, int start, int length)
        throws SAXException {
        if ( elementName.equals("") == false ) {
            System.out.print(new String(ch, start, length));
        }
    }
}
```

- A. All character data of elements can be output, ignorable white spaces are not output
- B. All character data of elements can be output, and ignorable white spaces are also output
- C. A portion of character data of elements cannot be output
- D. No character data of elements can be output



Correct Answer: B

QUESTION 2

Push the Exhibit Button to load the referenced "XML Document".

```
[XML Document]
<tns:TestML xmlns:tns="urn:xmlmaster:testml">
  <tns:record tns:level="1" tns:data="100" />
  <tns:record tns:level="2" tns:data="250" />
</tns:TestML>
```

Choose the XML Schema Document that correctly defines the structure of "XML Document".



- A. `<xs:schema
 xmlns:xs="http://www.w3.org/2001/XMLSchema"
 targetNamespace="urn:xmlmaster:testml"
 xmlns:tns="urn:xmlmaster:testml">

 <xs:element name="TestML" type="tns:testmlType" />
 <xs:complexType name="testmlType">
 <xs:sequence>
 <xs:element ref="tns:record" maxOccurs="unbounded" />
 </xs:sequence>
 </xs:complexType>

 <xs:element name="record" type="tns:recordType" />
 <xs:complexType name="recordType">
 <xs:attribute name="level" type="xs:int" />
 <xs:attribute name="data" type="xs:int" />
 </xs:complexType>
</xs:schema>`
- B. `<xs:schema
 xmlns:xs="http://www.w3.org/2001/XMLSchema"
 targetNamespace="urn:xmlmaster:testml"
 xmlns:tns="urn:xmlmaster:testml">

 <xs:element name="TestML" type="tns:testmlType" />
 <xs:complexType name="tns:testmlType">
 <xs:sequence>
 <xs:element ref="tns:record" maxOccurs="unbounded" />
 </xs:sequence>
 </xs:complexType>

 <xs:element name="record" type="tns:recordType" />
 <xs:complexType name="tns:recordType">
 <xs:attribute ref="tns:level" />
 <xs:attribute ref="tns:data" />
 </xs:complexType>
 <xs:attribute name="tns:level" type="xs:int" />
 <xs:attribute name="tns:data" type="xs:int" />
</xs:schema>`
- C. `<xs:schema
 xmlns:xs="http://www.w3.org/2001/XMLSchema"
 targetNamespace="urn:xmlmaster:testml"
 xmlns:tns="urn:xmlmaster:testml">

 <xs:element name="TestML" type="tns:testmlType" />
 <xs:complexType name="testmlType">
 <xs:sequence>
 <xs:element ref="tns:record" maxOccurs="unbounded" />
 </xs:sequence>
 </xs:complexType>

 <xs:element name="record" type="tns:recordType" />
 <xs:complexType name="recordType">
 <xs:attribute ref="tns:level" />
 <xs:attribute ref="tns:data" />
 </xs:complexType>
 <xs:attribute name="level" type="xs:int" />
 <xs:attribute name="data" type="xs:int" />
</xs:schema>`
- D. `<xs:schema
 xmlns:xs="http://www.w3.org/2001/XMLSchema"
 targetNamespace="urn:xmlmaster:testml"
 xmlns:tns="urn:xmlmaster:testml">

 <xs:element name="TestML">
 <xs:complexType>
 <xs:sequence>
 <xs:element name="record" maxOccurs="unbounded">
 <xs:complexType>
 <xs:attribute name="tns:level" type="xs:int" />
 <xs:attribute name="tns:data" type="xs:int" />
 </xs:complexType>
 </xs:element>
 </xs:sequence>
 </xs:complexType>
 </xs:element>
</xs:schema>`



A. Option A

B. Option B

C. Option C

D. Option D

Correct Answer: C

QUESTION 3

Push the Exhibit Button to load the referenced "XML Document".

```
[XML Document]
<Unfinished xmlns='urn:xmlmaster:UNFINISHED">
  <List>
    <Worker xmlns='urn:xmlmaster:WORKER"
Name="Jeff Workman"/>
    <Activity xmlns="urn:xmlmaster:VISIT"
Code="0002">
      <DateTime>2004-07-01 13:00</DateTime>
      <Location xmlns="urn:xmlmaster:CONTACT">
        <Name>John Smith</Name>
        <Address>Shinjuku Ward Tokyo</Address>
      </Location>
      <Job Procedure="Install Air Conditioner"
Time="2"/>
      <Status>COMPLETED</Status>
    </Activity>
    <Activity xmlns="urn:xmlmaster:VISIT"
Code="0003">
      <DateTime>2004-07-01 16:00</DateTime>
      <Location xmlns="urn:xmlmaster:CONTACT">
        <Name>Ken Miller</Name>
        <Address>Shibuya Ward Tokyo</Address>
      </Location>
      <Job Procedure="Clean air conditioner"
Time="4"/>
      <Status>WAITING</Status>
    </Activity>
  </List>
</Unfinished>
```

When processing the "XML Document" according to the method shown by "DOM Processing," which of the

following correctly describes the output results (print and/or println method output)? Although the expected processing result is choice "A", processing may not occur as expected.



```
[DOM Processing]
Process XML using the following method.
    printXML( doc );
The variable doc here references the Document instance of the loaded XML Document.
The DOM parser is namespace aware.
Assume no execution errors.

public static void printXML( Document doc ) {
    Node node = doc.getElementsByTagNameNS(
        "urn:xmlmaster:CONTACT", "Name").item(0);
    String name = node.getFirstChild().getNodeValue();

    NodeList nl = doc.getElementsByTagNameNS("urn:xmlmaster:VISIT", "Activity");
    for (int i = 0; i < nl.getLength(); i++) {
        Element elem = (Element)nl.item(i);
        node = elem.getElementsByTagNameNS(
            "urn:xmlmaster:VISIT", "DateTime").item(0);
        System.out.print("Not Visited:");
        System.out.print(node.getFirstChild().getNodeValue());
        System.out.print(" Worker=" + name);
        node = elem.getElementsByTagNameNS(
            "urn:xmlmaster:CONTACT", "Name").item(0);
        System.out.print(", Visit (");
        System.out.print(node.getFirstChild().getNodeValue());
        System.out.println(")");
    }
}
```

- A. Not Visited:2004-07-01 13:00 Worker=Jeff Workman, Visit(John Smith) Not Visited:2004-07-01 16:00 Worker=Jeff Workman, Visit(Ken Miller)
- B. Not Visited:2004-07-01 13:00 Worker=John Smith, Visit(John Smith) Not Visited:2004-07-01 16:00 Worker=John Smith, Visit(Ken Miller)
- C. Not Visited:2004-07-01 13:00 Worker=Jeff Workman, Visit(John Smith)
- D. Not Visited:2004-07-01 13:00 Worker=John Smith, Visit(John Smith)

Correct Answer: B

QUESTION 4

Which of the following is a correct statement about XML security?

- A. To encrypt an XML document using XML Encryption, the XML document must already have been normalized (normalization via Canonical XML, etc.)
- B. To attach a digital signature to an XML document using XML-Signature, the XML document must already have been normalized (normalized via Canonical XML, etc.)
- C. SSL cannot be used as security when sending an XML document
- D. To encrypt an XML document using XML Encryption, character encoding for the XML document must be UTF-16

Correct Answer: B

QUESTION 5



When processing the following "XML Document" according to the method shown by "SAX Processing," which of the following correctly describes the output results (print method output)?

[XML Document]

```
<doc><content>East<![CDATA[ &]]>West</content></doc>
```

[SAX Processing]

Use the following "ContentHandlerImpl" class, and parse the XML Document using SAX. Assume no execution errors.

```
class ContentHandlerImpl extends DefaultHandler {  
    public void characters(char[] ch, int start, int length) throws SAXException {  
        System.out.print(new String(ch, start, length));  
    }  
}
```

- A. EastWest
- B. EastandWest
- C. EastWest
- D. East

Correct Answer: B

[I10-002 PDF Dumps](#)

[I10-002 VCE Dumps](#)

[I10-002 Study Guide](#)