

100% Money Back
Guarantee

Vendor:XML Master

Exam Code:I10-002

Exam Name:XML Master: Professional V2

Version:Demo

QUESTION 1

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="2"/>
      <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 2

Push the Exhibit Button to load the referenced "testml.xsd".

```

[testml.xsd]
<xs:schema
xmlns:xs="http://www.w3.org/2001/XMLSchema">
  <xs:element name="TestML" type="testmlType" />
  <xs:complexType name="testmlType">
    <xs:sequence>
      <xs:element ref="person"
maxOccurs="unbounded" />
    </xs:sequence>
  </xs:complexType>

  <xs:element name="person" type="personType" />
  <xs:complexType name="personType">
    <xs:sequence>
      <xs:element ref="name" />
      <xs:element ref="phone" />
    </xs:sequence>
  </xs:complexType>

  <xs:element name="name" type="xs:string" />
  <xs:element name="phone" type="xs:string" />
</xs:schema>

```

Assume that "testml.xsd" is defined. Without rewriting this XML Schema Document ("testml.xsd"), create a new, separate XML Schema Document to partially change the schema definition to write a cellPhone element as a child element of the person element. As a result, the following "XML Document" will be valid against the new schema. Which of the following correctly describes the new XML Schema Document? Assume the XML parser correctly processes the XML schema schemaLocation attribute.

```
[XML Document]
<TestML>
  <person>
    <name>John Smith</name>
    <phone>03-0000-9999</phone>
    <cellPhone>000-1111-2222</cellPhone>
  </person>
</TestML>
```

- A `<xs:schema xmlns:xs="http://www.w3.org/2001/XMLSchema">`
 `<xs:import schemaLocation="testml.xsd" />`
 `<xs:complexType name="personType">`
 `<xs:sequence>`
 `<xs:element ref="name" />`
 `<xs:element ref="phone" />`
 `<xs:element ref="cellPhone" />`
 `</xs:sequence>`
 `</xs:complexType>`
 `<xs:element name="cellPhone" type="xs:string" />`
`</xs:schema>`
- B `<xs:schema xmlns:xs="http://www.w3.org/2001/XMLSchema">`
 `<xs:include schemaLocation="testml.xsd" />`
 `<xs:complexType name="newPersonType" substitutionGroup="personType">`
 `<xs:sequence>`
 `<xs:element ref="name" />`
 `<xs:element ref="phone" />`
 `<xs:element ref="cellPhone" />`
 `</xs:sequence>`
 `</xs:complexType>`
 `<xs:element name="cellPhone" type="xs:string" />`
`</xs:schema>`
- C `<xs:schema xmlns:xs="http://www.w3.org/2001/XMLSchema">`
 `<xs:redefine schemaLocation="testml.xsd">`
 `<xs:complexType name="personType">`
 `<xs:complexContent>`
 `<xs:extension base="personType">`
 `<xs:sequence>`
 `<xs:element ref="cellPhone" />`
 `</xs:sequence>`
 `</xs:extension>`
 `</xs:complexContent>`
 `</xs:complexType>`
 `</xs:redefine>`
 `<xs:element name="cellPhone" type="xs:string" />`
`</xs:schema>`
- D It is not possible to implement a function of the type proposed.

A. Option A

B. Option B

C. Option C

D. Option D

Correct Answer: C

QUESTION 3

Which of the following correctly explains SOAP (SOAP 1.1) and schema?

- A. By including a document type declaration in the SOAP message, you can perform a validation check on the XML Document included in the SOAP Body element
- B. You cannot perform a validation check on the XML Document included in the SOAP Body element
- C. An XML Document encoded according to the encoding method (SOAP Encoding) defined under the SOAP 1.1 specification can only be validated against W3C XML Schema
- D. A SOAP message cannot include a DTD; however, other XML coded Schema Documents can be included as a part of the XML

Correct Answer: D

QUESTION 4

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

```
[XML Document]
<root>
  <data x="1" y="2">3</data>
  <data x="1" y="4">6</data>
  <data x="1" y="6">9</data>
  <data x="1" y="8">12</data>
</root>
```

Assume that the character "3" is obtained from the "XML document". Select which XSLT style sheet correctly performs the transformation. (Multiple answers possible. Select two.)

- A.
- B.
- C.
- D.

Correct Answer: BD

QUESTION 5

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

```

[XML Document]
<Unfinished xmlns="urn:xmlmaster:UNFINISHED">
  <Item>
    <Worker xmlns="urn:xmlmaster:WORKER"
Name="Jeff Workman"/>
    <Activity xmlns="urn:xmlmaster:VISIT"
Code="0002">
      <DateTime>2004-07-01 13:00</DateTime>
      <Job Procedure="Install air conditioner"
Time="2"/>
      <Status>WAITING</Status>
    </Activity>
  </Item>
  <Item>
    <Worker xmlns="urn:xmlmaster:WORKER" Name="Jim
Worker"/>
    <Activity xmlns="urn:xmlmaster:VISIT"
Code="0003">
      <DateTime>2004-07-01 16:00</DateTime>
      <Job Procedure="Clean air conditioner"
Time="2"/>
      <Status>WAITING</Status>
    </Activity>
  </Item>
</Unfinished>

```

When processing the "XML Document" according to the method shown by "DOM Processing," which of the following is the most appropriate expression of the results under XML 1.0? Line feeds and/or indents are not reflected in the results. Assume that the processed XML Document has no indents (ignorable white space such as line feeds, tabs, etc.).

[DOM Processing]

Create XML using the following method.

```
Document output = createXML( doc, impl );
```

The variable doc here references the Document instance of the loaded XML Document.

The variable impl here references the DOMImplementation instance.

The DOM parser is namespace aware.

Assume no execution errors.

```

public static Document createXML( Document doc, DOMImplementation impl ) {
  String name = "Jim Worker";

  Document output = impl.createDocument(
    "urn:xmlmaster:LIST", "List", null);
  Element root = output.getDocumentElement();

  NodeList nl = doc.getElementsByTagNameNS("urn:xmlmaster:WORKER", "Worker");
  for (int i = 0; i < nl.getLength(); i++) {
    Element e1 = (Element)nl.item(i);
    if (e1.getAttribute("Name").equals(name)) {
      Element elem = output.createElementNS(
        "urn:xmlmaster:LIST", "VisitList");
      root.appendChild(output.importNode(e1, true));
      Element e2 = (Element)e1.getNextSibling();
      root.appendChild(output.importNode(e2, true));
    }
  }

  return output;
}

```

A. 2004-07-01 16:00 WAITING

B. 2004-07-01 16:00 WAITING

C. C. 2004-07-01 16:00 WAITING

D. D.D. 2004-07-01 16:00 WAITING

Correct Answer: C

QUESTION 6

Which of the following correctly describes the results of normalizing the following XML Document rec element by Exclusive XML Canonicalization?

```
[XML Document]
<a:root xmlns:a="urn:xmlmaster:a">
  <b:rec xmlns:b="urn:xmlmaster:b">
    <b:field />
  </b:rec>
</a:root>
```

A.

B.

C.

D.

Correct Answer: B

QUESTION 7

Which of the following is incorrect with respect to general characteristics of DOM (Level 2) processing and data binding tool processing?

A. Data types such as integers and dates cannot be handled by DOM; however, a data binding tool can be used to handle the data type

B. A validation against the schema cannot be performed with the DOM; however, a data binding tool can be used to perform a validation against the schema

C. A DOM parser cannot be used to execute a program created to use a different DOM parser; however, a data binding tool can be used to execute a program created to use a different data binding tool

D. For a compiled programming language, compared to DOM processing, a data binding tool can detect more element name mistakes and other programming errors during compiling

Correct Answer: C

QUESTION 8

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


```
[XML Document]
<List xmlns="urn:xmlmaster:LIST">
  <Date>2004-07-01</Date>
  <Worker Name="Jeff Workman"
xmlns="urn:xmlmaster:WORKER" />
  <VisitList>
    <Activity xmlns="urn:xmlmaster:VISIT"
Code="0001">
      <DateTime>2004-07-01 09:00</DateTime>
      <Job Procedure="Install freezer" Time="2" />
      <Status>WAITING</Status>
    </Activity>
  </VisitList>
</List>
```

When processing the "XML Document" according to the method shown by "DOM Processing," which of the following is the most appropriate expression of the results under XML 1.0? Line feeds and/or indents are not reflected in the results. Although the expected processing result is choice "D", processing may not occur as expected.

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

```
public static Document createXML( Document doc, DOMImplementation impl ) {
    String code = "0001";

    Document output = impl.createDocument(
        "urn:xmlmaster:REPORT", "Report", null);
    Element root = output.getDocumentElement();
    Element src = (Element)doc.getElementsByTagName(
        "urn:xmlmaster:WORKER", "Worker").item(0);
    root.appendChild(output.importNode(src, true));

    NodeList nl = doc.getElementsByTagNameNS("urn:xmlmaster:VISIT", "Activity");
    for (int i = 0; i < nl.getLength(); i++) {
        src = (Element)nl.item(i);
        if (code.equals(src.getAttribute("Code"))) {
            Element elem = (Element)output.importNode(src, true);
            root.appendChild(elem);
            Element status = (Element)elem.getElementsByTagName(
                "Status").item(0);
            status.getFirstChild().setNodeValue("COMPLETED");
            break;
        }
    }
    return output;
}
```

- A.
- B.
- C. C.C. COMPLETED
- D. D. 2004-07-01 09:00 COMPLETED

Correct Answer: D

QUESTION 9

Use XSLT transformation to create XML according to the schema (portions modified for this question) defined by TravelXML.

Perform XSLT transformation on the "XML Document" using the following "XSLT Style Sheet". Select which of the following correctly describes the results of performing a validation check on the post-transformation XML against the schema ("XML Schema" referenced when the Exhibit Button is pushed).

```
[XML Document]
<form xmlns="urn:xmlmaster">
  <field1>Sky Star Hotel</field1>
  <field2></field2>
  <field3></field3>
</form>

[XSLT Style Sheet]
<xsl:stylesheet version="1.0"
  xmlns:xsl="http://www.w3.org/1999/XSL/Transform"
  xmlns:fm="urn:xmlmaster"
  exclude-result-prefixes="fm">
  <xsl:template match="/">
    <xsl:element name="BookingNotification"
      xmlns="http://www.xmlns.org/2003/TravelXML">
      <xsl:element name="AccommodationInformation">

        <xsl:element name="AccommodationName">
          <xsl:value-of select="fm:form/fm:field1" />
        </xsl:element>
        <xsl:if test="fm:form/fm:field2/text()">
          <xsl:element name="ChainName">
            <xsl:value-of select="fm:form/fm:field2" />
          </xsl:element>
        </xsl:if>
        <xsl:if test="fm:form/fm:field3/text()">
          <xsl:element name="AccommodationPhoneNumber">
            <xsl:value-of select="fm:form/fm:field3" />
          </xsl:element>
        </xsl:if>

      </xsl:element>
    </xsl:element>
  </xsl:template>
</xsl:stylesheet>
```

- A. The root element (document element) of the post-transformation XML is an empty element, and the post-transformation XML is not valid against the schema
- B. The post-transformation XML does not include the required elements defined by the schema (all elements defined once they have appeared, namespace reflected); therefore, is not valid
- C. The post-transformation XML includes all of the required elements defined by the schema (all elements defined once they have appeared, namespace reflected); however, is not valid
- D. The post-transformation XML is valid

Correct Answer: D

QUESTION 10

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

QUESTION 11

Which of the following is clearly an unnecessary step in procedures to create digital signature via XML-Signature?

- A. Prepare a key for signing
- B. Normalize the subject of the signature (normalization via Canonical XML, etc.)
- C. Remove namespaces in the subject of the signature
- D. Calculate a digest of the subject of the signature

Correct Answer: C

QUESTION 12

Which of the following is incorrect with respect to XML?

- A. An XML document features high data readability
- B. An XML document can be transmitted over general-use protocols such as HTTP and SMTP
- C. XML is designed with the ideal structure for storage in an RDB (relational database)
- D. Some RDBs (relational databases) can output results data in XML format

Correct Answer: C