



**Free Questions for S90.05A by go4braindumps**

**Shared by Joyce on 06-06-2022**

**For More Free Questions and Preparation Resources**

**Check the Links on Last Page**

# Question 1

---

## Question Type: MultipleChoice

---

You are working on building a new Electronic Publications service that allows users to retrieve electronic copies of online books for display on specialized electronic reading devices. This service needs to contain two operations:

1. An operation that retrieves an entire book. This operation needs to receive a message based on a pre-defined "addBookRequest" element and then reply with a message based on the "addBookAcknowledgement" element.
2. An operation that retrieves information that describes a book. This operation needs to receive a message based on a pre-defined "getBookInfoRequest" element and then reply with a message based on the "getBookInfoResponse " element.

Examples of these XML fragments based on these four elements are shown here:

```
<addBookRequest>
  <userID>123DD</userID>
  <ISBN>013555555X</ISBN>
</addBookRequest>

<addBookAcknowledgement>
  <userID>123DD</userID>
  <ISBN>013555555X</ISBN>
  <status>OK</status>
</addBookAcknowledgement>

<getBookInfoRequest>
  <userID>123DD</userID>
  <ISBN>013555555X</ISBN>
</getBookInfoRequest>

<getBookInfoResponse>
  <userID>123DD</userID>
  <ISBN>013555555X</ISBN>
  <title>XML Schema and WSDL Design</title>
  <author>Doe, John</author>
</getBookInfoResponse>
```

You are now tasked with creating an XML schema that will correctly validate these four XML fragments. Which of the following is correct?

```

C A. <xsd:schema xmlns:xsd="http://www.w3.org/2001/XMLSchema">
  <xsd:element name="addBookRequest">
    <xsd:complexType>
      <xsd:sequence>
        <xsd:element name="userID" type="xsd:string"/>
        <xsd:element name="ISBN" type="xsd:string"/>
      </xsd:sequence>
    </xsd:complexType>
  </xsd:element>
  <xsd:element name="addBookAcknowledgement">
    <xsd:complexType>
      <xsd:sequence>
        <xsd:element name="userID" type="xsd:string"/>
        <xsd:element name="ISBN" type="xsd:string"/>
        <xsd:element name="status" type="xsd:string"/>
      </xsd:sequence>
    </xsd:complexType>
  </xsd:element>
  <xsd:element name="getBookInfoRequest">
    <xsd:complexType>
      <xsd:sequence>
        <xsd:element name="userID" type="xsd:string"/>
        <xsd:element name="ISBN" type="xsd:string"/>
      </xsd:sequence>
    </xsd:complexType>
  </xsd:element>
  <xsd:element name="getBookInfoResponse">
    <xsd:complexType>
      <xsd:sequence>
        <xsd:element name="userID" type="xsd:string"/>
        <xsd:element name="ISBN" type="xsd:string"/>
        <xsd:element name="title" type="xsd:string"/>
        <xsd:element name="author" type="xsd:string"/>
      </xsd:sequence>
    </xsd:complexType>
  </xsd:element>
</xsd:schema>

```

```

C B. <xsd:schema xmlns:xsd="http://www.w3.org/2001/XMLSchema">
  <xsd:element name="genericRequest">
    <xsd:complexType>
      <xsd:sequence>
        <xsd:element name="userID" type="xsd:string"/>
        <xsd:element name="ISBN" type="xsd:string"/>
      </xsd:sequence>
    </xsd:complexType>
  </xsd:element>
  <xsd:element name="addBookAcknowledgement">
    <xsd:complexType>
      <xsd:sequence>

```



```

C C. <xsd:schema xmlns:xsd="http://www.w3.org/2001/XMLSchema">
    <xsd:element name="genericRequest">
        <xsd:complexType>
            <xsd:sequence>
                <xsd:element name="userID" type="xsd:string"/>
                <xsd:element name="ISBN" type="xsd:string"/>
            </xsd:sequence>
        </xsd:complexType>
    </xsd:element>
    <xsd:element name="addBookRequest">
        <xsd:complexType>
            <xsd:sequence>
                <xsd:element name="genericRequest" type="genericRequest"/>
            </xsd:sequence>
        </xsd:complexType>
    </xsd:element>
    <xsd:element name="addBookAcknowledgement">
        <xsd:complexType>
            <xsd:sequence>
                <xsd:element name="genericRequest" type="genericRequest"/>
                <xsd:element name="status" type="xsd:string"/>
            </xsd:sequence>
        </xsd:complexType>
    </xsd:element>
    <xsd:element name="getBookInfoRequest">
        <xsd:complexType>
            <xsd:sequence>
                <xsd:element name="genericRequest" type="genericRequest"/>
            </xsd:sequence>
        </xsd:complexType>
    </xsd:element>
    <xsd:element name="getBookInfoResponse">
        <xsd:complexType>
            <xsd:sequence>
                <xsd:element name="genericRequest" type="genericRequest"/>
                <xsd:element name="title" type="xsd:string"/>
                <xsd:element name="author" type="xsd:string"/>
            </xsd:sequence>
        </xsd:complexType>
    </xsd:element>
</xsd:schema>

```

C D. All of the above.

### Options:

---

A- Option A

B- Option B

C- Option C

D- Option D

### Answer:

---

A

## Question 2

---

### Question Type: MultipleChoice

---

You are asked to create an XML schema for an Address Formatting service that needs to be able to accept a mailing address and then return the same address rearranged in a different format defined by the postal service. You are given the following specific instructions as to how the XML schema should be designed:

\* The XML schema requires two elements named "AddressLookup" and "AddressResponse".

\* The "AddressLookup" element must contain child elements named "address1", "address2", "city", "stateOrRegion", "postalCode", "country", in that order. Each of these elements must have the type string.

\* The "AddressResponse" has the same children as the "AddressLookup" message, plus it contains an "addressValid" element as its last child. This element is added to indicate whether a valid address was found, and therefore this element must be of type Boolean.

Which of the following XML schemas fulfills the requirements while also following the instructions?



```

A. <xsd:schema xmlns:xsd="http://www.w3.org/2001/XMLSchema"
    targetNamespace="http://www.example.org/addr/">
  <xsd:element name="AddressLookup">
    <xsd:complexType>
      <xsd:sequence>
        <xsd:element name="address1" type="xsd:string"/>
        <xsd:element name="address2" type="xsd:string"/>
        <xsd:element name="city" type="xsd:string"/>
        <xsd:element name="stateOrRegion" type="xsd:string"/>
        <xsd:element name="postalCode" type="xsd:string"/>
        <xsd:element name="country" type="xsd:string"/>
      </xsd:sequence>
    </xsd:complexType>
  </xsd:element>
  <xsd:element name="AddressResponse">
    <xsd:complexType>
      <xsd:sequence>
        <xsd:element name="AddressLookup"/>
        <xsd:element name="addressValid" type="xsd:boolean"/>
      </xsd:sequence>
    </xsd:complexType>
  </xsd:element>
</xsd:schema>

```

```

C B. <xsd:schema xmlns:xsd="http://www.w3.org/2001/XMLSchema"
    targetNamespace="http://www.example.org/addr/">
  <xsd:element name="AddressLookup">
    <xsd:complexType>
      <xsd:sequence>
        <xsd:element name="AddressLookup"/>
      </xsd:sequence>
    </xsd:complexType>
  </xsd:element>
  <xsd:element name="address1" type="xsd:string"/>
  <xsd:element name="address2" type="xsd:string"/>
  <xsd:element name="city" type="xsd:string"/>
  <xsd:element name="stateOrRegion" type="xsd:string"/>
  <xsd:element name="postalCode" type="xsd:string"/>
  <xsd:element name="country" type="xsd:string"/>
  <xsd:element name="AddressResponse">
    <xsd:complexType>
      <xsd:sequence>
        <xsd:element name="AddressResponse"/>
      </xsd:sequence>
    </xsd:complexType>
  </xsd:element>
  <xsd:element name="addressValid" type="xsd:boolean"/>
</xsd:schema>

```

C C. 

```
<xsd:schema xmlns:xsd="http://www.w3.org/2001/XMLSchema"
  targetNamespace="http://www.example.org/addr/">
  <xsd:element name="AddressLookup">
    <xsd:complexType>
      <xsd:sequence>
        <xsd:element name="address1" type="xsd:string"/>
        <xsd:element name="address2" type="xsd:string"/>
        <xsd:element name="city" type="xsd:string"/>
        <xsd:element name="stateOrRegion" type="xsd:string"/>
        <xsd:element name="postalCode" type="xsd:string"/>
        <xsd:element name="country" type="xsd:string"/>
      </xsd:sequence>
    </xsd:complexType>
  </xsd:element>
  <xsd:element name="AddressResponse">
    <xsd:complexType>
      <xsd:sequence>
        <xsd:element name="address1" type="xsd:string"/>
        <xsd:element name="address2" type="xsd:string"/>
        <xsd:element name="city" type="xsd:string"/>
        <xsd:element name="stateOrRegion" type="xsd:string"/>
        <xsd:element name="postalCode" type="xsd:string"/>
        <xsd:element name="country" type="xsd:string"/>
        <xsd:element name="addressValid" type="xsd:boolean"/>
      </xsd:sequence>
    </xsd:complexType>
  </xsd:element>
</xsd:schema>
```

C D. None of the above.

## Options:

---

A- Option A

**B-** Option B

**C-** Option C

**D-** Option D

**Answer:**

---

C

## Question 3

---

**Question Type:** MultipleChoice

---

The technical architecture department contacts you to tell you that all existing WSDL definitions need to be modified to add an additional concrete description in order to bind the service to a new version of SOAP. You begin with the WSDL definition for the Notification service, as shown here:

```
<definitions name="Notification"
  targetNamespace="http://www.example.org/wsdl/notify"
  xmlns:tns="http://www.example.org/wsdl/notify"
  xmlns:not="http://www.example.org/notify"
  xmlns="http://schemas.xmlsoap.org/wsdl/"
  xmlns:xsd="http://www.w3.org/2001/XMLSchema"
  xmlns:soap="http://schemas.xmlsoap.org/wsdl/soap/">
  <types>
    <xsd:schema
      targetNamespace="http://www.example.org/wsdl/notify">
      <xsd:import namespace="http://www.example.org/notify"
        schemaLocation="notify.xsd"/>
    </xsd:schema>
  </types>
  <message name="NotificationMessage">
    <part name="Notification" element="not:notification"/>
  </message>
  <portType name="NotificationInterface">
    <operation name="Notification">
      <input message="tns:NotificationMessage"/>
    </operation>
  </portType>
  <binding name="NotificationBinding"
    type="tns:NotificationInterface">
    <soap:binding style="document"
      transport="http://schemas.xmlsoap.org/soap/http"/>
    <operation name="Notification">
      <soap:operation
        soapAction="http://www.example.org/Notification"/>
      <input>
        <soap:body use="literal"/>
      </input>
    </operation>
  </binding>
  <service name="NotificationService">
    <port binding="tns:NotificationBinding"
      name="NotificationPort">
      <soap:address
        location="http://www.example.org/notification"/>
    </port>
  </service>
</definitions>
```

Given the fact that this WSDL definition has only one "operation" element, which new elements will need to be added in order to add the concrete description described above?

**Options:**

---

- A- a 'portType' element and a 'binding' element
- B- a 'portType' element, a 'binding' element and a 'service' element
- C- a 'portType' element only
- D- None of the above.

**Answer:**

---

D

## Question 4

---

**Question Type: MultipleChoice**

---

You are working with a project team that wants to deploy the Vacation Request service that allows employees to request information regarding the vacation days they are entitled to. The project team has come up with a sample XML fragment that they would like to use as the basis for the request message that the Vacation Request service will receive, as follows:

12311K/empID>

The "vacationRequest" element (which will be placed inside a SOAP "Body" element) contains a child "empID" element that will provide the employee ID used by the Vacation Request service to perform a search. Your task is to define an XML schema for this message and to incorporate the schema into the Vacation Request WSDL definition by embedding the schema content and mapping the XML Schema elements to the appropriate WSDL elements. Which of the following correctly accomplishes this?

```

C A. <definitions name="VacationRequest"
      targetNamespace="http://www.example.org/wsdl/vacation"
      xmlns:vac="http://www.example.org/vacation"
      xmlns="http://schemas.xmlsoap.org/wsdl/"
      xmlns:xsd="http://www.w3.org/2001/XMLSchema">
  <types>
    <xsd:schema xmlns:xsd="http://www.w3.org/2001/XMLSchema"
      targetNamespace="http://www.example.org/vacation">
      <xsd:element name="vacationRequest">
        <xsd:complexType>
          <xsd:sequence>
            <xsd:element name="empID" type="xsd:string"/>
          </xsd:sequence>
        </xsd:complexType>
      </xsd:element>
    </xsd:schema>
  </types>
  <message name="RequestVacationMessage">
    <part name="RequestVacation" element="vac:vacationRequest"/>
  </message>
  ...
</definitions>

```

```

C B. <definitions name="VacationRequest"
      targetNamespace="http://www.example.org/wsdl/vacation"
      xmlns:vac="http://www.example.org/vacation"
      xmlns="http://schemas.xmlsoap.org/wsdl/"
      xmlns:xsd="http://www.w3.org/2001/XMLSchema">
  <types>
    <xsd:schema xmlns:xsd="http://www.w3.org/2001/XMLSchema"
      targetNamespace="http://www.example.org/vacation">
      <xsd:complexType name="vacationRequest">
        <xsd:sequence>
          <xsd:element name="empID" type="xsd:string"/>
        </xsd:sequence>
      </xsd:complexType>
    </xsd:schema>
  </types>
  <message name="RequestVacationMessage">
    <part name="RequestVacation" element="vac:vacationRequest"/>
  </message>
  ...
</definitions>

```

C. `<definitions name="VacationRequest" targetNamespace="http://www.example.org/wsdl/vacation" xmlns:vac="http://www.example.org/vacation" xmlns="http://schemas.xmlsoap.org/wsdl/" xmlns:xsd="http://www.w3.org/2001/XMLSchema">  
<types>  
 <xsd:schema xmlns:xsd="http://www.w3.org/2001/XMLSchema" targetNamespace="http://www.example.org/vacation">  
 <xsd:element name="vacationRequest">  
 <xsd:complexType>  
 <xsd:sequence>  
 <xsd:element name="empID" type="xsd:string"/>  
 </xsd:sequence>  
 </xsd:complexType>  
 </xsd:element>  
 </xsd:schema>  
</types>  
<message name="RequestVacationMessage">  
 <part name="RequestVacation" type="vac:vacationRequest"/>  
</message>  
 ...  
</definitions>`

D. None of the above.

## Options:

---

A- Option A

B- Option B

C- Option C

D- Option D



**Answer:**

---

A

## Question 5

---

**Question Type: MultipleChoice**

---

You are asked to create an XML schema and WSDL definition for a Customer Lookup service. The service accepts a customer name and returns the corresponding customerID. You are given the following specific instructions as to how the XML schema should be designed:

- \* Two elements named "CustomerLookup" and "CustomerResponse" are required.
- \* The "CustomerLookup" element will be used to represent the request message sent to the service. It must have a child element named "customerName" that has the type string.
- \* The "CustomerResponse" element will be used to represent the response message sent out by the service. It must have a child element named "customerID" that has the type integer.

Which of the following WSDL definitions correctly describes these messages?

```

C A. <definitions name="CustomerLookup"
      targetNamespace="http://www.example.org/wsdl/cust"
      xmlns="http://schemas.xmlsoap.org/wsdl/"
      xmlns:cust="http://www.example.org/cust"
      xmlns:xsd="http://www.w3.org/2001/XMLSchema">
  <types>
    <xsd:schema xmlns:xsd="http://www.w3.org/2001/XMLSchema"
      targetNamespace="http://www.example.org/cust">
      <xsd:element name="CustomerLookup">
        <xsd:complexType>
          <xsd:sequence>
            <xsd:element name="customerName" type="xsd:string"/>
          </xsd:sequence>
        </xsd:complexType>
      </xsd:element>
      <xsd:element name="CustomerResponse">
        <xsd:complexType>
          <xsd:sequence>
            <xsd:element name="customerID" type="xsd:integer"/>
          </xsd:sequence>
        </xsd:complexType>
      </xsd:element>
    </xsd:schema>
  </types>
  <message name="CustomerLookup">
    <part name="CustomerLookup" element="cust:CustomerLookup"/>
  </message>
  <message name="CustomerResponse">
    <part name="CustomerResponse" element="cust:CustomerResponse"/>
  </message>
  ...
</definitions>

```

```

C B. <definitions name="CustomerLookup"
      targetNamespace="http://www.example.org/wsdl/cust"
      xmlns="http://schemas.xmlsoap.org/wsdl/"
      xmlns:cust="http://www.example.org/cust"
      xmlns:xsd="http://www.w3.org/2001/XMLSchema">
  <types>
    <xsd:schema xmlns:xsd="http://www.w3.org/2001/XMLSchema"
      targetNamespace="http://www.example.org/cust">
      <xsd:element name="CustomerLookup">
        <xsd:complexType>
          <xsd:sequence>
            <xsd:element name="customerName" type="xsd:string"/>
          </xsd:sequence>
        </xsd:complexType>
      </xsd:element>
      <xsd:element name="CustomerResponse">
        <xsd:complexType>

```

C C. 

```
<definitions name="CustomerLookup"
  targetNamespace="http://www.example.org/wsdl/cust"
  xmlns="http://schemas.xmlsoap.org/wsdl/"
  xmlns:cust="http://www.example.org/cust"
  xmlns:xsd="http://www.w3.org/2001/XMLSchema">
  <types>
    <xsd:schema xmlns:xsd="http://www.w3.org/2001/XMLSchema"
      targetNamespace="http://www.example.org/cust">
      <xsd:element name="CustomerLookup">
        <xsd:complexType>
          <xsd:sequence>
            <xsd:element name="customerName" type="xsd:string"/>
          </xsd:sequence>
        </xsd:complexType>
      </xsd:element>
      <xsd:element name="CustomerResponse">
        <xsd:complexType>
          <xsd:sequence>
            <xsd:element name="customerID" type="xsd:integer"/>
          </xsd:sequence>
        </xsd:complexType>
      </xsd:element>
    </xsd:schema>
  </types>
  <message name="CustomerLookup">
    <part name="CustomerLookup" element="cust:customerName"/>
  </message>
  <message name="CustomerResponse">
    <part name="CustomerResponse" element="cust:customerID"/>
  </message>
  ...
</definitions>
```

C D. None of the above.

**Options:**

---

A- Option A

B- Option B

C- Option C

D- Option D

**Answer:**

---

A

## Question 6

---

**Question Type:** MultipleChoice

---

You are asked to create an XML schema for an Employee service that is required to search an HR database in order to retrieve the maximum number of hours an employee is authorized to work within a week. You are given the following specific instructions as to how the XML schema should be designed:

- \* Add two complex types wrapped in "element" elements named "EmployeeHoursRequestType" and "EmployeeHoursResponseType"
- \* The "EmployeeHoursRequestType" element must contain a child "ID" element of type integer. The "EmployeeHoursResponseType" must contain a child "ID" element of type integer, and a child "WeeklyHoursLimit" element of type short.
- \* The "xsd" prefix needs to be used for all XML Schema language elements.

Which of the following XML schemas fulfills the requirements while also following the instructions?

```
C A. <xsd:schema xmlns:xsd="http://www.w3.org/2001/XMLSchema"
      targetNamespace="http://www.example.org/schema/accounting/">
  <xsd:element name="EmployeeHoursRequestType">
    <xsd:complexType>
      <xsd:sequence>
        <xsd:element name="ID" type="xsd:integer"/>
      </xsd:sequence>
    </xsd:complexType>
  </xsd:element>
  <xsd:element name="EmployeeHoursResponseType">
    <xsd:complexType>
      <xsd:sequence>
        <xsd:element name="ID" type="xsd:integer"/>
        <xsd:element name="WeeklyHoursLimit" type="xsd:int"/>
      </xsd:sequence>
    </xsd:complexType>
  </xsd:element>
</xsd:schema>
```

```
C B. <xsd:schema xmlns:xsd="http://www.w3.org/2001/XMLSchema"
      targetNamespace="http://www.example.org/schema/accounting/">
  <xsd:complexType name="EmployeeHoursRequestType">
    <xsd:sequence>
      <xsd:element name="ID" type="xsd:integer"/>
    </xsd:sequence>
  </xsd:complexType>
  <xsd:complexType name="EmployeeHoursResponseType">
    <xsd:sequence>
      <xsd:element name="ID" type="xsd:integer"/>
      <xsd:element name="WeeklyHoursLimit" type="xsd:short"/>
    </xsd:sequence>
  </xsd:complexType>
</xsd:schema>
```

```
C C. <xsd:schema xmlns:xsd="http://www.w3.org/2001/XMLSchema"
      targetNamespace="http://www.example.org/schema/accounting/">
  <xsd:element name="EmployeeHoursRequestType">
    <xsd:complexType>
      <xsd:sequence>
        <xsd:element name="ID" type="xsd:integer"/>
      </xsd:sequence>
    </xsd:complexType>
  </xsd:element>
  <xsd:element name="EmployeeHoursResponseType">
    <xsd:complexType>
      <xsd:sequence>
        <xsd:element name="ID" type="xsd:integer"/>
        <xsd:element name="WeeklyHoursLimit" type="xsd:short"/>
      </xsd:sequence>
    </xsd:complexType>
  </xsd:element>
</xsd:schema>
```

### Options:

---

A- Option A

B- Option B

C- Option C

D- Option D

### Answer:

---

C

## Question 7

---

### Question Type: MultipleChoice

---

You are working with an Authentication service that submits user credentials for authentication purposes. Atypical XML document passed to the service looks like this:

```
<user>
  <date>2005-12-05</date>
  <login>jsmith</login>
  <password>1B45D33E</password>
</user>
```

You are contacted by the IT department manager and told that from now on, all XML documents that are passed to and from services must have a namespace in order to more easily identify their related XML schema definitions. You revise the above sample document by adding a namespace declaration, as well as prefixes to the "user" element, as shown here:

```
<user:user xmlns:user="http://www.example.org/user">  
  <date>2005-12-05</date>  
  <login>jsmith</login>  
  <password>1B45D33E</password>  
</user:user>
```

Assuming the "date" element is of type date, the "login" element is of type string, and the "password" element is of type hexBinary, which of the following XML schemas will correctly validate the revised sample?



- C A. 

```
<xsd:schema xmlns:xsd="http://www.w3.org/2001/XMLSchema"
  targetNamespace="http://www.example.org/user">
  <xsd:element name="user">
    <xsd:complexType>
      <xsd:sequence>
        <xsd:element name="date" type="xsd:date"/>
        <xsd:element name="login" type="xsd:string"/>
        <xsd:element name="password" type="xsd:hexBinary"/>
      </xsd:sequence>
    </xsd:complexType>
  </xsd:element>
</xsd:schema>
```
- C B. 

```
<xsd:schema xmlns:xsd="http://www.w3.org/2001/XMLSchema"
  xmlns="http://www.example.org/user">
  <xsd:element name="user">
    <xsd:complexType>
      <xsd:sequence>
        <xsd:element name="date" type="xsd:date"/>
        <xsd:element name="login" type="xsd:string"/>
        <xsd:element name="password" type="xsd:hexBinary"/>
      </xsd:sequence>
    </xsd:complexType>
  </xsd:element>
</xsd:schema>
```
- C C. 

```
<xsd:schema xmlns:xsd="http://www.w3.org/2001/XMLSchema"
  xmlns:user="http://www.example.org/user">
  <xsd:element name="user">
    <xsd:complexType>
      <xsd:sequence>
        <xsd:element name="date" type="xsd:date"/>
        <xsd:element name="login" type="xsd:string"/>
        <xsd:element name="password" type="xsd:hexBinary"/>
      </xsd:sequence>
    </xsd:complexType>
  </xsd:element>
</xsd:schema>
```
- C D. None of the above.

**Options:**

---

**A-** Option A

**B-** Option B

**C-** Option C

**D-** Option D

**Answer:**

---

A

**To Get Premium Files for S90.05A Visit**

**<https://www.p2pexams.com/products/s90.05a>**

**For More Free Questions Visit**

**<https://www.p2pexams.com/arcitura-education/pdf/s90.05a>**

