➢ **Vendor:** **Cisco**

➢ **Exam Code:** **300-835**

➢ **Exam Name:** **Implementing Automation for Cisco Collaboration Solutions**

➢ **New Updated Questions from** **Braindump2go** **(Updated in** **Sep/2020**)

Visit Braindump2go and Download Full Version 300-835 Exam Dumps

QUESTION 16
Refer to the exhibit. An administrator is creating a script using the Python xAPI over WebSockets(pyxows) library. The goal of this script is to monitor the volume of the endpoint and set the volume to 60 whenever the volume has been set higher than that amount.
Which code snippet accomplishes this task when it is added?

```
import xows
import asyncio

async def start(ip, usr, pw):
    async with xows.XoWSClient(ip,username=usr, password=pw) as client:
        async def callback(data, id_):
            if id_ == 0:
                <SNIPPET>

        await client.subscribe(['Status', 'Audio', 'Volume'], callback, True)
        await client.wait_until_closed()

asyncio.run(start('10.10.10.1', 'admin', 'T357c45e'))
```

A. `if data ['Status']['Audio']['Volume'] > 60:`
    `   await client.xCommand(['Audio', 'Volume', 'Set'], Level=60)`

B. `if data ['Status']['Audio']['Volume'] > 60:`
    `   await client.xGet(['Status', 'Audio', 'Volume']))`

C. `if data ['Status']['Audio']['Volume'] > 60:`
    `   await client.subscribe(['Status', 'Audio', 'Volume'].callback, Level=60))`

D. `if data ['Status']['Audio']['Volume'] > 60:`
    `   await client.xSet(['Configuration', 'Audio', 'DefaultVolume'], 60))`

Answer: A

QUESTION 17
Which two Webex technologies can be used to embed voice and video collaboration into a company web page? (Choose two.)

A. Webex Teams Widgets
B. Webex Teams Browser SDK
C. Webex Meetings XML API
D. Webex Teams REST API
E. Webex Teams Java SDK

Answer: AC

QUESTION 18
Which statement is true for JavaScript Macros deployed on a Cisco Collaboration room device?

A. Macros can register to receive any event that is local to the device.
B. There is no hard limit to the number of macros that can be deployed.
C. A macro can store data using a local JavaScript datastore.
D. Macros execute in the order defined in the Macro Editor.

Answer: C

QUESTION 19
When the behavior of a Cisco collaboration device is customized, which use case requires an external control system because implementing JavaScript macro does not suffice?

A. Add a Join Webex meeting button to the touch panel.
B. Move the shutters up and down.
C. Trigger a "room-reset" to restore default configurations.
D. Implement an in-room control panel for speed-dialing.

Answer: D

QUESTION 20
Refer to the exhibit. This Python script uses the websocket-based, xAPI library "pyxows' to monitor and print event details to the console when users interact with UI Extension/In-Room Control `Action Buttons' deployed to the Touch-10 interface of a Webex room device.
Which two code snippets successfully capture such events? (Choose two.)

300-835 Exam Dumps **300-835 Exam Questions** **300-835 PDF Dumps** **300-835 VCE Dumps**

```
import xows
import asyncio

async def start(ip, usr, pw):
    async with xows.XoWSClient(ip, username=usr, password=pw) as client:
        async def callback(data, id_):
            print(f'Feedback {id}(Id {id_}): {data}')


        await client.wait_until_closed()

async def task():
    await start('10.10.10.1, 'admin', 'T357c45e')

asyncio.run(task())
```

A. await client.xEvent([`Event', `**'], callback)
B. await client.subscribe([`Event', `ActionButton', `Clicked'], callback)
C. await client.subscribe([`Event', `**'], callback)
D. await client.xEvent([`Event', `UserInterface', `Extensions', `Panel', `Clicked'], callback)
E. await client.subscribe([`Event', `UserInterface', `Extensions', `Panel', `Clicked'], callback)

Answer: DE

QUESTION 21
Refer to the exhibit. The exhibit shows the XML schema to associate a device for the AddUser operation in AXL. Which XML snippet is correct to associate a device with name "SEP151515151515"?

```
<xsd:element name="associatedDevices" minOccurs="0" maxOccurs="1">
  <xsd:annotation>
    <xsd:documentation>Also known as controlled devices.</xsd:documentation>
  </xsd:annotation>
  <xsd:complexType>
    <xsd:sequence minOccurs="0">
      <xsd:element name="device" type="axlapi:String50" minOccurs="0" maxOccurs="unbounded">
        <xsd:annotation>
          <xsd:documentation>Name of device.</xsd:documentation>
        </xsd:annotation>
      </xsd:element>
    </xsd:sequence>
  </xsd:complexType>
</xsd:element>
<xsd:element name="primaryExtension" minOccurs="0" maxOccurs="1">
  <xsd:annotation>
    <xsd:documentation>Only DNs associated with the associatedDevices of the User can be used as primary extension.</xsd:documentation>
  </xsd:annotation>
  <xsd:complexType>
    <xsd:sequence minOccurs="0">
      <xsd:element name="pattern" type="axlapi:String255" minOccurs="0" maxOccurs="1">
        <xsd:annotation>
          <xsd:documentation>DN or Pattern.</xsd:documentation>
        </xsd:annotation>
      </xsd:element>
      <xsd:element name="routePartitionName" type="axlapi:String50" minOccurs="0" maxOccurs="1">
        <xsd:annotation>
          <xsd:documentation>Route Partition to which the DN is associated.</xsd:documentation>
        </xsd:annotation>
      </xsd:element>
    </xsd:sequence>
  </xsd:complexType>
</xsd:element>
```

A. `<associatedDevices>`
     `<device name="SEP151515151515">`
   `<associatedDevices>`

B. `<routePartitionName>`
     `<device>SEP151515151515</device>`
   `</routePartitionName>`

C. `<routePartitionName>`
     `<device name="SEP151515151515">`
   `</routePartitionName>`

D. `<associatedDevices>`
     `<device>SEP151515151515</device>`
   `<associatedDevices>`

Answer: D

QUESTION 22
Refer to the exhibit. This exhibit is based on the simple Python script to update a Webex registered room device (local IP = 10.10.20.153) with a new custom half-wake message.
What is the URL to complete the requests.post() method?

```
import requests

requests.post('[          ]',
              verify = False,
              auth = ( 'integrator', 'integrator' ),
              headers = { 'Content-Type': 'text/xml' },
              data = '''<Configuration>
                          <UserInterface>

                              <CustomMessage>New Welcome Message</CustomMessage>

                          </UserInterface>

                        </Configuration>'''
)
```

A. http://10.10.20.153:22/
B. http://10.10.20.153/jsxapi

C. http://api.webex.com/v1/ui
D. http://10.10.20.153/putxml

Answer: D

QUESTION 23
An In-Room Control Panel can be configured as "global" (always available). Which order panel type is supported?

A. Background
B. Do Not Disturb
C. Never
D. Out-of-Call

Answer: D

QUESTION 24
Refer to the exhibit. This exhibit is a simple Python scrip to monitor call activity for a local Webex room device. Which code snippet listens for call events and completes the script?

```
from pyxows import xows
import asyncio

async def main():

    async with xows.XoWSClient( '10.10.20.153', 'integrator', 'integrator' ) as client:

        async def callback( data, id ):
            print( f'Call status event: { data }' )

        await client.wait_until_closed()

asyncio.run( main() )
```

A. await client.xfeedback([`Status', `Call', `Status'], callback, False)
B. await client.subscribe([`Feedback', `Call', `Status'], callback, False)
C. await client.subscribe([`Status', `Call', `Status'], callback, False)
D. await client.xfeedback([`Feedback', `Call', `Status'], callback, False)

Answer: C

QUESTION 25
Which two scenarios could be automated by combining the Webex Meetings XML API with other API-enabled systems? (Choose two.)

A. scheduling new-hire orientation meetings as part of an HR system workflow
B. triggering Webex meeting recording when the host disconnect from the meeting.
C. automatically launching the weekly Webex scrum meeting on Mondays at 9 AM
D. reassigning an employee's scheduled Webex meeting to their manager then they leave the company
E. muting users in a Webex meeting when their Cisco Jabber presence status transitions to Away

Answer: AE

QUESTION 26
Refer to the exhibit. Alice is a Webex Meetings admin for her organization. Which code snippet completes a Webex Meetings XML API request that returns all users?

```
<?xml version="1.0" encoding = "UTF-8"?>
<serv:message xmlns:serv="http://www.webex.com/schemas/2002/06/service"
        xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance">
    <header>
        <securityContext>
            <siteName>apidemoeu</siteName>
            <webExID>alice@example.com</webExID>
            <sessionTicket>AAABb6TooBMAABUYA0gAKEgyU0sAAAAE9FpXDKRKO</sessionTicket>
        </securityContext>
    </header>
    <body>


    </body>
</serv:message>
```

A. ```
<bodyContent xsi:type=<java:com.webex.service.binding.user.GetUser">
</bodyContent>
```
B.  ```
<bodyContent xsi:type=<java:com.webex.service.binding.user.LstsummaryUser">
</bodyContent>
```
C. ```
<bodyContent xsi:type=<java:com.webex.service.binding.user.GetUser">
</bodyContent>
```
D.  ```
<bodyContent xsi:type=<java:com.webex.service.binding.user.LstsummaryUser">
    <webExID>*</webExID>
</bodyContent>
```

Answer: A