300-835$^{Q\&As}$

Automating Cisco Collaboration Solutions (CLAUTO)

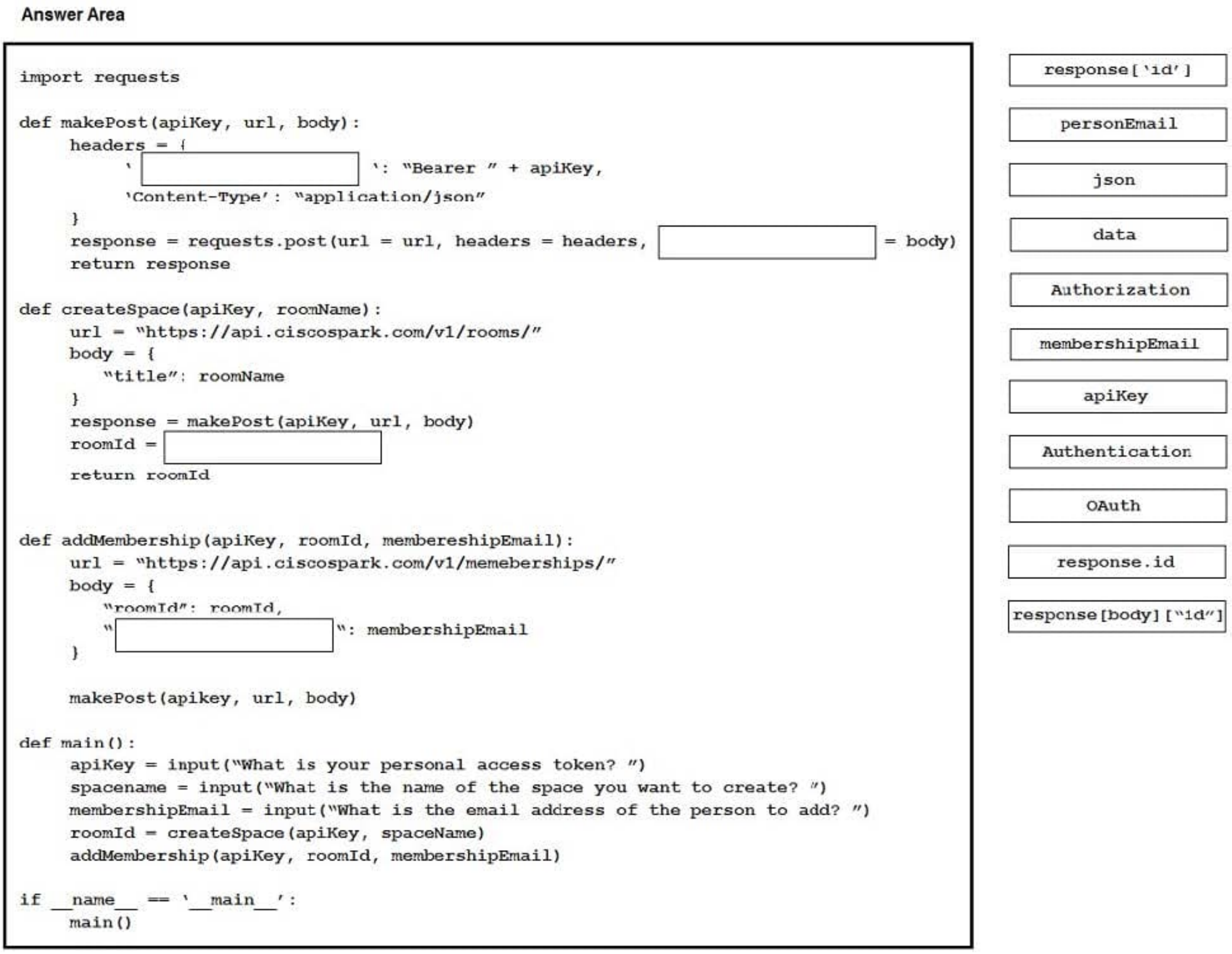
Pass Cisco 300-835 Exam with 100% Guarantee

QUESTION 1

DRAG DROP

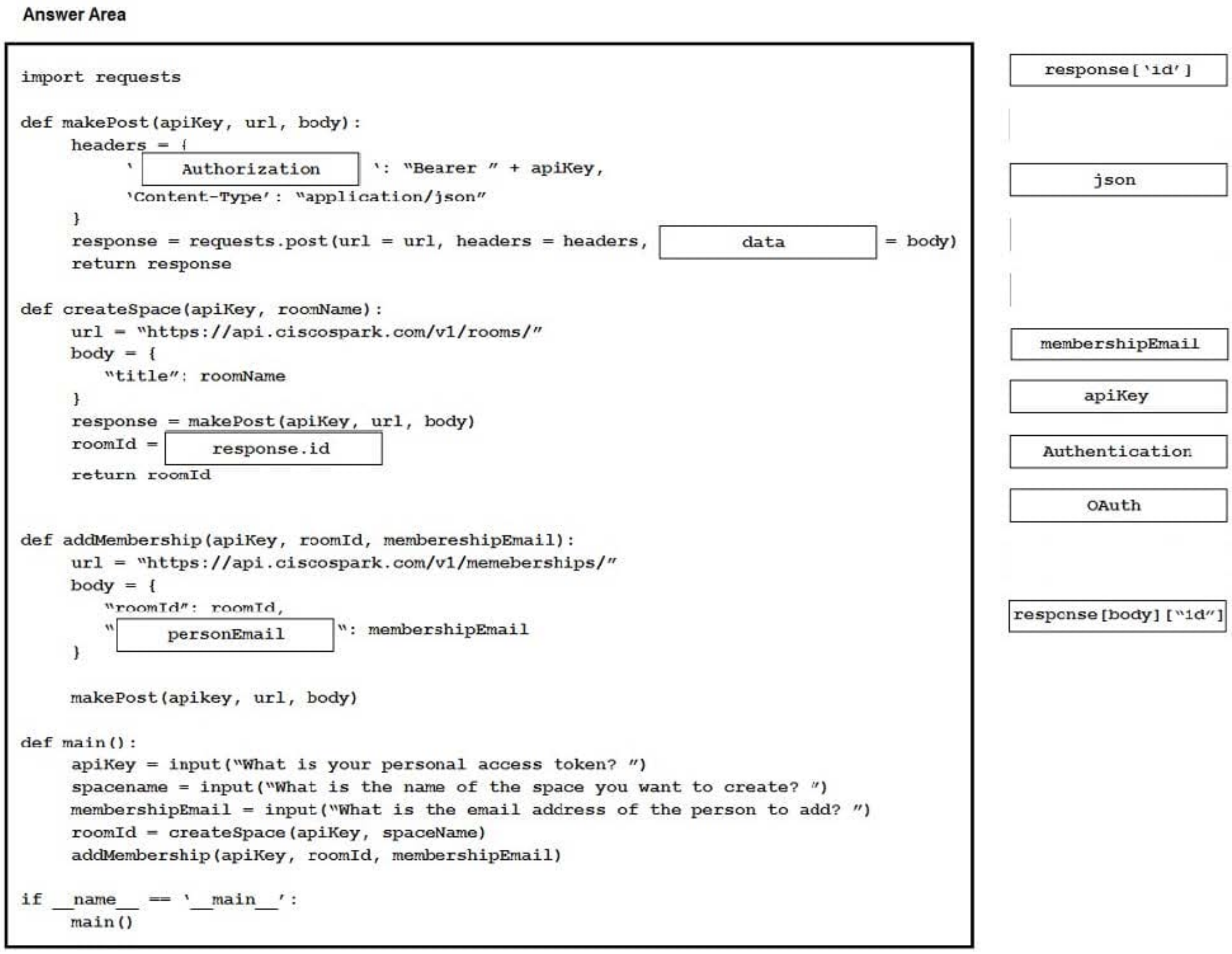
Drag and drop the correct items from the right to complete this Python script to automate the creation of Cisco Webex Teams spaces and memberships. Not all options are used.

Select and Place:



```
import requests

def makePost(apiKey, url, body):
     headers = {
          ‘ □ ‘: “Bearer ” + apiKey,
          ‘Content-Type’: “application/json”
     }
     response = requests.post(url = url, headers = headers, □ = body)
     return response

def createSpace(apiKey, roomName):
     url = “https://api.ciscospark.com/v1/rooms/”
     body = {
        “title”: roomName
     }
     response = makePost(apiKey, url, body)
     roomId = □
     return roomId


def addMembership(apiKey, roomId, membereshipEmail):
     url = “https://api.ciscospark.com/v1/memeberships/”
     body = {
        “roomId”: roomId,
        “ □ “: membershipEmail
     }

     makePost(apikey, url, body)

def main():
     apiKey = input(“What is your personal access token? ”)
     spacename = input(“What is the name of the space you want to create? ”)
     membershipEmail = input(“What is the email address of the person to add? ”)
     roomId = createSpace(apiKey, spaceName)
     addMembership(apiKey, roomId, membershipEmail)

if __name__ == ‘__main__’:
     main()
```

Correct Answer:



Answer Area

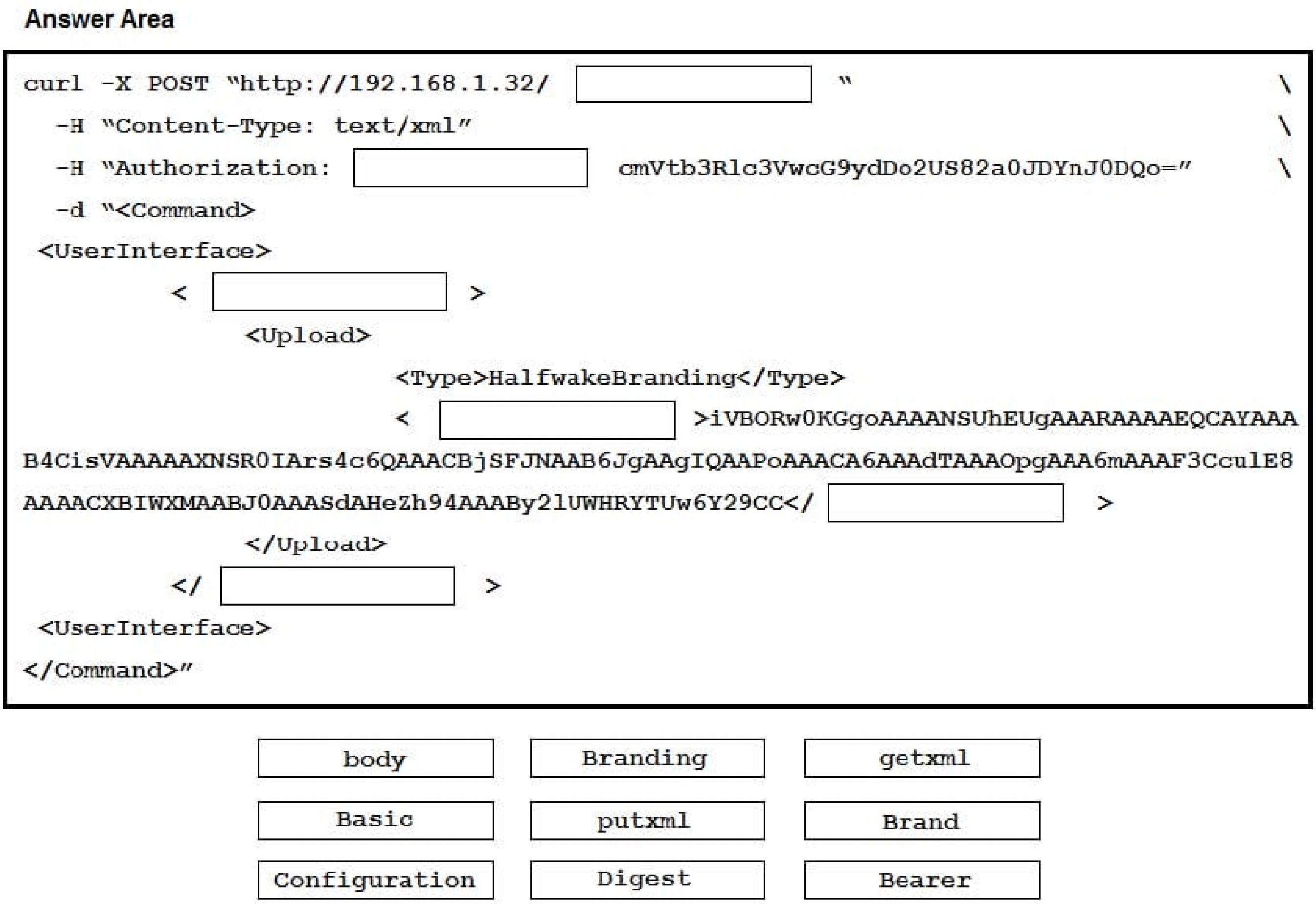
```
import requests

def makePost(apiKey, url, body):
     headers = {
          ' Authorization ': "Bearer " + apiKey,
          'Content-Type': "application/json"
     }
     response = requests.post(url = url, headers = headers, data = body)
     return response

def createSpace(apiKey, roomName):
     url = "https://api.ciscospark.com/v1/rooms/"
     body = {
        "title": roomName
     }
     response = makePost(apiKey, url, body)
     roomId = response.id
     return roomId


def addMembership(apiKey, roomId, membereshipEmail):
     url = "https://api.ciscospark.com/v1/memeberships/"
     body = {
        "roomId": roomId,
        " personEmail ": membershipEmail
     }

     makePost(apikey, url, body)

def main():
     apiKey = input("What is your personal access token? ")
     spacename = input("What is the name of the space you want to create? ")
     membershipEmail = input("What is the email address of the person to add? ")
     roomId = createSpace(apiKey, spaceName)
     addMembership(apiKey, roomId, membershipEmail)

if __name__ == '__main__':
     main()
```

QUESTION 2

DRAG DROP

Drag and drop the code to complete the curl command that creates a halfwake brand logo using the xAPI HTTP API. Not all options are used. Code can be used more than one time.

Select and Place:

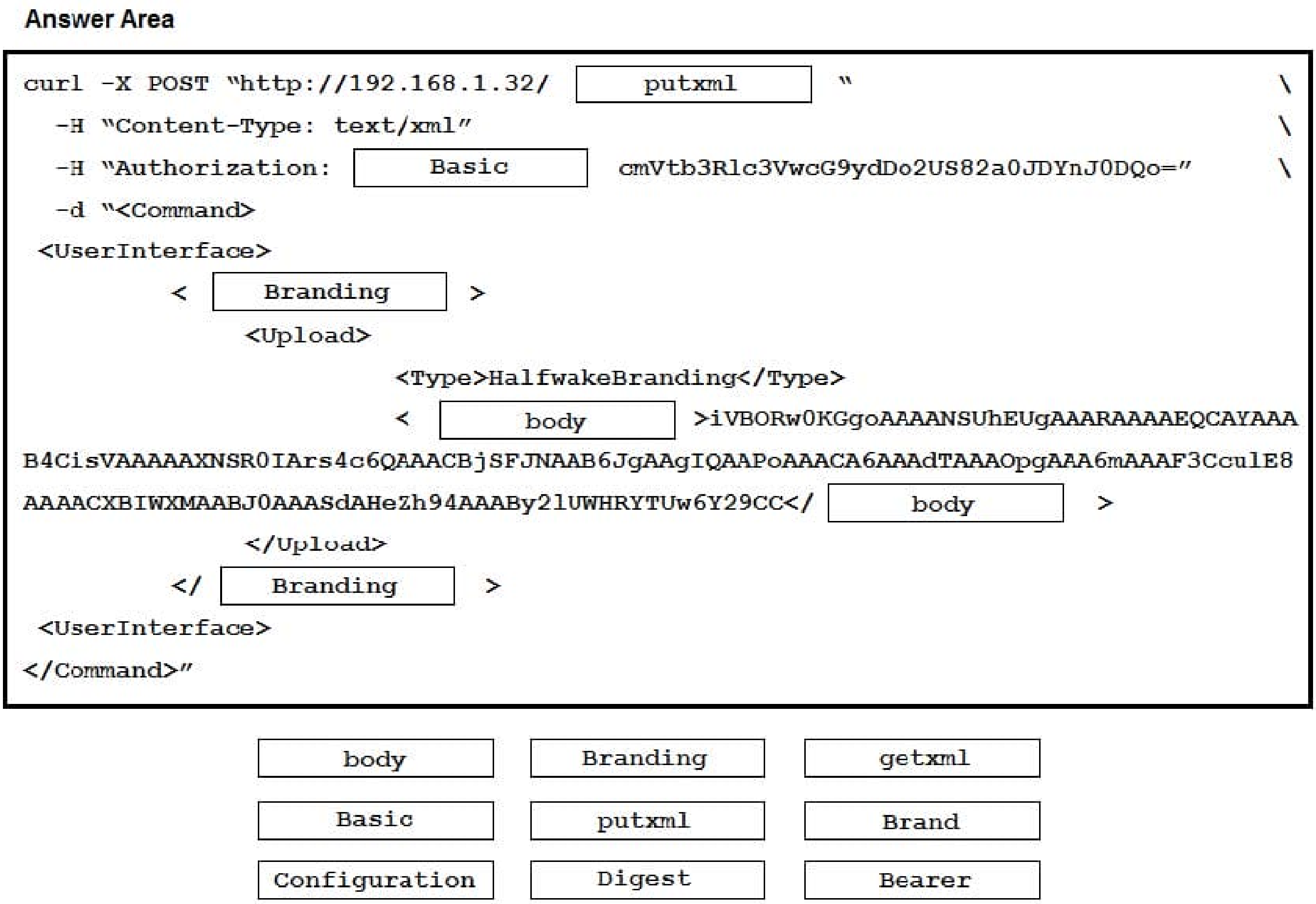


Answer Area

```
curl -X POST "http://192.168.1.32/ □ "                                          \
  -H "Content-Type: text/xml"                                                   \
  -H "Authorization: □ cmVtb3Rlc3VwcG9ydDo2US82a0JDYnJ0DQo="                     \
  -d "<Command>
 <UserInterface>
          < □ >
               <Upload>
                         <Type>HalfwakeBranding</Type>
                         < □ >iVBORw0KGgoAAAANSUhEUgAAARAAAAEQCAYAAA
B4CisVAAAAAXNSR0IArs4c6QAAACBjSFJNAAB6JgAAgIQAAPoAAACA6AAAdTAAAOpgAAA6mAAAF3CculE8
AAAACXBIWXMAABJ0AAASdAHeZh94AAABy2lUWHRYTUw6Y29CC</ □ >
               </Upload>
          </ □ >
 <UserInterface>
</Command>"
```

body	Branding	getxml
Basic	putxml	Brand
Configuration	Digest	Bearer

Correct Answer:

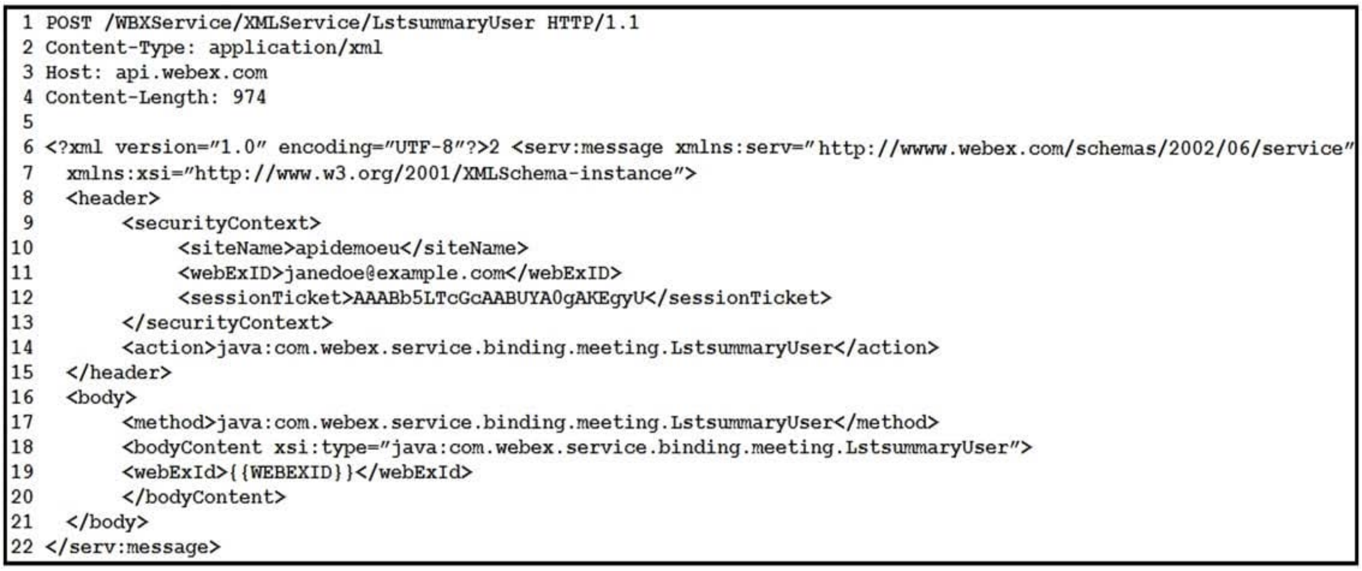


Answer Area

```
curl -X POST "http://192.168.1.32/ [ putxml ] "                              \
  -H "Content-Type: text/xml"                                                \
  -H "Authorization: [ Basic ] cmVtb3Rlc3VwcG9ydDo2US82a0JDYnJ0DQo="         \
  -d "<Command>
 <UserInterface>
         < [ Branding ] >
              <Upload>
                        <Type>HalfwakeBranding</Type>
                        < [ body ] >iVBORw0KGgoAAAANSUhEUgAAARAAAAEQCAYAAA
B4CisVAAAAAXNSR0IArs4c6QAAACBjSFJNAAB6JgAAgIQAAPoAAACA6AAAdTAAAOpgAAA6mAAAF3CculE8
AAAACXBIWXMAABJ0AAASdAHeZh94AAABy21UWHRYTUw6Y29CC</ [ body ] >
              </Upload>
         </ [ Branding ] >
 <UserInterface>
</Command>"
```

body	Branding	getxml
Basic	putxml	Brand
Configuration	Digest	Bearer

QUESTION 3



```
 1 POST /WBXService/XMLService/LstsummaryUser HTTP/1.1
 2 Content-Type: application/xml
 3 Host: api.webex.com
 4 Content-Length: 974
 5
 6 <?xml version="1.0" encoding="UTF-8"?>2 <serv:message xmlns:serv="http://wwww.webex.com/schemas/2002/06/service"
 7   xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance">
 8   <header>
 9        <securityContext>
10             <siteName>apidemoeu</siteName>
11             <webExID>janedoe@example.com</webExID>
12             <sessionTicket>AAABb5LTcGcAABUYA0gAKEgyU</sessionTicket>
13        </securityContext>
14        <action>java:com.webex.service.binding.meeting.LstsummaryUser</action>
15   </header>
16   <body>
17        <method>java:com.webex.service.binding.meeting.LstsummaryUser</method>
18        <bodyContent xsi:type="java:com.webex.service.binding.meeting.LstsummaryUser">
19        <webExId>{{WEBEXID}}</webExId>
20        </bodyContent>
21   </body>
22 </serv:message>
```

Refer to the exhibit. A Webex Meetings XML API HTTP request message with several invalid portions is shown. Which reference points to a line in the exhibit that correctly indicates that this is a LstsummaryUser request?

A. line 17

B. line 1

C. line 18

D. line 14

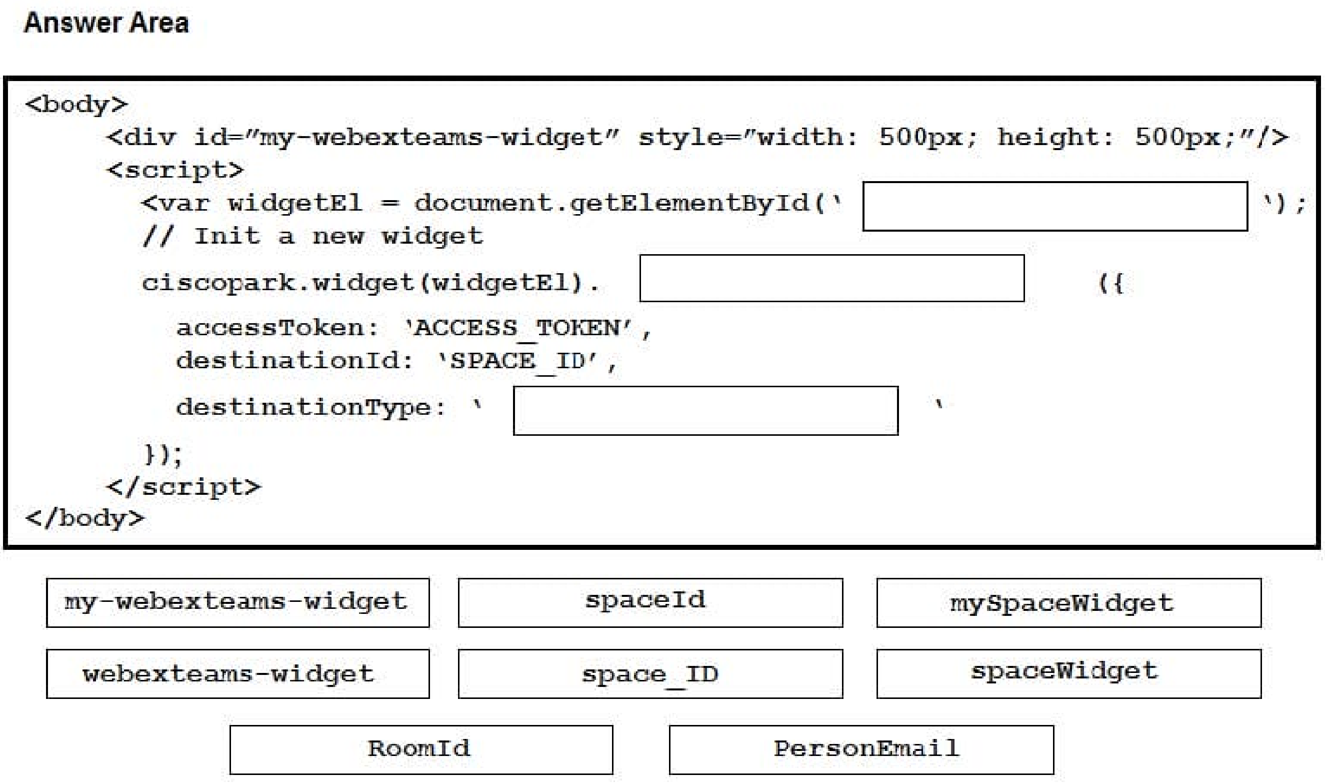
Correct Answer: B

QUESTION 4

DRAG DROP

Drag and drop the code to add a Webex Teams Space Widget in an existing HTML page. Not all items are used.

Select and Place:



Answer Area

```
<body>
     <div id="my-webexteams-widget" style="width: 500px; height: 500px;"/>
     <script>
       <var widgetEl = document.getElementById('                      ');
       // Init a new widget
       ciscopark.widget(widgetEl).                      ({
         accessToken: 'ACCESS_TOKEN',
         destinationId: 'SPACE_ID',
         destinationType: '                      '
       });
     </script>
</body>
```

my-webexteams-widget

spaceId

mySpaceWidget

webexteams-widget

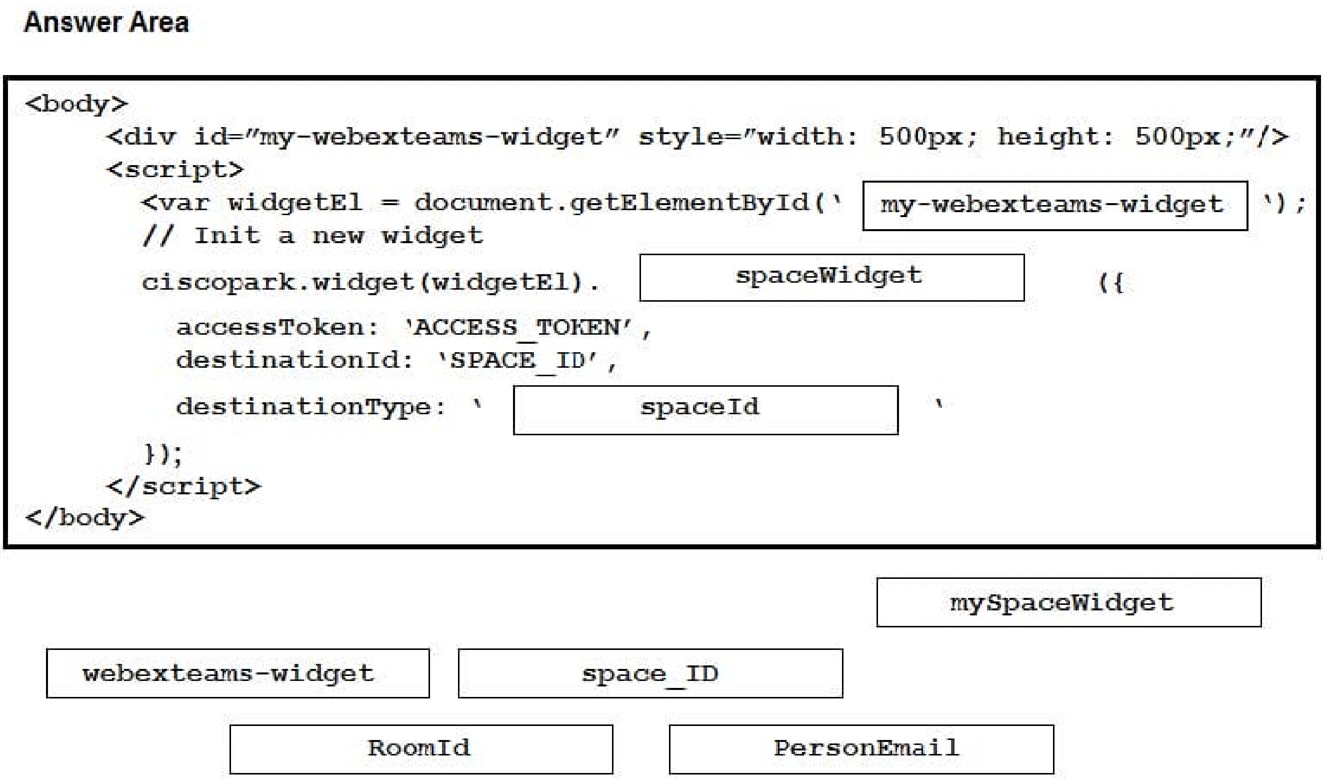
space_ID

spaceWidget

RoomId

PersonEmail

Correct Answer:



Answer Area

```
<body>
    <div id="my-webexteams-widget" style="width: 500px; height: 500px;"/>
    <script>
      <var widgetEl = document.getElementById(' my-webexteams-widget ');
      // Init a new widget
      ciscopark.widget(widgetEl). spaceWidget ({
        accessToken: 'ACCESS_TOKEN',
        destinationId: 'SPACE_ID',
        destinationType: ' spaceId '
      });
    </script>
</body>
```

mySpaceWidget

webexteams-widget

space_ID

RoomId

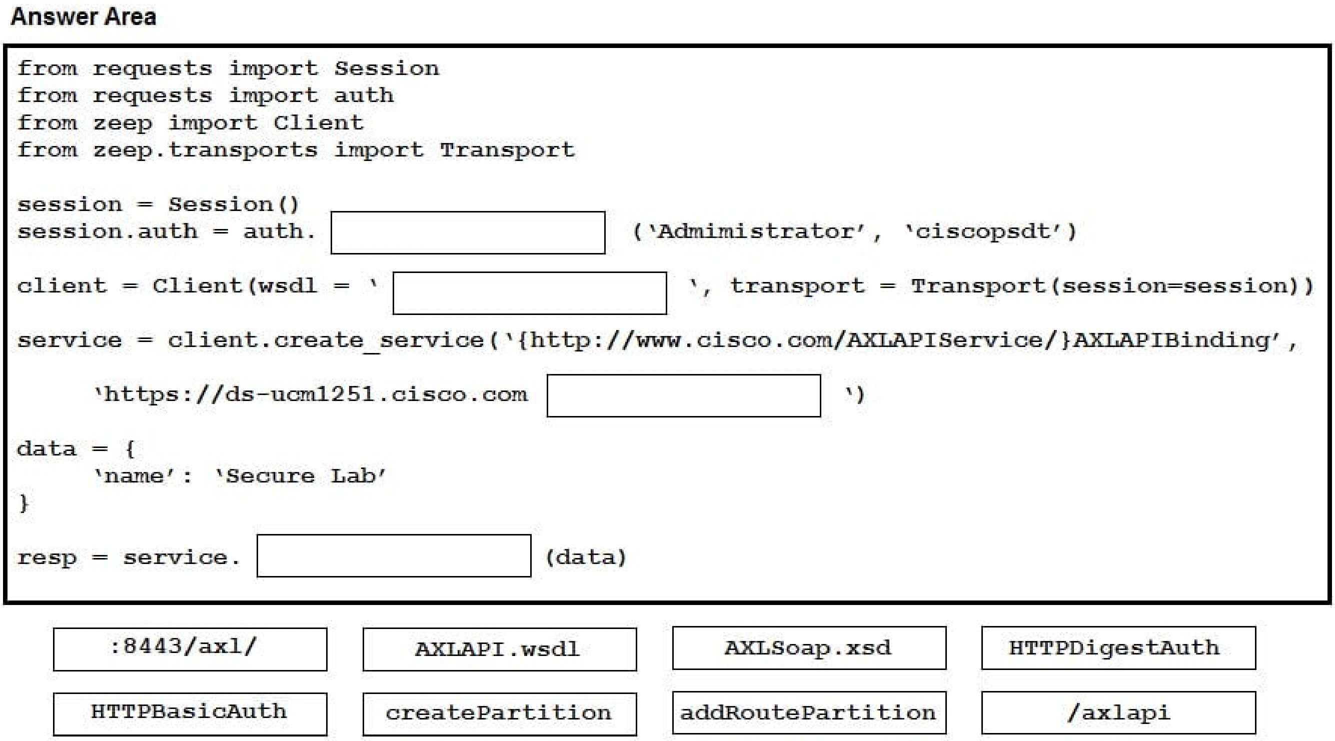
PersonEmail

QUESTION 5

DRAG DROP

Drag drop the code to create a valid AXL API script to create a new Route Partition, using the Python Zeep library. Not all options are used.

Select and Place:



Answer Area

```
from requests import Session
from requests import auth
from zeep import Client
from zeep.transports import Transport

session = Session()
session.auth = auth.                    ('Administrator', 'ciscopsdt')

client = Client(wsdl = '                    ', transport = Transport(session=session))

service = client.create_service('{http://www.cisco.com/AXLAPIService/}AXLAPIBinding',

     'https://ds-ucm1251.cisco.com                    ')

data = {
     'name': 'Secure Lab'
}

resp = service.                    (data)
```

:8443/axl/

AXLAPI.wsdl

AXLSoap.xsd

HTTPDigestAuth

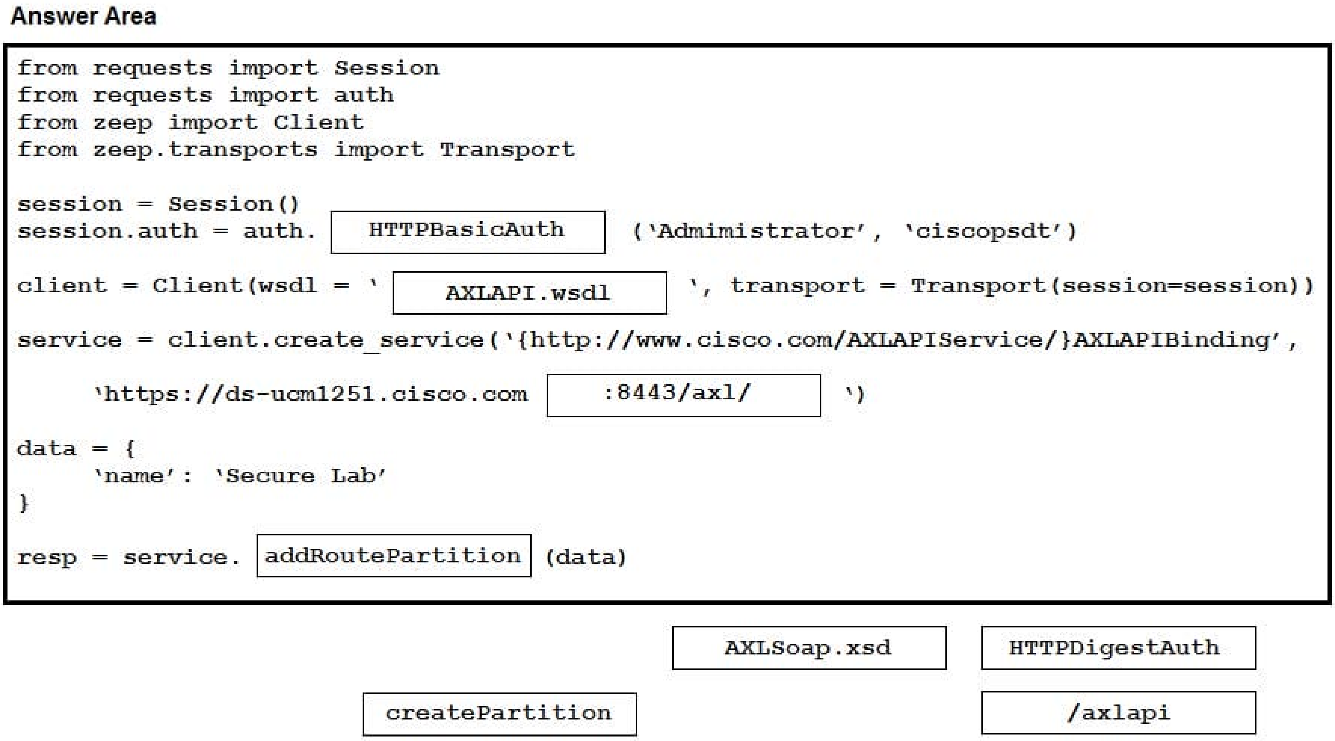
HTTPBasicAuth

createPartition

addRoutePartition

/axlapi

Correct Answer:



Answer Area

```
from requests import Session
from requests import auth
from zeep import Client
from zeep.transports import Transport

session = Session()
session.auth = auth. HTTPBasicAuth ('Administrator', 'ciscopsdt')

client = Client(wsdl = ' AXLAPI.wsdl ', transport = Transport(session=session))

service = client.create_service('{http://www.cisco.com/AXLAPIService/}AXLAPIBinding',

     'https://ds-ucm1251.cisco.com :8443/axl/ ')

data = {
     'name': 'Secure Lab'
}

resp = service. addRoutePartition (data)
```

AXLSoap.xsd

HTTPDigestAuth

createPartition

/axlapi

QUESTION 6

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.

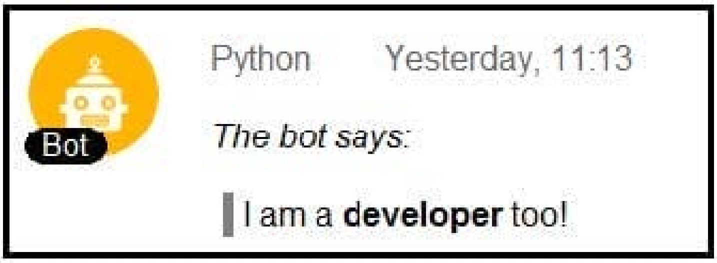
C. A macro can store data using a local JavaScript datastore.

D. Macros execute in the order defined in the Macro Editor.

Correct Answer: C

QUESTION 7

DRAG DROP



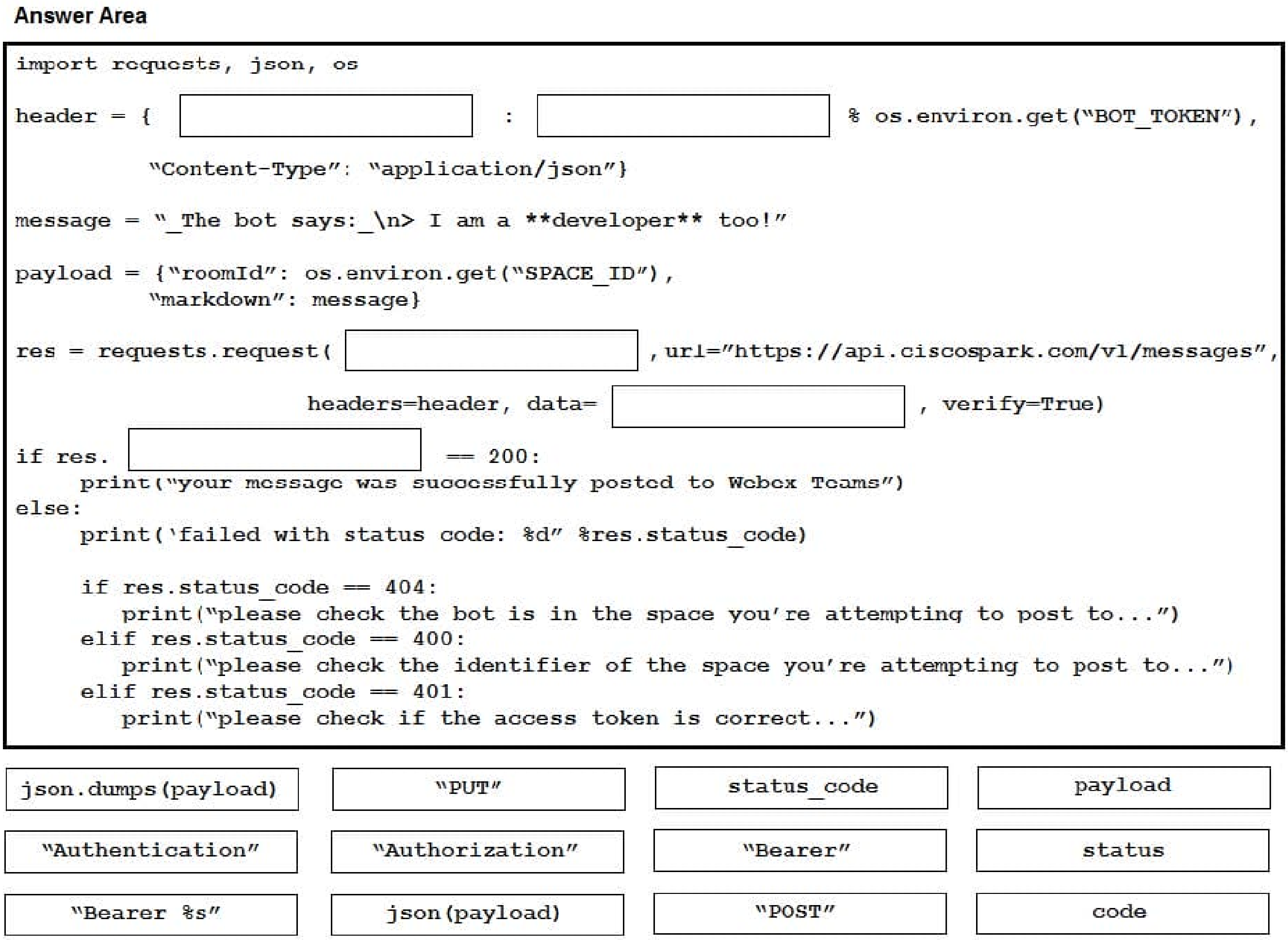
Python Yesterday, 11:13

The bot says:

> I am a **developer** too!

Refer to the exhibit. A developer has implemented ChatOps to a Webex Teams space as described in the exhibit. The Python script that pushes the notifications to the Teams space is shown. Drag and drop the code to complete the script. Not all options are used.

Select and Place:

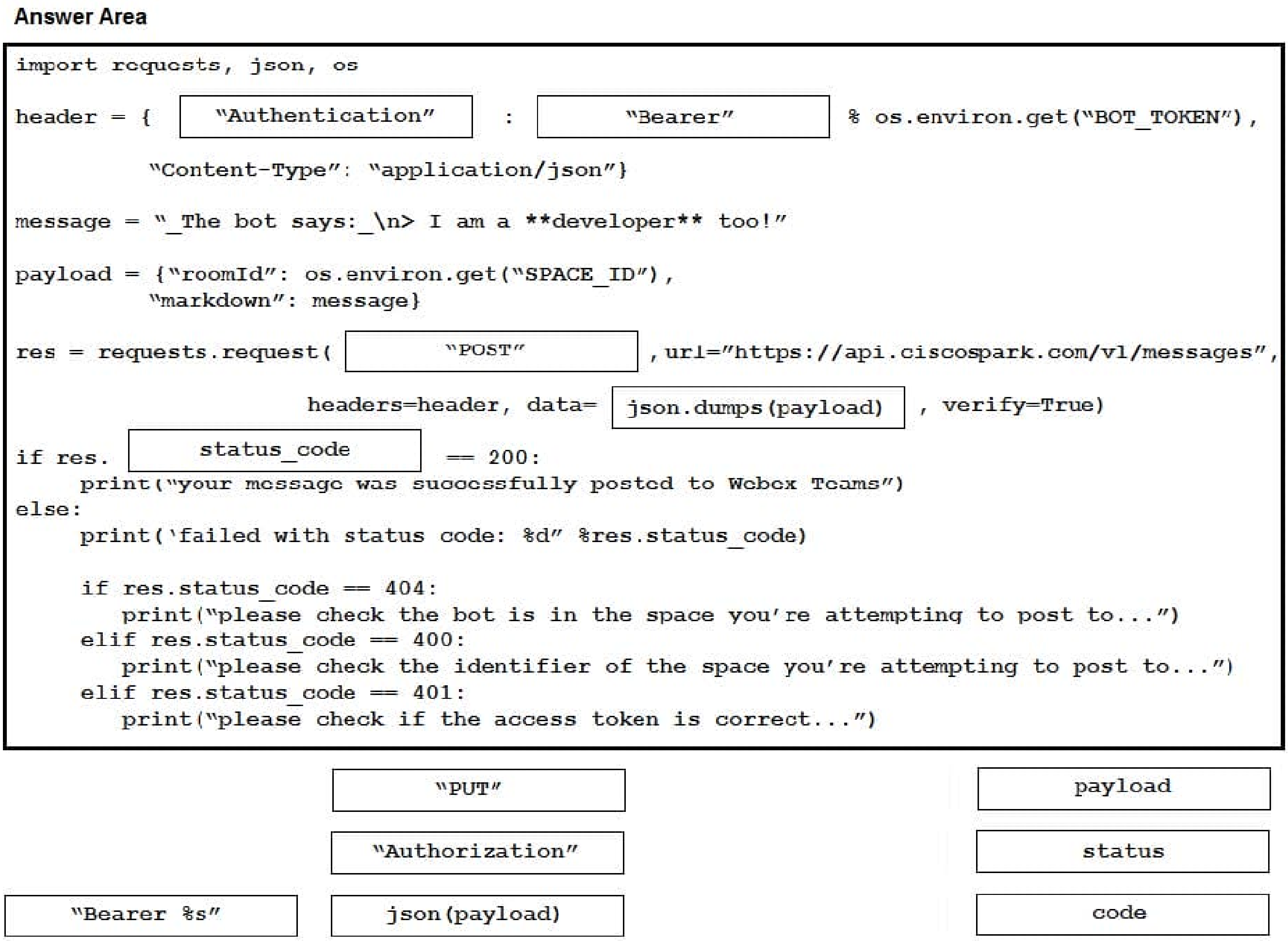


Answer Area

```
import requests, json, os

header = { [____________] : [____________] % os.environ.get(“BOT_TOKEN”),

         “Content-Type”: “application/json”}

message = “_The bot says:_\n> I am a **developer** too!”

payload = {“roomId”: os.environ.get(“SPACE_ID”),
         “markdown”: message}

res = requests.request( [____________] ,url=”https://api.ciscospark.com/v1/messages”,

                    headers=header, data= [____________] , verify=True)

if res. [____________]  == 200:
     print(“your message was successfully posted to Webex Teams”)
else:
     print(‘failed with status code: %d” %res.status_code)

     if res.status_code == 404:
        print(“please check the bot is in the space you’re attempting to post to...”)
     elif res.status_code == 400:
        print(“please check the identifier of the space you’re attempting to post to...”)
     elif res.status_code == 401:
        print(“please check if the access token is correct...”)
```

json.dumps(payload)	“PUT”	status_code	payload
“Authentication”	“Authorization”	“Bearer”	status
“Bearer %s”	json(payload)	“POST”	code

Correct Answer:



Answer Area

```
import requests, json, os

header = { "Authentication" : "Bearer" % os.environ.get("BOT_TOKEN"),
          "Content-Type": "application/json"}

message = "_The bot says:_\n> I am a **developer** too!"

payload = {"roomId": os.environ.get("SPACE_ID"),
          "markdown": message}

res = requests.request( "POST" ,url="https://api.ciscospark.com/v1/messages",
                     headers=header, data= json.dumps(payload) , verify=True)
if res. status_code  == 200:
     print("your message was successfully posted to Webex Teams")
else:
     print('failed with status code: %d" %res.status_code)

     if res.status_code == 404:
        print("please check the bot is in the space you're attempting to post to...")
     elif res.status_code == 400:
        print("please check the identifier of the space you're attempting to post to...")
     elif res.status_code == 401:
        print("please check if the access token is correct...")
```

"PUT"

payload

"Authorization"

status

"Bearer %s"

json(payload)

code

QUESTION 8

Which two characteristics of REST APIs? (Choose two.)

A. Resources are accessed using Uniform Resource Identifiers.

B. API operations for Create/Read/Update/Delete are mapped to standard HTTP methods.

C. Cookies are used for the duration of the session.

D. REST API extends Remote Procedure Call.

E. The server manages the session state.

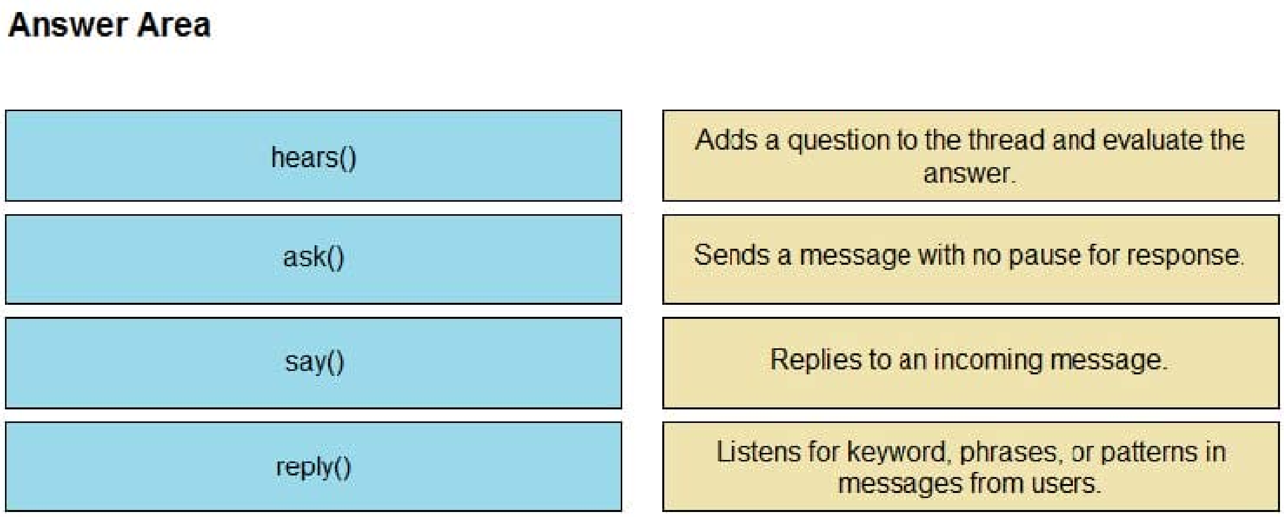
Correct Answer: AB

QUESTION 9

DRAG DROP

Drag and drop the bootkit functions from the left onto the correct actions on the right.

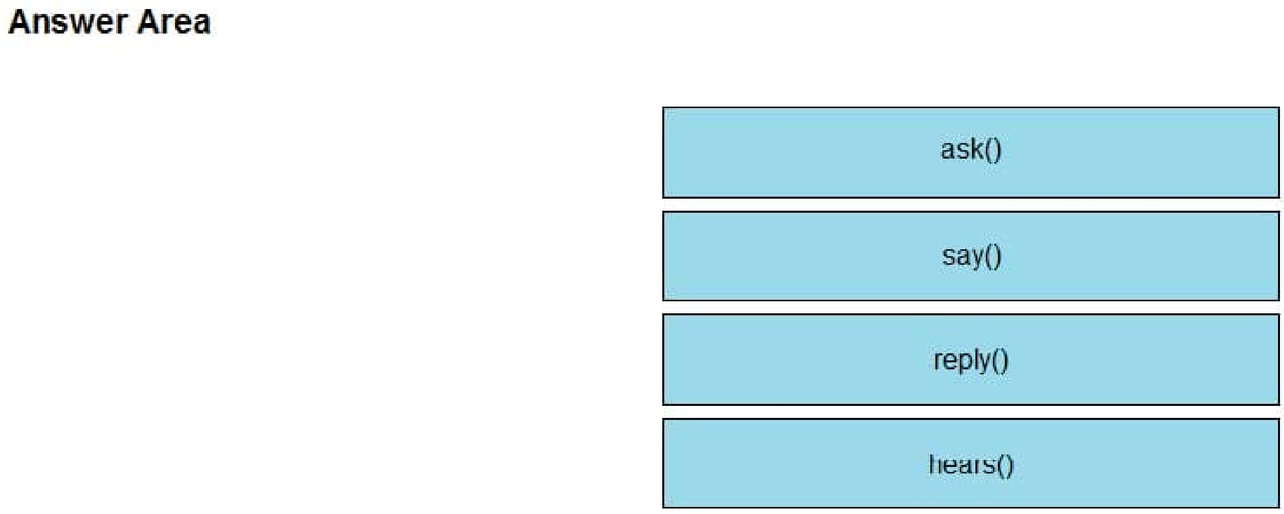
Select and Place:



Answer Area

hears()	Adds a question to the thread and evaluate the answer.
ask()	Sends a message with no pause for response.
say()	Replies to an incoming message.
reply()	Listens for keyword, phrases, or patterns in messages from users.

Correct Answer:



Answer Area

ask()

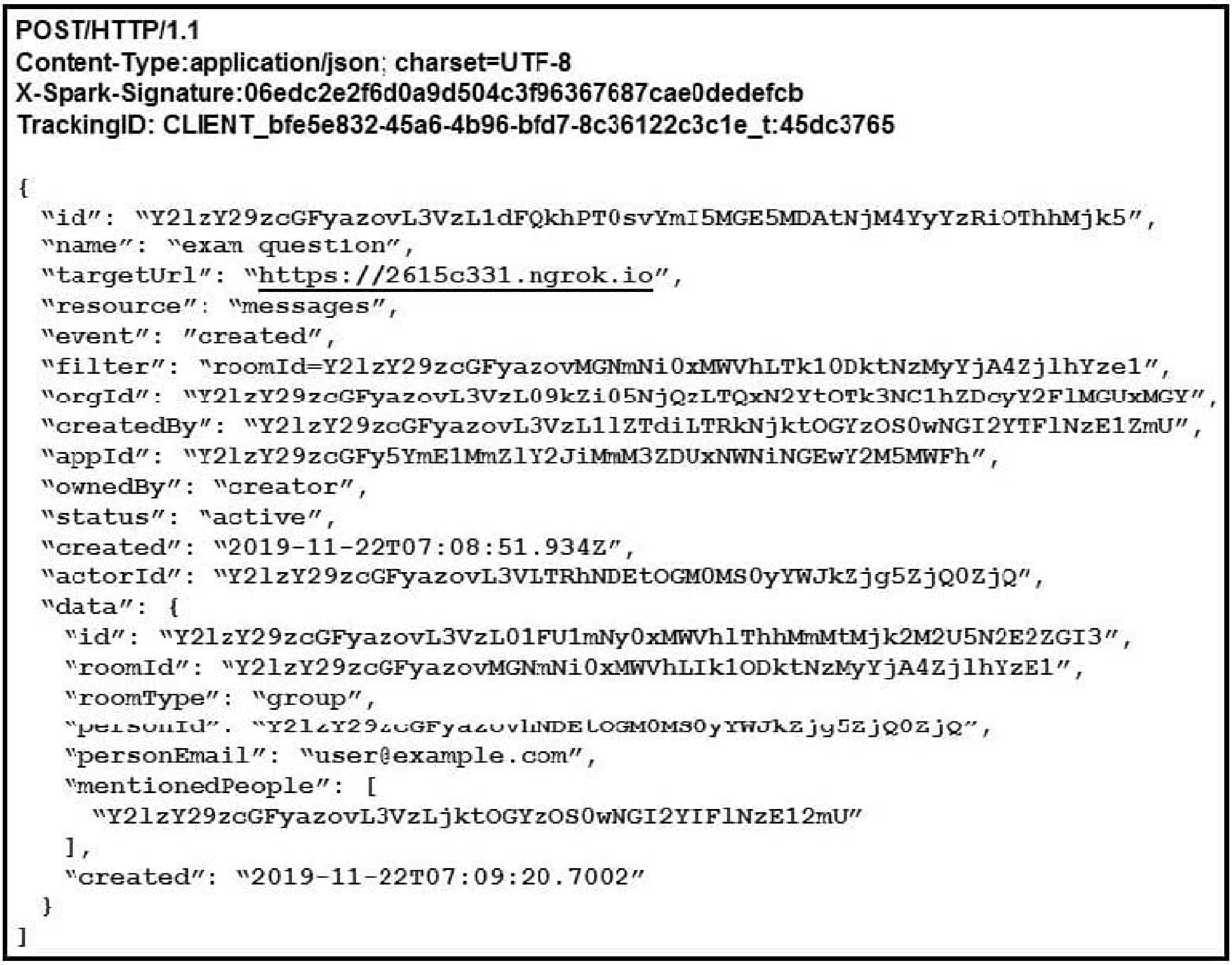
say()

reply()

hears()

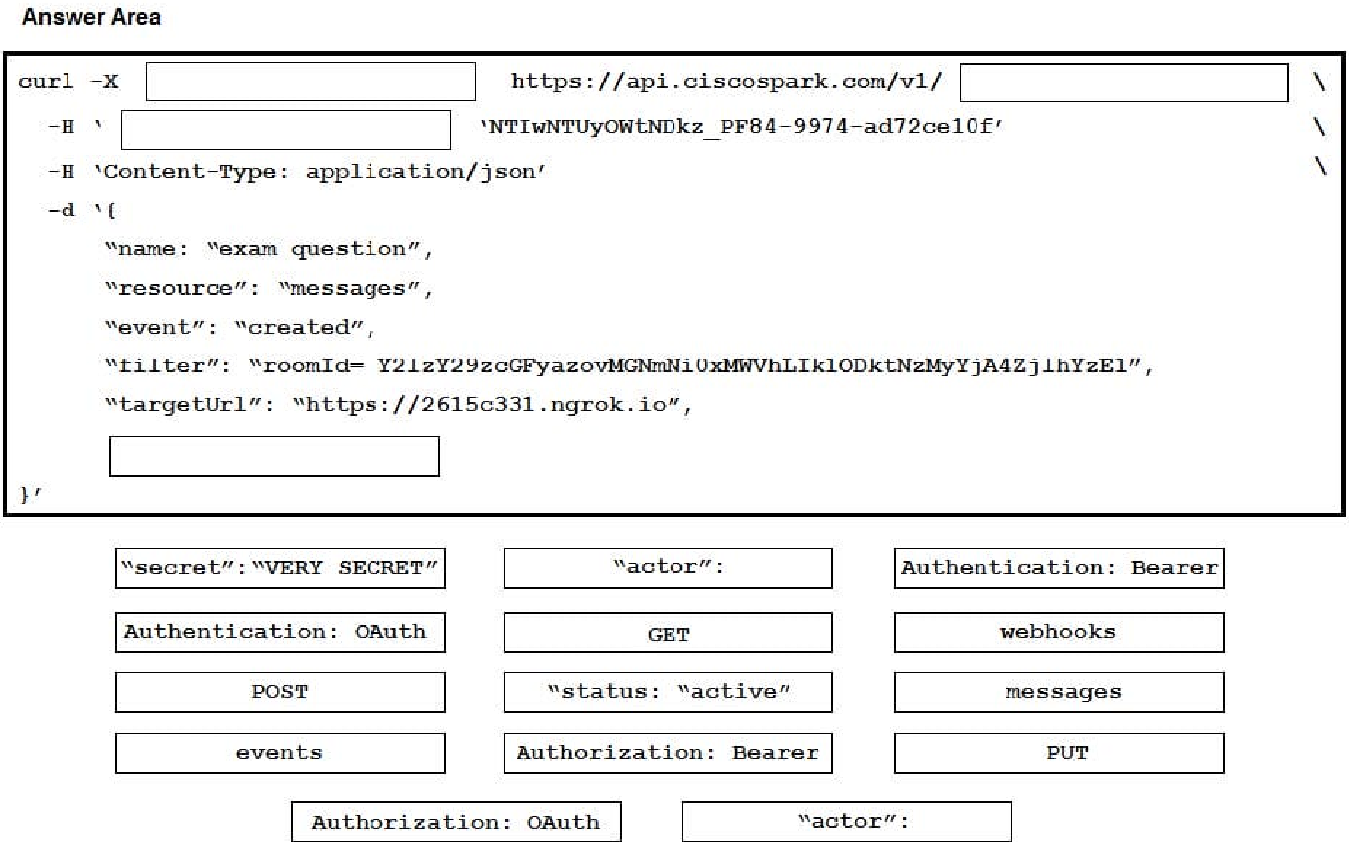
QUESTION 10

DRAG DROP Refer to the exhibit. A bot is receiving notifications such as the one displayed in the exhibit. Drag and drop the code onto the snippet to complete the API request that was sent to Webex Teams so that the bot will receive these notifications. Not all options are used.



```
POST/HTTP/1.1
Content-Type:application/json; charset=UTF-8
X-Spark-Signature:06edc2e2f6d0a9d504c3f96367687cae0dedefcb
TrackingID: CLIENT_bfe5e832-45a6-4b96-bfd7-8c36122c3c1e_t:45dc3765

{
  “id”: “Y2lzY29zcGFyazovL3VzL1dFQkhPT0svYmI5MGE5MDAtNjM4YyYzRiOThhMjk5”,
  “name”: “exam question”,
  “targetUrl”: “https://2615c331.ngrok.io”,
  “resource”: “messages”,
  “event”: ”created”,
  “filter”: “roomId=Y2lzY29zcGFyazovMGNmNi0xMWVhLTk10DktNzMyYjA4ZjlhYze1”,
  “orgId”: “Y2lzY29zcGFyazovL3VzL09kZi05NjQzLTQxN2YtOTk3NC1hZDcyY2FlMGUxMGY”,
  “createdBy”: “Y2lzY29zcGFyazovL3VzL1lZTdiLTRkNjktOGYzOS0wNGI2YTFlNzE1ZmU”,
  “appId”: “Y2lzY29zcGFy5YmE1MmZlY2JiMmM3ZDUxNWNiNGEwY2M5MWFh”,
  “ownedBy”: “creator”,
  “status”: “active”,
  “created”: “2019-11-22T07:08:51.934Z”,
  “actorId”: “Y2lzY29zcGFyazovL3VLTRhNDEtOGM0MS0yYWJkZjg5ZjQ0ZjQ”,
  “data”: {
    “id”: “Y2lzY29zcGFyazovL3VzL01FU1mNy0xMWVhlThhMmMtMjk2M2U5N2E2ZGI3”,
    “roomId”: “Y2lzY29zcGFyazovMGNmNi0xMWVhLIk1ODktNzMyYjA4ZjlhYzE1”,
    “roomType”: “group”,
    “personId”: “Y2lzY29zcGFyazovlNDEtOGM0MS0yYWJkZjg5ZjQ0ZjQ”,
    “personEmail”: “user@example.com”,
    “mentionedPeople”: [
      “Y2lzY29zcGFyazovL3VzLjktOGYzOS0wNGI2YIFlNzE12mU”
    ],
    “created”: “2019-11-22T07:09:20.7002”
  }
]
```

Select and Place:

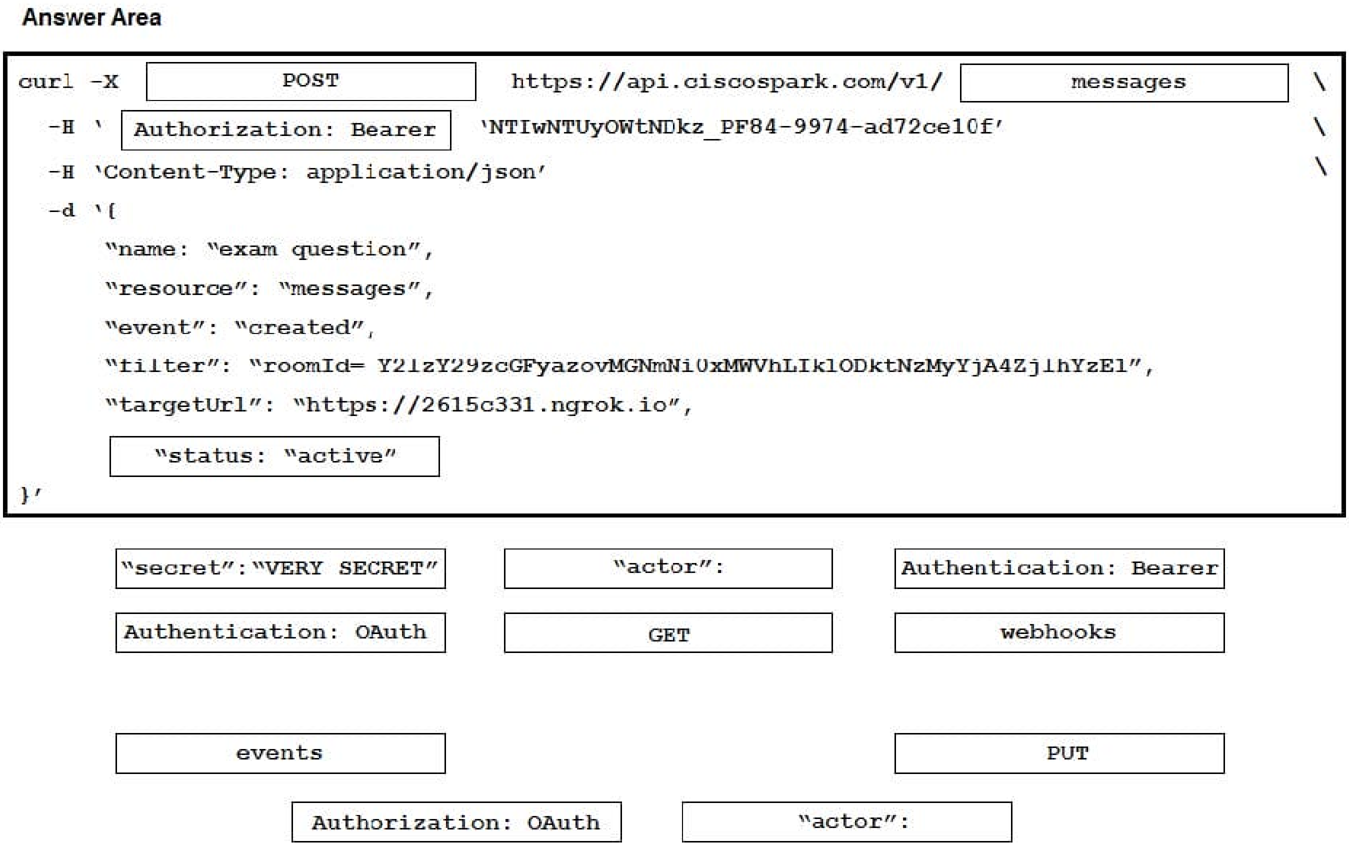


Answer Area

```
curl -X                      https://api.ciscospark.com/v1/                      \
  -H '                       'NTIwNTUyOWtNDkz_PF84-9974-ad72ce10f'               \
  -H 'Content-Type: application/json'                                            \
  -d '{
      "name: "exam question",
      "resource": "messages",
      "event": "created",
      "filter": "roomId= Y21zY29zcGFyazovMGNmNi0xMWVhLIkl0DktNzMyYjA4Zj1hYzE1",
      "targetUrl": "https://2615c331.ngrok.io",

}'
```

```
"secret":"VERY SECRET"      "actor":                  Authentication: Bearer
Authentication: OAuth       GET                       webhooks
POST                        "status: "active"         messages
events                      Authorization: Bearer     PUT
          Authorization: OAuth        "actor":
```

Correct Answer:



Answer Area

```
curl -X [ POST ] https://api.ciscospark.com/v1/ [ messages ] \
  -H ' [ Authorization: Bearer ] 'NTIwNTUyOWtNDkz_PF84-9974-ad72ce10f' \
  -H 'Content-Type: application/json' \
  -d '{
      "name: "exam question",
      "resource": "messages",
      "event": "created",
      "filter": "roomId= Y2lzY29zcGFyazovMGNmNi0xMWVhLIklODktNzMyYjA4ZjlhYzE1",
      "targetUrl": "https://2615c331.ngrok.io",
     [ "status: "active" ]
}'
```

"secret":"VERY SECRET"

"actor":

Authentication: Bearer

Authentication: OAuth

GET

webhooks

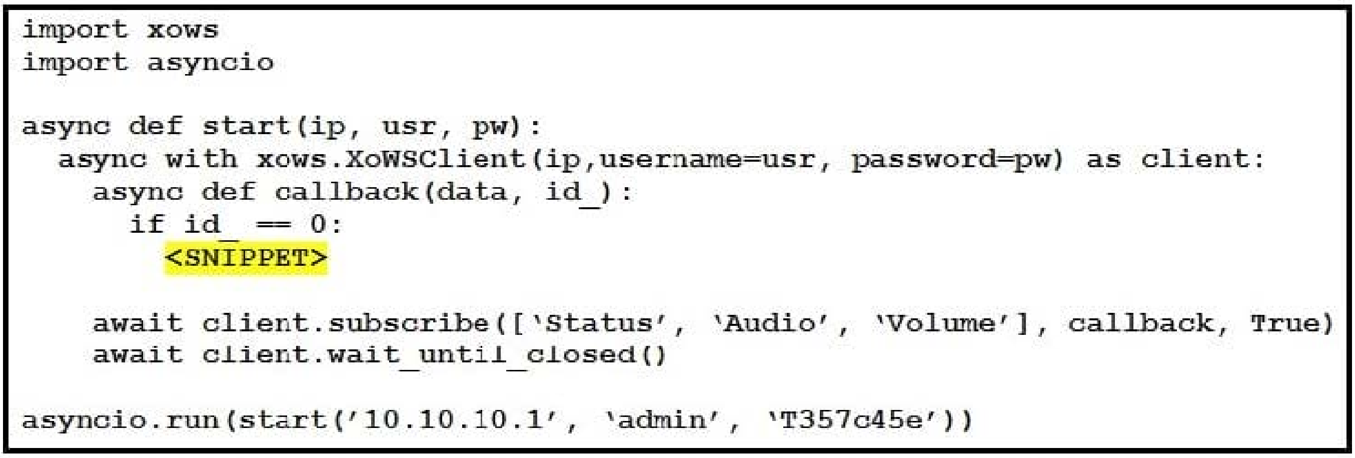
events

PUT

Authorization: OAuth

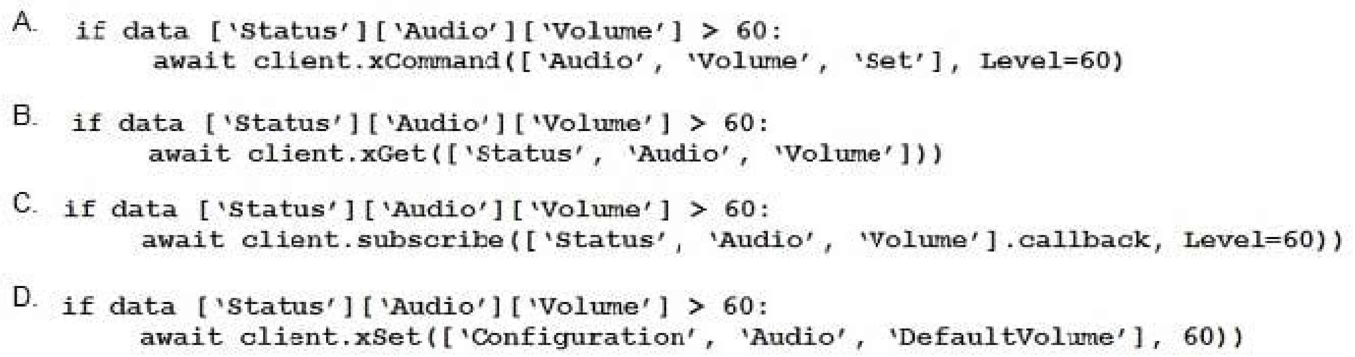
"actor":

QUESTION 11



```
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'))
```

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?



```
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))
```

A. Option A

B. Option B

C. Option C

D. Option D

Correct Answer: A

QUESTION 12

Which Cisco Meeting Server REST API object resource is used to retrieve active call information?

A. /activecalls

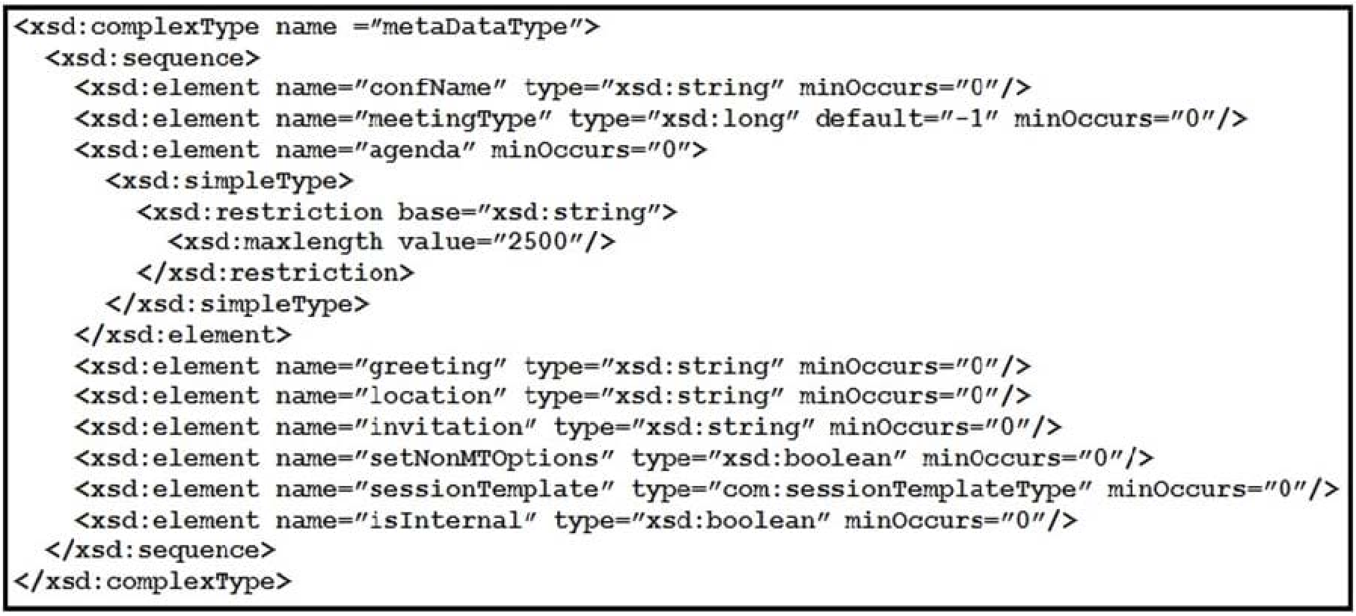
B. /calls

C. /getactivecalls

D. /callProfiles

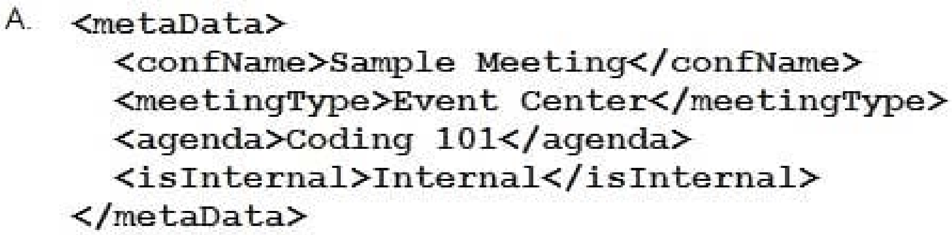
Correct Answer: B

QUESTION 13



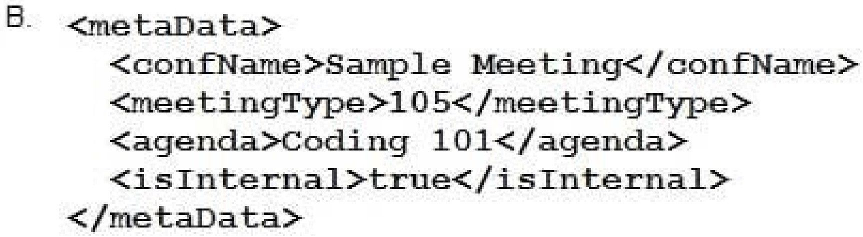
```
<xsd:complexType name ="metaDataType">
  <xsd:sequence>
    <xsd:element name="confName" type="xsd:string" minOccurs="0"/>
    <xsd:element name="meetingType" type="xsd:long" default="-1" minOccurs="0"/>
    <xsd:element name="agenda" minOccurs="0">
      <xsd:simpleType>
        <xsd:restriction base="xsd:string">
          <xsd:maxlength value="2500"/>
        </xsd:restriction>
      </xsd:simpleType>
    </xsd:element>
    <xsd:element name="greeting" type="xsd:string" minOccurs="0"/>
    <xsd:element name="location" type="xsd:string" minOccurs="0"/>
    <xsd:element name="invitation" type="xsd:string" minOccurs="0"/>
    <xsd:element name="setNonMTOptions" type="xsd:boolean" minOccurs="0"/>
    <xsd:element name="sessionTemplate" type="com:sessionTemplateType" minOccurs="0"/>
    <xsd:element name="isInternal" type="xsd:boolean" minOccurs="0"/>
  </xsd:sequence>
</xsd:complexType>
```

Refer to the exhibit. Based on this Webex Meetings XML API schema snippet, which XML object is a valid element?



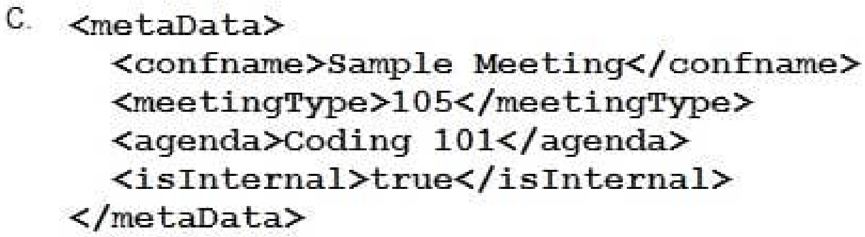
A.

```
<metaData>
   <confName>Sample Meeting</confName>
   <meetingType>Event Center</meetingType>
   <agenda>Coding 101</agenda>
   <isInternal>Internal</isInternal>
</metaData>
```



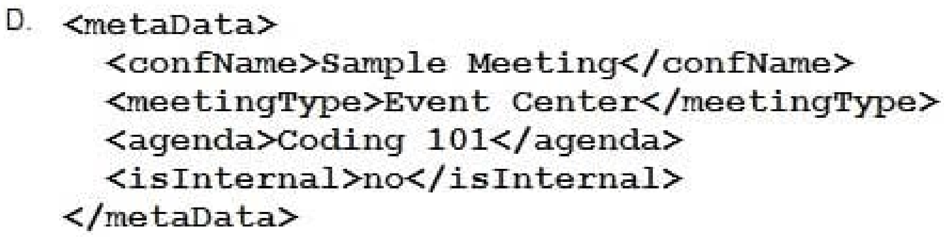
B.

```
<metaData>
   <confName>Sample Meeting</confName>
   <meetingType>105</meetingType>
   <agenda>Coding 101</agenda>
   <isInternal>true</isInternal>
</metaData>
```



C.

```
<metaData>
   <confname>Sample Meeting</confname>
   <meetingType>105</meetingType>
   <agenda>Coding 101</agenda>
   <isInternal>true</isInternal>
</metaData>
```



D.

```
<metaData>
   <confName>Sample Meeting</confName>
   <meetingType>Event Center</meetingType>
   <agenda>Coding 101</agenda>
   <isInternal>no</isInternal>
</metaData>
```

A. Option A

B. Option B

C. Option C

D. Option D

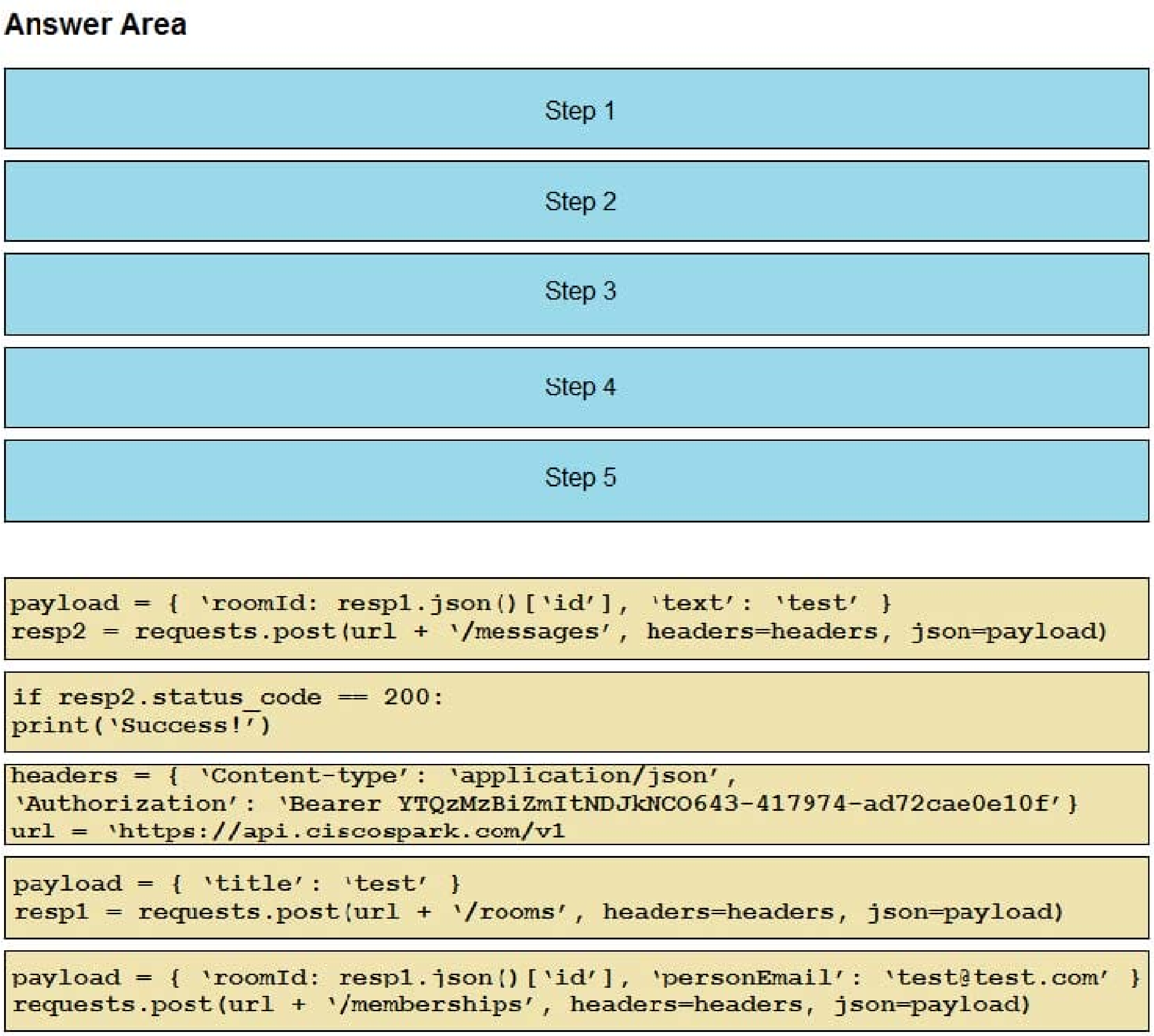
Correct Answer: C

QUESTION 14

DRAG DROP

Drag and drop the code snippets into the correct order to create a Python Webex Teams REST API script that performs this sequence of tasks: create a Space, add a user to the Space, post the word “test” in the Space. Assume that the Python “requests” module has already been imported.

Select and Place:



Answer Area

Step 1

Step 2

Step 3

Step 4

Step 5

```
payload = { 'roomId: resp1.json()['id'], 'text': 'test' }
resp2 = requests.post(url + '/messages', headers=headers, json=payload)
```

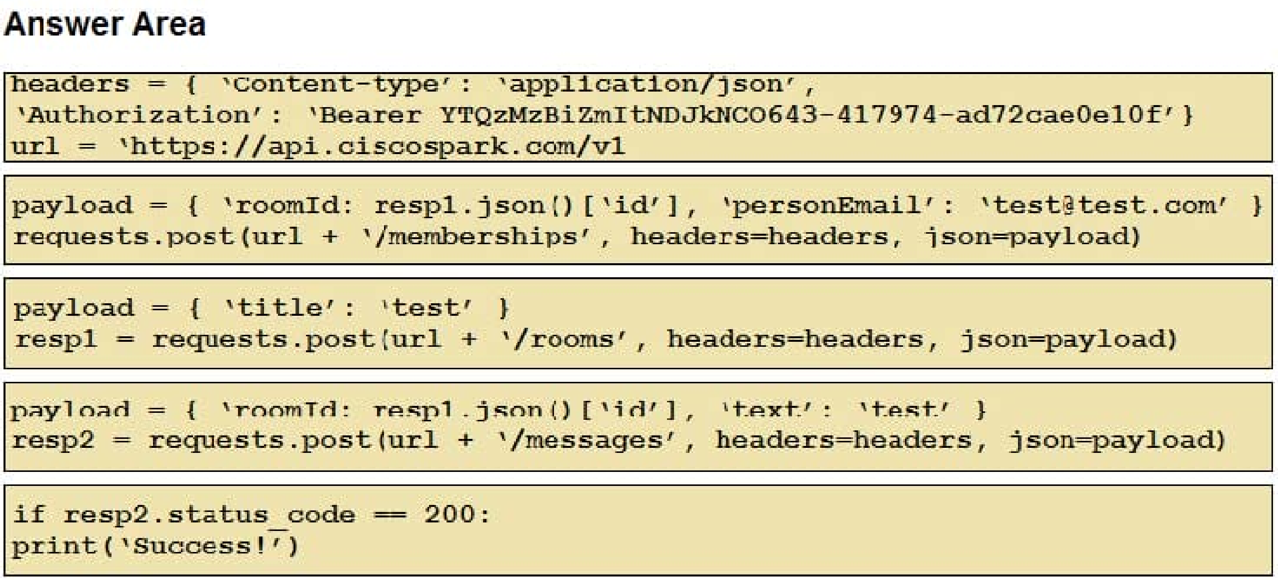
```
if resp2.status_code == 200:
print('Success!')
```

```
headers = { 'Content-type': 'application/json',
'Authorization': 'Bearer YTQzMzBiZmItNDJkNCO643-417974-ad72cae0e10f'}
url = 'https://api.ciscospark.com/v1
```

```
payload = { 'title': 'test' }
resp1 = requests.post(url + '/rooms', headers=headers, json=payload)
```

```
payload = { 'roomId: resp1.json()['id'], 'personEmail': 'test@test.com' }
requests.post(url + '/memberships', headers=headers, json=payload)
```

Correct Answer:



Answer Area

```
headers = { ‘Content-type’: ‘application/json’,
‘Authorization’: ‘Bearer YTQzMzBiZmItNDJkNCO643-417974-ad72cae0e10f’}
url = ‘https://api.ciscospark.com/v1
```

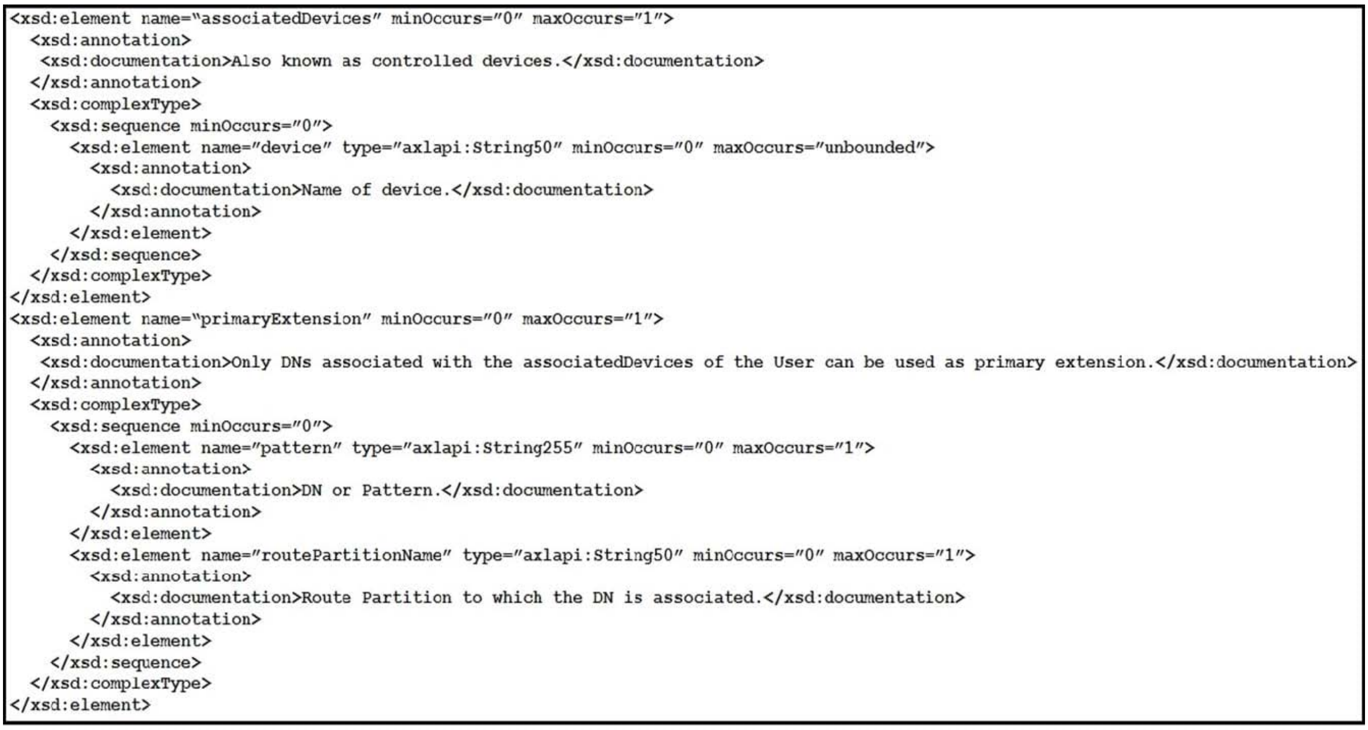
```
payload = { ‘roomId: resp1.json()[‘id’], ‘personEmail’: ‘test@test.com’ }
requests.post(url + ‘/memberships’, headers=headers, json=payload)
```

```
payload = { ‘title’: ‘test’ }
resp1 = requests.post(url + ‘/rooms’, headers=headers, json=payload)
```

```
payload = { ‘roomId: resp1.json()[‘id’], ‘text’: ‘test’ }
resp2 = requests.post(url + ‘/messages’, headers=headers, json=payload)
```

```
if resp2.status_code == 200:
print(‘Success!’)
```

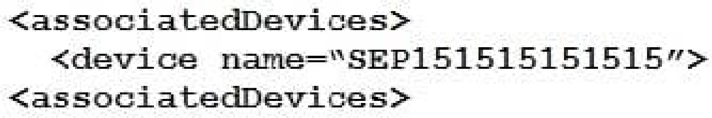
QUESTION 15



```
<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>
```

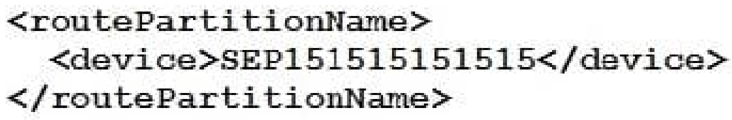
Refer to the exhibit. The exhibit shows the XML schema to associate a device for the AddUser operation in AXL. Which XML snippet is correct to associate a device with name "SEP151515151515"?

A.



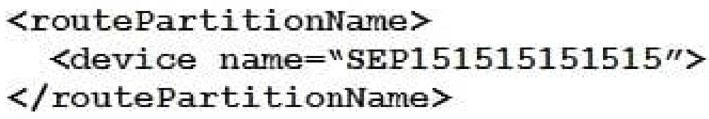
```
<associatedDevices>
   <device name="SEP151515151515">
<associatedDevices>
```

B.



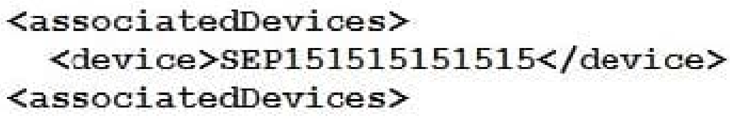
```
<routePartitionName>
   <device>SEP151515151515</device>
</routePartitionName>
```

C.



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

D.



```
<associatedDevices>
   <device>SEP151515151515</device>
<associatedDevices>
```

A. Option A

B. Option B

C. Option C

D. Option D

Correct Answer: D