



Visit and Download Full Version Certificationtime Exam Dumps
<https://certificationtime.com/updated/300-920-exam-dumps-pdf/>



Cisco

Exam Questions 300-920

Developing Applications for Cisco Webex and Webex Devices (DEVWBX)

<https://certificationtime.com/>





QUESTION 1

```

: <https://api.ciscospark.com/v1/people?
max=1&cursor=bGltaxQ9MSZluZGV4PTI=>; rel="next"
Content-Type: application/json; charset=UTF-8
{
  "notFoundIds": null,
  "items": [
    {
      "id": "Y2lzY29zcGFyazovL3VzL1BFTlylmZjViLTQ4OdlTYjUyOS11OWQ3NTRjYWJiMzI",
      "emails": ["johnDoe@example.com"],
      "displayName": "John Doe",
      "firstName": "John",
      "lastName": "Doe",
      ...
    }
  ]
}

```

Refer to the exhibit. A Webex Teams REST API response is shown with the HTTP Header missing. Which HTTP header expected in this response?

- A. Push
- B. Link
- C. Patch
- D. Put

Answer: B

QUESTION 2

A Webex Teams bot is deployed but soon it stops responding. Which two explanations are the cause of the issue? (Choose two.)

- A. A new webhook was created, which marks the old webhook as inactive.
- B. The web server that is set to receive webhooks is not configured to return a 200 message.
- C. And the webhook is disabled.
- D. The webhook secret is expired and must be refreshed.
- E. The refresh token is not being used.
- F. The bot owner regenerated the access token on developer.webex.com.

Answer: AB

QUESTION 3

```

<xsd:complexType name="lstRecordingResponse">
  <xsd:complexContent>
    <xsd:extension base="serv:bodyContentType">
      <xsd:sequence>
        <xsd:element name="matchingRecords" type="
serv:matchingRecordsType" minOccurs="0"/>
        <xsd:element name="recording" type="ep:recordingType" minOccurs="
0" maxOccurs="unbounded"/>
      ...
    <xsd:complexType name="recordingType">
      <xsd:sequence>
        <xsd:element name="recordingID" type="xsd:int"/>
        <xsd:element name="hostWebExID" type="xsd:string"/>
        <xsd:element name="name" type="xsd:string"/>
        <xsd:element name="createTime" type="xsd:string"/>
        <xsd:element name="streamURL" type="xsd:string"/>
        <xsd:element name="fileURL" type="xsd:string"/>
        <xsd:element name="password" type="xsd:string" minOccurs="0" />
      ...
    
```

Refer to the exhibit. A snippet from the XSD schema of the Webex Meeting XML API 'LstRecordingResponse' element is listed in the exhibit. Assuming that a variable named 'resp' exists that contains the XML response from a successful 'LstRecording' request, which code snippet correctly generates a simple report that lists meeting names and recording file download links?

- A.

```
list = (new window.DOMParser()).parseFromString(resp, 'application/json')
for (const item of lst.getElementsByTagNameNS('*', 'matchingRecords')){
document.writeln(
  <p> Meeting Name:${item.getElementsByTagNameNS('*', 'name') [0].textContent}<br>');
document.write(
  'Download:${item.getElementsByTagNameNS('*', 'fileURL') [0].textContent}<br>');
```
- B.

```
list = (new window.DOMParser()).parseFromString(resp, 'LstRecordingResponse')
for (const item of lst.getElementsByTagNameNS('*', 'matchingRecords')){
document.writeln(
  <p> Meeting Name:${item.getElementsByTagNameNS('*', 'name') [0].textContent}<br>');
document.write(
  'Download:${item.getElementsByTagNameNS('*', 'fileURL') [0].textContent}<br>');
```



```
list = (new window.DOMParser()).parseFromString(resp, 'text/xml')
for (const item of lst.getElementsByTagNameNS('*', 'matchingRecords')){
document.writeln(
  <p> Meeting Name:${item.getElementsByTagNameNS('*', 'hostWebExID') [0] }<br>');
document.write(
  'Download:${item.getElementsByTagNameNS('*', 'streamURL') [0].textContent}<br>');
```

• C.

```
list = (new window.DOMParser()).parseFromString(resp, 'text/xml')
for (const item of lst.getElementsByTagNameNS('*', 'recording')){
document.writeln(
  <p> Meeting Name:${item.getElementsByTagNameNS('*', 'name') [0].textContent}<br>');
document.write(
  'Download:${item.getElementsByTagNameNS('*', 'fileURL') [0].textContent}<br>');
```

• D.

Answer: A

QUESTION 4

```
1  /**
2  * A small In-Room controls panel with 4 quick dial numbers
3  */
4  const xapi = require('xapi');
5
6  //These match the widget ids of the In-Room control buttons
7  const numbers = {
8    dial_catering: 'catering@example.com',
9    dial_it_support: 'itsupport@example.com',
10   dial_taxi: 'taxi@example.com',
11   dial_reception: 'reception@example.com',
12  },
13
14  function dial(number) {
15    xapi.command('call', {Number:number});
16  }
17
18  function listenToGui() {
19    xapi.event.on('UserInterface Extensions Widget Action', (event)-> {
20      if (event.Type === 'clicked') {
21        const number = numbers[event.WidgetId];
22        if(number) dial(number);
23        else console.log('Unknown button pressed', event.WidgetId);
24      }
25    });
26  }
```

Severity ▾ Enter filter Show history

```
22:38:35 [system] > Starting macros...
22:38:35 Quick dial > Loading...
22:38:36 Quick dial > Ready!
22:38:36 Quick dial > 'Unhandled promise rejection' (code: 3,
message: 'Ambiguous command',
data: {xpath: '/Call'})
```

Refer to the exhibit. An end user reports that the speed dial button is not working on their Webex Device, and when loading into the Macro Editor, this error was presented. On which line is the incorrect syntax?

- A. line 4
- B. line 14
- C. line 15
- D. line 22

Answer: C

QUESTION 5

Refer to the exhibit. What is the Webex Teams REST API HTTP response status code, based on this code snippet?

```
{
  "message": "The user has sent too many requests in a given amount of time. Please refer to the
Retry-After response header to wait before making a new request.",
  "errors": [
    {
      "description" : "The user has sent too many requests in a given amount of time. Please
refer to the Retry-After response header to wait before making a new request."
    }
  ],
  "trackingId": "ROUTER_5D224C5A-C32B-01BB-007E-2D30B4D4007E"
}
```

- A. 401
- B. 403
- C. 429
- D. 501

Answer: C

QUESTION 6



```
const xml = '<?xml version="1.0" encoding="UTF-8"?>
<serv:message xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
xmlns:serv="http://www.webex.com/schemas/2002/06/service"
xsi:schemaLocation="http://www.webex.com/schemas/2002/06/service"
http://www.webex.com/schemas/2002/06/service/service.xsd">
  <header>
    <securityContext>
      <webExID>admin@cisco.com</webExID>
      <password>password</password>
      <siteName>cisco</siteName>
      <returnAdditionalInfo>true</returnAdditionalInfo>
    </securityContext>
  </header>
  <body>
    <bodyContent xsi:type="java:com.webex.service.binding.user.SetUser">
      <webExId>user@cisco.com</webExId>
      <personalMeetingRoom>
        <hostPIN>3421</hostPIN>
      </personalMeetingRoom>
    </bodyContent>
  </body>
</serv:message>
var xmlhttp = new XMLHttpRequest();

<< missing code >>

xmlhttp.setRequestHeader('Content-Type', 'text/xml');
xmlhttp.send(xml);
```

Refer to the exhibit. A developer must construct an HTTP Request to use the XML API to set a Personal Meeting Room PIN for a given user. Which code completes the code to create the request?

- A. xmlhttp.open("GET", "https://cisco.webex.com/WBXService/XMLService");
- B. xmlhttp.open("PATCH", "https://cisco.webex.com/WBXService/XMLService");
- C. xmlhttp.open("PUT", "https://cisco.webex.com/WBXService/XMLService");
- D. xmlhttp.open("POST", "https://cisco.webex.com/WBXService/XMLService");

Answer: D

Full Access

<https://certificationtime.com/updated/300-920-exam-dumps-pdf/>