

300-920^{Q&As}

Developing Applications for Cisco Webex and Webex Devices
(DEVWBX)

Pass Cisco 300-920 Exam with 100% Guarantee

Free Download Real Questions & Answers **PDF** and **VCE** file from:

<https://www.leads4pass.com/300-920.html>

100% Passing Guarantee
100% Money Back Assurance

Following Questions and Answers are all new published by Cisco
Official Exam Center

- ⚙️ **Instant Download** After Purchase
- ⚙️ **100% Money Back** Guarantee
- ⚙️ **365 Days** Free Update
- ⚙️ **800,000+** Satisfied Customers



QUESTION 1

DRAG DROP

Drag and drop the code onto the snippet to construct the JavaScript to create a new meeting with the Webex Meetings XML API. Options can be used more than once.

Select and Place:

```
var http = require('https');
var xml = '<?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>mySite</siteName>
      <webExID>Ciscouser</webExID>
      < <div data-bbox="238 374 441 398" data-label="Text">

<div data-bbox="125 418 328 442" data-label="Text">

</div data-bbox="153 445 434 459" data-label="Text">

</div data-bbox="153 459 220 473" data-label="Text">

<div data-bbox="105 474 917 574" data-label="Text">

</div data-bbox="105 574 814 619" data-label="Text">

<div data-bbox="108 621 264 650" data-label="Text">

</div data-bbox="111 675 311 696" data-label="Text">

<div data-bbox="328 675 528 696" data-label="Text">

<div data-bbox="111 707 311 729" data-label="Text">

<div data-bbox="328 707 528 729" data-label="Text">

</div data-bbox="68 751 189 765" data-label="Text">

Correct Answer:


```

```
var http = require('https');
var xml = '<?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>mySite</siteName>
      <webExID>Ciscouser</webExID>
      < { 'method' : 'POST' } >
PD4WiqNfRkxBR19B...RERJTkdfU0hBMjU2X0FMR09SSVRITV8=
</ accessToken >
    </securityContext></header>
    <body>
    <bodyContent xsi:type="java.com.webex.service.binding.meeting.CreateMeeting">
      <metaData><confName>Sample Meeting</confName>
      <meetingType>105</meetingType></metaData>
      <schedule><startDate>12/13/2019 11:59:5</startDate></schedule>
    </bodyContent>
    </body>
  </serv:message>
var req = http.request('https://api.webex.com/WBXService/XMLService',
  sessionTicket );
req.write(xml);
req.end();
```

```

  { 'method' : 'PUT' }

```

QUESTION 2

```
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

Correct Answer: C

Reference: <https://community.cisco.com/t5/telepresence-and-video/ce9-2-1-macro-framework-discussions/td-p/3220093>

QUESTION 3

```
xcommand UserInterface Extensions List
OK
*r ExtensionsListResult (status=OK):
*r ExtensionsListResult Extensions Version: "1.5"
*r ExtensionsListResult Extensions Panel 1 Icon: Lightbulb
*r ExtensionsListResult Extensions Panel 1 Type: Statusbar
*r ExtensionsListResult Extensions Panel 1 Name: Panel
*r ExtensionsListResult Extensions Panel 1 PanelId: "panel_1"
*r ExtensionsListResult Extensions Panel 1 Order: 1
*r ExtensionsListResult Extensions Panel 1 Page 1 Name: "Page"
*r ExtensionsListResult Extensions Panel 1 Page 1 Row 1 Name: "Row"
*r ExtensionsListResult Extensions Panel 1 Page 1 Row 1 Widget 1 WidgetId: "widget_1"
*r ExtensionsListResult Extensions Panel 1 Page 1 Row 1 Widget 1 Type: Slider
*r ExtensionsListResult Extensions Panel 1 Page 1 Row 1 Widget 1 Options: "size=3"
** end
--
```

Refer to the exhibit. With this in-room configuration, which command successfully sets the slider to 50?

- A. xcommand UserInterface Extensions Widget SetSlider Value: 50
- B. xcommand UserInterface Extensions Widget Slider: 50 Widget: "widget_1"
- C. xconfiguration UserInterface Extensions widget_1 50
- D. xcommand UserInterface Extensions Widget SetValue WidgetId: "widget_1" Value: 50

Correct Answer: A

Reference: <https://www.cisco.com/c/dam/en/us/td/docs/telepresence/endpoint/ce98/sx-mx-dx-room-kit-boards-customization-guide-ce98.pdf>

QUESTION 4

DRAG DROP

Drag and drop the definitions from the left onto the correct Webex Teams REST API HTTP response status codes on the right.

Select and Place:

Too many requests have been sent in a given amount of time, and the request has been rate limited. A Retry-After header that specifies how many seconds you need to wait before a successful request can be made should be present.	401
The request to a resource used an unsupported HTTP request method.	403
Authentication credentials were missing or incorrect.	405
The request is understood but has been refused, or access is not allowed.	429

Correct Answer:

Authentication credentials were missing or incorrect.

The request is understood but has been refused, or access is not allowed.

The request to a resource used an unsupported HTTP request method.

Too many requests have been sent in a given amount of time, and the request has been rate limited. A Retry-After header that specifies how many seconds you need to wait before a successful request can be made should be present.

Reference: <https://developer.webex.com/docs/api/basics>

QUESTION 5

DRAG DROP

Drag and drop the methods from the left into the correct order of execution on the right to use webex-js-sdk in a browser to call and share the screen with another Webex user. Not all methods are used.

Select and Place:

meeting.addMedia()	step 1
meeting.updateShare()	step 2
webex.call()	step 3
call.getMediaStreams()	step 4
webex.meetings.create()	
meeting.getMediaStreams()	

Correct Answer:

	webex.meetings.create()
	meeting.getMediaStreams()
webex.call()	meeting.addMedia()
call.getMediaStreams()	meeting.updateShare()

Reference: https://github.com/webex/webex-js-sdk/blob/master/packages/node_modules/%40webex/plugin-meetings/README.md (see start wireless share)

QUESTION 6

Which two statements about incoming WebEx call notifications on iOS and Android applications are true? (Choose two.)

- A. When running in the background on Android, Firebase Cloud Messaging must use a notification message.
- B. When running in the background on Android, Firebase Cloud Messaging must use a data message.

- C. Webhooks are required with the iOS SDK only when the application is in the background.
- D. When running in the background on iOS, Application Push Notifications must be configured to be silent.
- E. Webhooks are required in the foreground and background for iOS and Android applications.

Correct Answer: BC

Reference: <https://github.com/webex/webex-android-sdk-example-push-notification-server/blob/master/README.md>
<https://developer.webex.com/docs/sdks/ios>

QUESTION 7

DRAG DROP

Drag and drop the components to create the authentication header used for the Webex Teams API. Not all options are used.

Select and Place:

<input type="text"/>	:	<input type="text"/>	<input type="text"/>
----------------------	---	----------------------	----------------------

- | | |
|----------------------------------|----------------|
| admin:Password12345! | Authorization |
| Token | Bearer |
| BH782kjh-2jkhmnb3rlkj3lkjnm3klj3 | Authentication |
| Basic | |

Correct Answer:

Authorization	:	Bearer	Token
---------------	---	--------	-------

- | | |
|----------------------------------|----------------|
| admin:Password12345! | |
| BH782kjh-2jkhmnb3rlkj3lkjnm3klj3 | Authentication |
| Basic | |

Reference: <https://developer.webex.com/docs/guest-issuer>

QUESTION 8

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. And the webhook is disabled.
- C. The webhook secret is expired and must be refreshed.
- D. The refresh token is not being used.
- E. The bot owner regenerated the access token on developer.webex.com.

Correct Answer: AB

Reference: <https://developer.authorize.net/api/reference/features/webhooks.html>

QUESTION 9**DRAG DROP**

Drag and drop the code to complete the JavaScript code snippet to create a meeting using the Webex Meetings XML API. Options may be used more than once.

Select and Place:

```

<script>
  const init = {
    method: 'POST'
    body: <message xmlns:xsi= "http://www.w3.org/2001/XMLSchema-instance">
      <header><securityContext>
        <siteName>apidemoeu</siteName>
        <webExID>alice</webExID>
        <[redacted]>AAABb6FFuSBR19BRERJTkdfU0hBMjU2X0FMR09SSVRITV8=
      </[redacted]>
      </securityContext></header>
      <body>
        <bodyContent xsi:type="*ava:com.webex.service.binding.meeting.CreateMeeting">
          <accessControl><[redacted]>Cisco1234</[redacted]>
        </accessControl>
        <metaData><confName>Sample Meeting</confName>
        <meetingType>105</meetingType></metaData>

        <schedule><startDate>[redacted]</startDate></schedule>
      </bodyContent>
    </body></message>' }
  fetch('https://api.webex.com/WBXService/XMLService', init)
    .then(response -> response.text())
    .then(str => (new window.DOMParser()).parseFromString(str, '[redacted]'))

    .then(data => document.write(data.getElementsByTagNameNS('*', '[redacted]')
[0].textContent))
</script>

```

result	application/json
text/xml	1/13/2020 16:00:00
meetingPassword	sessionTicket
Jan 24, 2019 4:00 PM	status

Correct Answer:

```

<script>
  const init = {
    method: 'POST'
    body: <message xmlns:xsi= "http://www.w3.org/2001/XMLSchema-instance">
      <header><securityContext>
        <siteName>apidemoeu</siteName>
        <webExID>alice</webExID>
        < [redacted] sessionTicket >AAABb6FFuSBR19BRERJTkdfU0hBMjU2X0FMR09SSVRITV8=
      </ [redacted] status >
      </securityContext></header>
      <body>
        <bodyContent xsi:type="*ava:com.webex.service.binding.meeting.CreateMeeting">
          <accessControl>< [redacted] meetingPassword >Cisco1234</ [redacted] meetingPassword >
        </accessControl>
        <metaData><confName>Sample Meeting</confName>
        <meetingType>105</meetingType></metaData>

        <schedule><startDate> [redacted] Jan 24, 2019 4:00 PM </startDate></schedule>
      </bodyContent>
    </body></message> }
  fetch('https://api.webex.com/WBXService/XMLService', init)
    .then(response => response.text())
    .then(str => (new window.DOMParser()).parseFromString(str, ' [redacted] text/xml '))

    .then(data => document.write(data.getElementsByTagNameNS('*', ' [redacted] result ')))
  [0].textContent)
</script>

```

result	application/json
text/xml	1/13/2020 16:00:00
meetingPassword	sessionTicket
Jan 24, 2019 4:00 PM	status

QUESTION 10

```
1 const webex = windows.webex = Webex.init();
2
3 webex.once('ready', () => {
4   const jwt = document.getElementById('context').value;
5
6   webex.authorization.requestAccessTokenFromJwt({ jwt })
7     .then(() => {
8       if (webex.canAuthorize) {
9         alert('Authenticated!');
10      }
11    })
12    .catch((err) => {
13      alert('Authentication.error:' + err);
14    });
15});
```

Refer to the exhibit. On line 4, the script retrieves a context from a DOM element that was generated from a server-side component. How does that server-side component obtain the value for the `context` element?

- A. by opening a dialog asking the end-user to paste his personal access token
- B. by completing an authorization code grant flow using the identifier and secret of an OAuth integration
- C. by embedding the access token of a Bot account
- D. by creating a guest token using the identifier and secret of a Guest Issuer application

Correct Answer: B

QUESTION 11

```
const Webex = require('webex');

const webex = Webex.init({
  orcdentials: {
    access_token: 'A_VALID_ACCESS_TOKEN'
  }
});
```

Refer to the exhibit. When using the Webex Browser SDK to create calls and share screens, which two statements are valid given a `webex` object such as displayed in the exhibit? (Choose two.)

- A. After a meeting is joined, it cannot be left programmatically until the host ends the meeting.
- B. The webex meetings.register() function must be invoked before attempting to join any meeting.

- C. The joinMeeting() function throws an error of type `media stopped` if a media stream is stopped.
- D. Given a Webex meeting number the webex meetings join() function can be used to join the meeting.
- E. The mediaSettings for a joined meeting accepts boolean attributes to send and receive audio, video, and screen share.

Correct Answer: AB

Reference: <https://developer.webex.com/docs/sdks/browser>

QUESTION 12

```
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");`

Correct Answer: D

The post method can be used for HTTP request that sets up a personal meeting room PIN for a user.

QUESTION 13

```
const webex = require('webex/env');
const BOT_ID = 'Y2ABCDEFG3VzL1BFT1BMRS9mNWTy1jGIyZilm';
// ...
let notification = {
  "id": "Y21zY29zcGFyazovL3VzL1dFQkwMQ50DgxNjA5NDk3",
  "name": "notifier",
  "resource": "messages",
  "event": "created",
  "filter": "roomId=Y21zy29zcGFyazovL3VzL1JyYjU4LtkxNDctZjE0YmIwYzRkMTU0",
  "orgId": "OTZhYmMyWEtM2RjYy0xMWUuZnDgxOWNkYz1h",
  "createdBy": "Y21zY29zcGFyazovL3VzL0FQUEExOTdjODUwM2YyNjZhYmY2NmMSOTIiYzFm",
  "ownedBy": "creator",
  "status": "active",
  "actorId": "Y21zY29zcGFyazovL3VzL1BFT18MRS9mNWIyijGIyZilmOWMONDdmMjkwNDY",
  "data": {
    "id": "Y21zY29zcGFyazovL3VzLz1YjAtMzRkNC0xMWVhLTgwZmMtYjFlNmRiMjk0YjM5",
    "roomId": "Y21zY29zcGFyazovL3VzLiJPT00vYmJjZWZyYjU4LtkxNDctZjE0YmIwYzRkMTU0",
    "personId": "Y21zY29zcGFyazovL3VzL1BFTijctOGIyZilmOWMONDdmMjkwNDY",
    "personEmail": "alice@example.com",
    "created": "2019-10-18T14:26:26.000Z"
  }
};
```

Refer to the exhibit. A webhook has been created so that an application is notified when users mention a bot in a Webex Teams space. The exhibit shows an example of a notification received by the application. Which code snippet correctly processes the JSON payload using the Webex Node.js SDK in order to print out messages that mention the bot?

- A.
- ```
if ((notification.name == 'notifier') &&
 (notification.event == 'created')) {
 if (notification.resource == 'messages') {
 webex.messages.get(notification.data.id).then(
 msg => console.log(msg.text));
 }
}
```
- B.
- ```
if ((notification.resource == 'messages') &&
    (notification.event == 'created')) {
  webex.messages.get(notification.data).then(
    msg => console.log(msg.text));
}
```
- C.
- ```
if ((notification.resource == 'messages') &&
 (notification.event == 'created')) {
 if (notification.data.personId != BOT_ID) {
 webex.messages.get(notification.data.id).then(
 msg => console.log(msg.text));
 }
}
```
- D.
- ```
if ((notification.resource == 'notifier') &&
    (notification.event == 'created')) {
  if (notification.data.personId != BOT_ID) {
    webex.messages.get(notification.data.id).then(
      msg => console.log(msg.text));
  }
}
```

A. Option A

B. Option B

C. Option C

D. Option D

Correct Answer: B

QUESTION 14

```
{
  "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-01BE-007E-2D30B4D4007E"
}
```

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

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

Correct Answer: C

Reference: <https://developer.webex.com/docs/api/v1/messages/get-message-details>

QUESTION 15

DRAG DROP

Drag and drop the code to create a valid Guest Issuer JWT token to use with the Webex Teams Browser SDK. Not all options are used.

Select and Place:


```
const jwt = require('jsonwebtoken')
let token = jwt.sign(
  {
    'sub': ,
    'name': ,
    'iss': ,
  },
  Buffer.from(, 'base64'),
  { expiresIn:  }
);
```

Correct Answer:

```
const jwt = require('jsonwebtoken')
let token = jwt.sign(
  {
    'sub': 'User Identifier',
    'name': 'User Display Name',
    'iss': 'Guest Application Identifier',
  },
  Buffer.from('Guest Application Secret', 'base64'),
  { expiresIn: 'Time span or seconds count' }
);
```

'Guest User Key'

Reference: <https://developer.webex.com/docs/guest-issuer>

[300-920 VCE Dumps](#)

[300-920 Exam Questions](#)

[300-920 Braindumps](#)