

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.pass2lead.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

Which list of API calls must be made to provision 50 new Webex Room devices for an organization?

- A. POST to /places with `displayName`, POST the returned `placeId` to /devices/
- B. POST `displayName` to /places, POST the returned `placeId` to /devices/activationCode
- C. POST to /licenses, POST the returned `licenseId` and a specified `displayName` to /devices
- D. POST to /licenses, POST `displayName` to /places, POST `licenseId` and `placeId` to /devices

Correct Answer: A

Reference: <https://developer.webex.com/docs/api/v1/devices/list-devices>

QUESTION 2

DRAG DROP

Drag and drop the code to complete the snippet that sends a message to the room "My First Room" using the Browser SDK. Not all options are used.

Select and Place:

```
var webex = require ('webex/env');
webex.rooms.list({
  max: 10
})
  .then(function(rooms) {
    var room = rooms.items.filter(function (room) {
      return room.title === 'My First Room!';
    }) [0];

    return webex.  .  ({
       'Hello World!',
       room.id
    });
  })

//Make sure to log errors in case something goes wrong.
.catch (function(reason) {
  console.error(reason);
  process.exit(1);
});
```

- | | |
|---------------------------------------|---------------------------------------|
| <input type="text" value="send"/> | <input type="text" value="text:"/> |
| <input type="text" value="create"/> | <input type="text" value="roomId:"/> |
| <input type="text" value="messages"/> | <input type="text" value="message:"/> |

Correct Answer:

```
var webex = require ('webex/env');
webex.rooms.list({
  max: 10
})
  .then(function(rooms) {
    var room = rooms.items.filter(function (room) {
      return room.title === 'My First Room!';
    }) [0];

    return webex.messages.create ({
      text: 'Hello World!',
      roomId: room.id
    });
  })

//Make sure to log errors in case something goes wrong.
.catch (function(reason) {
  console.error(reason);
  process.exit(1);
});
```

send

message:

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

QUESTION 3

```
1  const WebExClient = require('webex-api-client');
2  const securityContext = {
3    webExID: 'Test User',
4    password: 'pass123',
5    siteName: 'hello-world'
6  };
7
8  const requestBuilder = new WebExClient.Builder
9  (securityContext, 'https://hello-world.webex.com/WBXService/XMLService');
10
11 const createMeeting =
12   requestBuilder
13   .metaData({
14     confName: 'Sample Meeting',
15     meetingType: 1,
16   })
17   .participants({
18     attendees: [
19       {
20         name: 'Jane Doe',
21         email: 'jdoe@gmail.com'
22       }
23     ]
24   })
25   .schedule({
26     startDate: new Date(),
27     _____,
28     _____
29   })
30   .setService('CreateMeeting')
31   .build();
32
33 createMeeting
34   .exec()
35   .then((resp) => console.log('success'));
36
37
```

Refer to the exhibit. Which code for blank lines 26, 27, and 28 gives invitees 900 seconds before the scheduled time to join the meeting, sets the meeting to last for 30 minutes, and sets the meeting timezone to Pacific US?

- A.

```
26 openMeeting: 900
27 stopTime: 30,
28 timeZone: 4
```
- B.

```
26 openTime: 900
27 duration: 30,
28 timeZoneID: 4
```
- C.

```
26 joinTime: 900
27 meetingLength: 30,
28 timezone: 5
```
- D.

```
26 joinTime: 900
27 endTime: 30,
28 timezoneID: pt_us
```

A. Option A

B. Option B

C. Option C

D. Option D

Correct Answer: B

Reference: <https://github.com/cisco-ie/webex-api-client>

QUESTION 4

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 5

DRAG DROP

Drag and drop the code to complete the JavaScript to display the first URL from a user's list of Webex Meetings recordings. Not all options are used.

Select and Place:

```
const init = { [redacted] },
body: '<message xmlns:xsi= "http://www.v3.org/2001/XMLSchema-instance">
  <header><securityContext>
    <siteName>apidemoeu</siteName>
    <webExID>alice</webExID>
    <sessionTicket>AAABb6CkTRgAABUYARZAD2X0FMR09SSVRITV8=</sessionTicket>
  </securityContext></header>
  <body>
    <bodyContent xsi:type=" [redacted] ">

    <createTimeScope>
      <createTimeStart>11/21/2019 0:0:0</createTimeStart>
      <createTimeEnd>11/21/2019 23:59:59</createTimeEnd>
    </createTimeScope>
  </bodyContent></body></message> };
fetch(' [redacted] ',init)
  .then(response => response.text())
  .then(str => (new window.DOMParser()).parseFromString(str, 'text/xml'))
  .then(data => document.write(data.getElementsByTagNameNS(
    ' [redacted] ',
    'fileURL' ) [0].textContent));
```

`http://www.webex.com/schemas/
2002/06/service/meeting`

`method: 'POST'`

`java:com.webex.service.binding
.ep.LstRecording`

`java:com.webex.service.binding
.meeting.Recordings`

`http://www.webex.com/schemas/
2002/06/service/ep`

`method: 'GET'`

`https://api.webex.com/WBX
Service/XMLService`

`https://api.webex.com/v1/
recordings`

Correct Answer:


```
const init = { method: 'POST' },
body: '<message xmlns:xsi= "http://www.v3.org/2001/XMLSchema-instance">
  <header><securityContext>
    <siteName>apidemoeu</siteName>
    <webExID>alice</webExID>
    <sessionTicket>AAABb6CkTRgAABUYARZAD2X0FMR09SSVRITV8=</sessionTicket>
  </securityContext></header>
  <body>
    <bodyContent xsi:type="java:com.webex.service.binding.ep.LstRecording">
      <createTimeScope>
        <createTimeStart>11/21/2019 0:0:0</createTimeStart>
        <createTimeEnd>11/21/2019 23:59:59</createTimeEnd>
      </createTimeScope>
    </bodyContent></body></message> };
fetch('https://api.webex.com/WBXService/XMLService', init)
  .then(response => response.text())
  .then(str => (new window.DOMParser()).parseFromString(str, 'text/xml'))
  .then(data => document.write(data.getElementsByTagNameNS(
    'http://www.webex.com/schemas/2002/06/service/ep',
    'fileURL') [0].textContent));
```

http://www.webex.com/schemas/2002/06/service/meeting

java:com.webex.service.binding.meeting.Recordings

method: 'GET'

https://api.webex.com/v1/recordings