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.passapply.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 two statements describe valid limitations for Webex Teams bot accounts in a group space? (Choose two.)

A. A bot account cannot send a message without an SSL certificate.

B. A bot account cannot hear a message unless the bot is @ mentioned.

C. A bot account cannot send a message without a valid access token.

D. A bot account cannot respond to another bot.

E. A bot account cannot send a message with an @mention.

Correct Answer: BC

Reference: https://developer.webex.com/docs/bots

QUESTION 2

Which REST API request is used to list all the Webex Room Kit devices within a large organization so that a new custom In-Room Control can be deployed on all the devices?

A.
```
var request = require("request");
var options = { method: 'GET',
        url: 'https://api.ciscospark.com/v1/devices',
        qs: { product: 'Roomkit' },
        headers:
          { 'Content Type': 'application/json',
            Authorization: 'Bearer Yz6FgoWx7Pgb57C9z' }};

request(options, function(error, reponse, body) {
        if (error) throw new Error(error);
        console.log(body);
});
```

B.
```
var request = require("request");
var options = { method: 'GET',
        url: 'https://api.ciscospark.com/v1/devices',
        qs: { product: 'Roomkit' , placeID: 'Yzb60gRx3kBq5iB2w' },
        headers:
          { 'Content Type': 'application/json',
            Authorization: 'Bearer Yz6FgoWx7Pgb57C9z' }};

request(options, function(error, reponse, body) {
        if (error) throw new Error(error);
        console.log(body);
});
```

C.
```
var request = require("request");
var options = { method: 'GET',
        url: 'https://api.ciscospark.com/v1/devices/Yzb60gRx3kBq5iB2w'
        qs: { deviceName: 'Roomkit' },
        headers:
         { 'Content Type': 'application/json',
           Authorization: 'Bearer Yz6FgoWx7Pgb57C9z' }};

request(options, function(error, reponse, body) {
        if (error) throw new Error(error);
        console.log(body);
});
```

D.
```
var request = require("request");
var options = { method: 'GET',
        url: 'https://api.ciscospark.com/v1/devices',
        qs: { upgradeChannel: 'Roomkit' },
        headers:
         { 'Content Type': 'application/json',
           Authorization: 'Bearer Yz6FgoWx7Pgb57C9z' }};

request(options, function(error, reponse, body) {
        if (error) throw new Error(error);
        console.log(body);
});
```

A. Option A

B. Option B

C. Option C

D. Option D

Correct Answer: A

The qs: option is required to list all roomkit devices. Product: `RoomKit\\' is the correct option because it will list all roomkit devices.

QUESTION 3

DRAG DROP

Drag and drop the code segments to construct a script that carries out these functions:

1.

adds a new room

2.

adds two users, Bob, and Alice, to the room

3.

when both the users have been added, sends a welcome message

Select and Place:

```
const Webex = require('webex');
const webex = Webex.init{{
credentials {
access_token: "VALID TOKEN"
}
});
```

```
                         (empty box)
```

```
.then(room=>{
```

```
                         (empty box)
```

```
                         (empty box)
```

```
                         (empty box)
```

```
])then( ()=> {
```

```
                         (empty box)
```

```
});
```

```
});
```

```
webex.messages.create({
markdown: '**Hi Everyone**',
roomId: room.id
})
```

```
webex.membership.create({
roomId: room.id,
personEmail: 'bob@example.com'
})
```

```
return Promise.all([
```

```
webex.rooms.create(title: 'My First Room'])
```

```
webex.membership.create({
roomId: room.id,
personEmail: 'alice@example.com'
})
```

Correct Answer:

```
const Webex = require('webex');
const webex = Webex.init{{
credentials {
access_token: "VALID TOKEN"
}
});

    webex.rooms.create(title: 'My First Room'])

.then(room=>{

    return Promise.all([

    webex.membership.create({
    roomId: room.id,
    personEmail: 'bob@example.com'
    })

    webex.membership.create({
    roomId: room.id,
    personEmail: 'alice@example.com'
    })

])then( ()=> {

    webex.messages.create({
    markdown: '**Hi Everyone**',
    roomId: room.id
    })

});

});
```

Reference: https://github.com/webex/webex-js-sdk/tree/master/packages/node_modules/webex (see usage)

QUESTION 4

Which two statements about using the Compliance Office role to manage data in Webex Teams are true? (Choose two.)

A. For group rooms with members from multiple organizations, only the compliance officer of the owning organization can manage all data.

B. The compliance officer can manage data only for users who have granted the spark-compliance messages scope.

C. For direct rooms between two organizations, each compliance officer can manage data only from their organization.

D. For group rooms with members from multiple organizations, either compliance offcer can manage all data.

E. For direct rooms between two organizations, the compliance officer of either organization can manage all data.

Correct Answer: CE

Reference: https://developer-portal-intb.ciscospark.com/docs/api/guides/compliance

---

QUESTION 5

Which XML snippet does an admin use with a Webex Meetings XML API request to retrieve info on multiple users?

A. .*

B. .*

C.

D.

Correct Answer: A

Reference: https://developer.cisco.com/docs/webex-xml-api-reference-guide/#!getuser