

[Download Full Version 350-901 Exam Dumps\(Updated in Feb/2023\)](#)

```
#!/usr/bin/python3
import requests, sys

head = { 'Content-Type': '<item 1>',
        'Authorization': 'Bearer NWU4NjQ0ODJkZTItM...4-ad72cae0e10f' }

res = requests.post(url = 'https://api.ciscopark.com/v1/<item 2>',
                    headers = head, json = { '<item 3>': sys.argv[1] })
spaceId = res.json()['id']

members = [ 'johndoe@example.com', 'janedoe@example.com' ]
for member in members:
    res = requests.post(url='https://api.ciscopark.com/v1/<item 4>',
                        headers = head,
                        json = { 'roomId' spaceId, '<item 5>': member})
```

Create a Room

Creates a room. The authenticated user is automatically added as a member of the room. See the [Memberships API](#) to learn how to add more people to the room. To create a 1:1 room, use the [Create Messages](#) endpoint to send a message directly to another person by using the `toPersonId` or `toPersonEmail` parameters.

POST /v1/rooms

Body Parameters

title

string **Required**

A user-friendly name for the room.

teamId

string

The ID for the team with which this room is associated.

Create a Membership

Add someone to a room by Person ID or email address; optionally making them a moderator.

POST /v1/memberships

Body Parameters

roomId

string **Required**

The room ID.

personId

string

The person ID.

personEmail

string

The email address of the person.

isModerator

boolean

Whether or not the participant is a room moderator.

[Download Full Version 350-901 Exam Dumps\(Updated in Feb/2023\)](#)

application/xml	<item 1>
application/json	<item 2>
name	<item 3>
userName	<item 4>
title	<item 5>
personEmail	
/members	
/memberships	
/rooms	
/spaces	

Correct Answer:

[Download Full Version 350-901 Exam Dumps\(Updated in Feb/2023\)](#)

application/xml	application/json
application/json	/rooms
name	title
userName	/memberships
title	personEmail
personEmail	
/members	
/memberships	
/rooms	
/spaces	

QUESTION 16

A network operations team is using the cloud to automate some of their managed customer and branch locations. They require that all of their tooling be ephemeral by design and that the entire automation environment can be recreated without manual commands. Automation code and configuration state will be stored in git for change control and versioning. The engineering high-level plan is to use VMs in a cloud- provider environment then configure open source tooling onto these VMs to poll, test, and configure the remote devices, as well as deploy the tooling itself. Which configuration management and/or automation tooling is needed for this solution?

- A. Ansible
- B. Ansible and Terraform
- C. NSO
- D. Terraform
- E. Ansible and NSO

Correct Answer: B

[Download Full Version 350-901 Exam Dumps\(Updated in Feb/2023\)](#)

QUESTION 17

Refer to the exhibit. Which set of API requests must be executed by a Webex Teams bot after receiving a webhook callback to process messages in a room and reply with a new message back to the same room?

Responding to Events

After creating a bot, you can use its access token with the Webex REST APIs to perform actions as the bot, such as [sending a message](#) with an interactive [card](#) to someone. To respond to events within Webex Teams, such as someone sending your bot a message or adding it to a group space, you'll need to configure webhooks. Webhooks will let you know when an activity has occurred so you can take action. Check out the [Webhooks Guide](#) for more information about configuring webhooks.

With cards, you can give your users even more ways to interact with your bot or service, right in the Webex Teams clients. See the [Cards Guide](#) for more information.

Differences Between Bots and People

One key difference between Webex Teams Bots and regular users is that, in group rooms, bots **only have access to messages in which they are mentioned**. This means that `messages:created` webhooks only fire when the bot is mentioned in a room.

Also, [listing messages](#) requires that you specify a special `?mentionedPeople=me` query parameter.

```
GET /messages?mentionedPeople=me&roomId=SOME_INTERESTING_ROOM
Authorization: Bearer THE_BOTS_ACCESS_TOKEN
```

Bot Frameworks & Tools

There are several bot frameworks that can greatly simplify the bot development process by abstracting away the low-level communications with the Webex REST API, such as creating and sending API requests and configuring webhooks. Instead, you can focus on the building the interaction and business logic of your bot.

[Flint](#) is an open source bot framework with support for regex pattern matching for messages and more.