

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

```
base_url = "https://api.meraki.com/api/v0"
network_id = "N_1234567890"
requests.put(
    base_url + "/networks/" + network_id + "/ssids/0",
    headers = {
        "X-Cisco-Meraki-API-Key": api_key,
        "Content-Type": "application/json"
    },
    data = json.dumps ({
        "number": 0,
        "name": ssid,
        "enabled": True,
        "authMode": "psk",
        "psk": wireless_password,
        "encryptionMode": "wpa",
        "wpaEncryptionMode": "WPA2 only"
    })
```

application/json organization_id network_id

wireless_password base_url

api_key encryptionMode

QUESTION 25

Which snippet presents the correct API call to configure, secure, and enable an SSID using the Meraki API?

A.

```
curl -X PUT \
  -url 'https://api.meraki.com/api/v0/networks/:networkId/ssids/2' \
  -H 'X-Cisco-Meraki-API-Key: 15da0c6ffff295f16267f88f98694cf29a86ed87' \
  -H 'Accept: application/json' \
  -H 'Content-type: application/json' \
  --data-raw '{
    "name": "My SSID",
    "enabled": false,
    "authMode": "psk",
    "encryptionMode": "wpa",
    "psk": "meraki123",
    "wpaEncryptionMode": "WPA1 and WPA2"
  }'
```

B.

[350-901 Exam Dumps](#) [350-901 PDF Dumps](#) [350-901 VCE Dumps](#) [350-901 Q&As](#)

<https://www.ensurepass.com/350-901.html>

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

```
curl -X PUT \
--url 'https://api.meraki.com/api/v0/networks/:networkId/ssids/2' \
-H 'X-Cisco-Meraki-API-Key: 15da0c6ffff295f16267f88f98694cf29a86ed87' \
-H 'Accept: application/json' \
-H 'Content-type: application/json' \
--data-raw '{
  "name": "My SSID",
  "enabled": true,
  "authMode": "psk",
  "encryptionMode": "wpa",
  "psk": "meraki123",
  "wpaEncryptionMode": "WPA1 and WPA2"
}'
```

C.

```
curl -X PUT \
--url 'https://api.meraki.com/api/v0/networks/:networkId/ssids/2' \
-H 'X-Cisco-Meraki-API-Key: 15da0c6ffff295f16267f88f98694cf29a86ed87' \
-H 'Accept: application/json' \
-H 'Content-type: application/json' \
--data-raw '{
  "enabled": true,
  "useVlanTagging": true
}'
```

D.

```
curl -X PUT \
--url 'https://api.meraki.com/api/v0/networks/:networkId/ssids/2' \
-H 'X-Cisco-Meraki-API-Key: 15da0c6ffff295f16267f88f98694cf29a86ed87' \
-H 'Accept: application/json' \
-H 'Content-type: application/json' \
--data-raw '{
  "name": "My SSID",
  "enabled": true,
}'
```

- A. Option A
- B. Option B
- C. Option C
- D. Option D

Correct Answer: B

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

QUESTION 26

DRAG DROP

Drag and drop the expressions from below onto the code to implement error handling. Not all options are used.

```
base_url = "https://api.meraki.com/api/v0"
posturl = '%s/networks/%s/staticRoutes' % ((str(base_url), str(networked)))
headers = {
    'x-cisco-meraki-api-key': api_key,
    'Content-Type': 'application/json'
}
routes = [ {
    "subnet": "10.16.4.0/22",
    "gatewayIp": "10.1.0.20",
    "name": "ROUTE1",
    "enabled": true
}
{
    "subnet": "10.253.254.0/24",
    "gatewayIp": "10.1.0.20",
    "name": "ROUTE2",
    "enabled": true
}
{
    "subnet": "10.168.0.0/21",
    "gatewayIp": "10.1.0.20",
    "name": "ROUTE3",
    "enabled": true
} ]

for route in routes:
    print("Adding static: " + str(route['subnet']))
    response = requests.post(posturl, json=route, headers=headers)
    
    print("Done!")
    
    print("Failed to add static: " + str(route['subnet']) + "\n" + response.text)
```

if response == 601:

else:

when:

if response == 201:

elif:

Correct Answer:

```
base_url = "https://api.meraki.com/api/v0"
posturl = '%s/networks/%s/staticRoutes' % ((str(base_url), str(networked)))
headers = {
    'x-cisco-meraki-api-key': api_key,
    'Content-Type': 'application/json'
}
routes = [ {
    "subnet": "10.16.4.0/22",
    "gatewayIp": "10.1.0.20",
    "name": "ROUTE1",
    "enabled": true
}
{
    "subnet": "10.253.254.0/24",
    "gatewayIp": "10.1.0.20",
    "name": "ROUTE2",
    "enabled": true
}
{
    "subnet": "10.168.0.0/21",
    "gatewayIp": "10.1.0.20",
    "name": "ROUTE3",
    "enabled": true
} ]

for route in routes:
    print("Adding static: " + str(route['subnet']))
    response = requests.post(posturl, json=route, headers=headers)
    if response == 201:
        print("Done!")
    else:
        print("Failed to add static: " + str(route['subnet']) + "\n" + response.text)
```

if response == 601:

else:

when:

if response == 201:

elif:

QUESTION 27

DRAG DROP

Refer to the exhibit. A developer is creating a Python script to use the Webex Teams REST API to list joined spaces, retry after the server-specified amount of time if a "Too many requests" response is received, and print any other error that is received. Drag and drop the code snippets from the left onto the item numbers on the right that match the missing sections in the exhibit to complete the script. Not all code snippets are used.

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

405	<item 1>
429	<item 2>
minutes	<item 3>
seconds	<item 4>
response.headers['Retry-After']	
response.header	
response.status_code	
response.status	

Correct Answer:

405	response.status_code
429	429
minutes	response.headers['Retry-After']
seconds	seconds
response.headers['Retry-After']	
response.header	
response.status_code	
response.status	