

CISCO *Live!*

GO BEYOND

#CiscoLiveAPJC



Preparing for the DevNet Associate Certification

Yusuf Bhajji, Director, Business Development
Kareem Iskander, Lead Technical Advocate
BRKCRT-2080

Cisco Webex App

Questions?

Use Cisco Webex App to chat with the speaker after the session

How

- 1 Find this session in the Cisco Live Mobile App
- 2 Click “Join the Discussion”
- 3 Install the Webex App or go directly to the Webex space
- 4 Enter messages/questions in the Webex space

Webex spaces will be moderated by the speaker until November 15, 2024.



<https://ciscolive.ciscoevents.com/ciscolivebot/#BRKCRT-2080>



Agenda

- Introduction to Cisco Certifications
- Certification Roadmap
- What to expect on the exam and how to prepare
- Exam vs Real life

Introduction to Cisco Certifications

“DevNet professional, DEVASC and Automation skills Required”



Network skills or Developer skills

200-301 CCNA

1.0 Network Fundamentals

2.0 Network Access

3.0 IP Connectivity

4.0 IP Services

5.0 Security Fundamentals

6.0 Automation and Programmability

VS

200-901 DEVASC

1.0 Software Development and Design

2.0 Understanding and Using APIs

3.0 Cisco Platforms and Development

4.0 Application Deployment and Security

5.0 Infrastructure and Automation

6.0 Network Fundamentals

Cisco Career Certifications

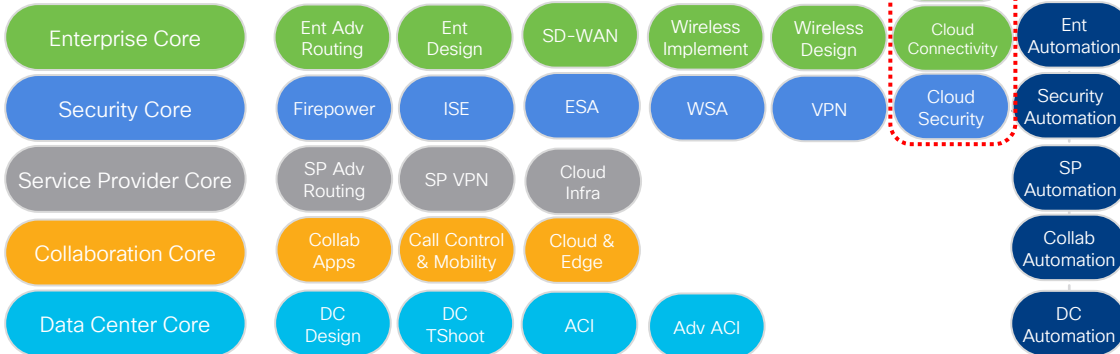


One Exam

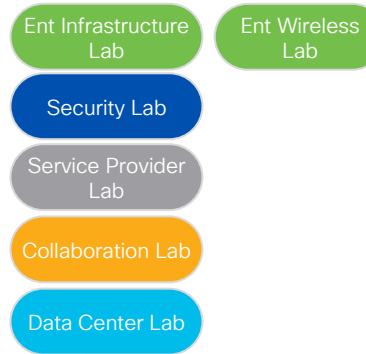
CCNA



Two Exams: Core + 1 Concentration



Core + Lab



One Exam

DevNet Associate



Two Exams: Core + 1 Concentration



Core + Lab



One Exam

CyberOps Associate



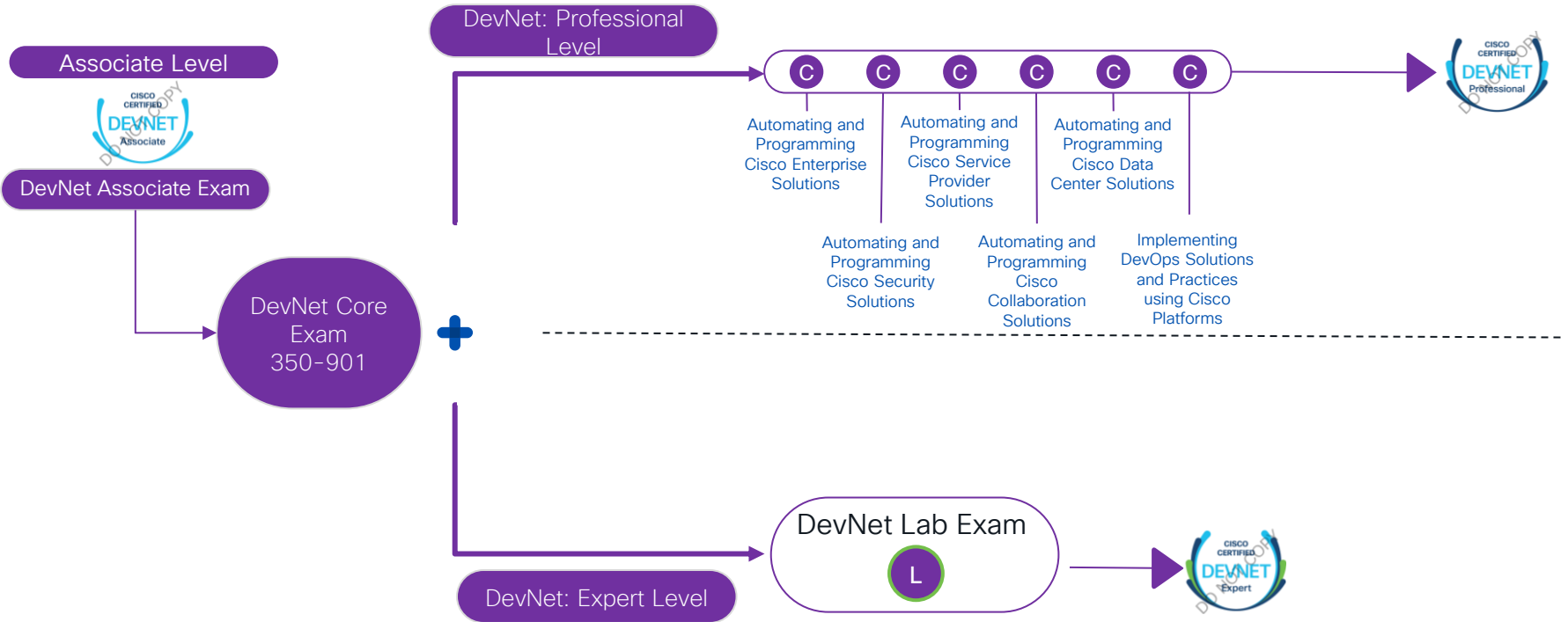
Two Exams: Core + 1 Concentration



CyberOps Expert



Cisco DevNet Certification Track



Certification roadmap publishing process

Stay in sync with the latest updates on a regular, rotating schedule

How it works:

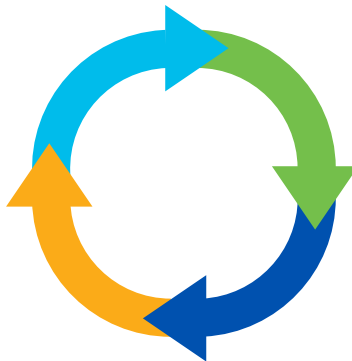
1. Cisco **reviews** each technology on the same quarterly schedule each year to make sure our exams align with the latest Cisco technologies.
2. We **announce** blueprint changes 3–6 months in advance along with revised exam topics and release notes, if applicable.
3. We **publish** the updated exam 3–6 months after the exam blueprint publication, if applicable.

Data Center & Collaboration

Q1: Review/Job Task analysis
Q2: New blueprints published
Q3: Updated exam go live

Security & CyberOps

Q4: Review/Job Task analysis
Q1: New blueprints published
Q2: Updated exam go live



Enterprise & DevNet

Q2: Review/Job Task analysis
Q3: New blueprints published
Q4: Updated exam go live

CCNA & Service Provider

Q3: Review/Job Task analysis
Q4: New blueprints published
Q1: Updated exam go live

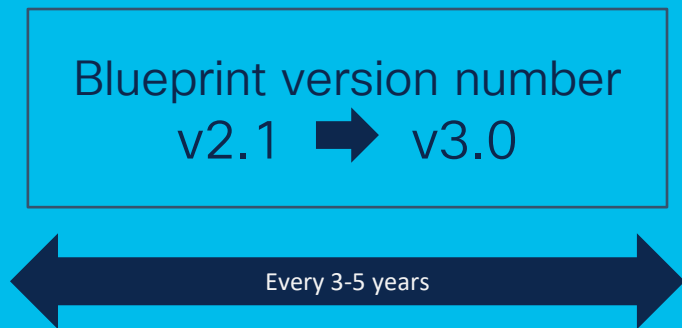
Dates shown reflect Cisco's fiscal year calendar.

Q1: August–October, Q2: November–January, Q3: February–April, Q4: May–July

Revision Framework

Major Revision

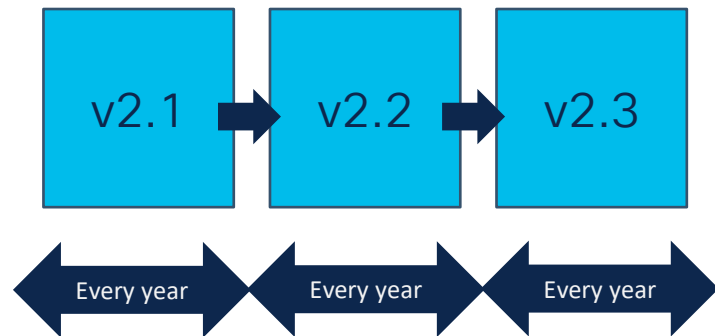
(Traditional Revision Model)



- Large revisions
- Major changes
- Steep learning curve
- Wider Alignment (Product & Technology)

Minor Revision

(Agile Revision Model)



- Smaller modular revisions
- Incremental changes
- Easy bite-size learning model
- Frequent alignment (Product & Technology)

Certification Roadmap

How it works:

1. Cisco **reviews** each technology on the same quarterly schedule each year to make sure our exams align with the latest Cisco technologies.
2. We **announce** blueprint changes 3-6 months in advance along with revised exam topics and release notes.
3. We **publish** the updated exam 3-6 months after the exam blueprint publication.



DEVASC v1.0 VS DEVASC v1.1

3.3: Describe the capabilities of Cisco compute management platforms and APIs (UCS Manager, ~~UCS Director~~, and Intersight)

3.4: Describe the capabilities of Cisco collaboration platforms and APIs (Webex ~~Teams~~, Webex devices, Cisco Unified Communication Manager including AXL and UDS interfaces, and Finesse)

3.5: Describe the capabilities of Cisco security platforms and APIs (~~XDR~~, Firepower, Umbrella, ~~Secure endpoint~~~~AMP~~, ISE, and Secure ~~Malware Analytics~~~~ThreatGrid~~)

3.9.b: Manage spaces, participants, and messages in Webex ~~Teams~~

5.3: Describe the use and roles of network simulation and test tools (such as ~~Cisco Modeling Labs~~~~VIRL~~ and pyATS)

5.6: Describe the capabilities of automation tools such as Ansible, ~~Terraform~~~~Chef~~, and Cisco NSO

<https://learningnetwork.cisco.com/s/cisco-certification-roadmaps>

DevNet

CyberOps

Service Provider

CCNA

Security

Enterprise

Collaboration

Data Center

FAQs

Exam Number

Release Notes

Exam Topic Blueprint

Learning Matrix

DevNet Exams

[Release Notes](#)

[DevNet v1.1
Learning Matrix](#)

DevNet Associate Exam Updates

First date to test: January 22, 2024

200-901 DEVASC v1.1

[Exam Topics](#)

DevNet Professional Core Exam Updates

First date to test: January 22, 2024

350-901 DEVCOR v1.1

[Exam Topics](#)

DevNet Professional Concentration Exam Updates

What to expect in the exam



Exam Blueprint

<https://learningnetwork.cisco.com/s/devnet-associate-exam-topics>

The Cisco Learning Network



Meet Cisco U. ▾

Certifications ▾

Communities ▾

Webinars & Videos ▾

Certifications / DevNet Associate Certification and Training Program / 200-901 DEVASC Exam Topics

200-901 DEVASC v1.1 Exam Topics

Exam Description

To earn your DevNet Associate certification, you must pass the **200-901 DEVASC** exam. This 120-minute exam tests your knowledge of:



Software
Development and
Design



Understanding
and Using APIs



Cisco Platforms
and Development



Application
Deployment and
Security



Infrastructure and
Automation



Network
Fundamentals

Expand each item below to view related exam topics.

1.0 Software Development and Design	15%	▾
2.0 Understanding and Using APIs	20%	▾
3.0 Cisco Platforms and Development	15%	▾
4.0 Application Deployment and Security	15%	▾
5.0 Infrastructure and Automation	20%	▾
6.0 Network Fundamentals	15%	▾

Interpret the Blueprint:

DevNet Associate (200-901)

Domain

Domain
Weight

15%

4.0 Application Development and Security

4.1 Describe benefits of edge computing

Task

4.2 Identify attributes of different application deployment models (private cloud, public cloud, hybrid cloud, and edge)

4.3 Identify the attributes of these application deployment types

4.3.a Virtual machines

Subtask

4.3.b Bare metal

Blueprint Verbs

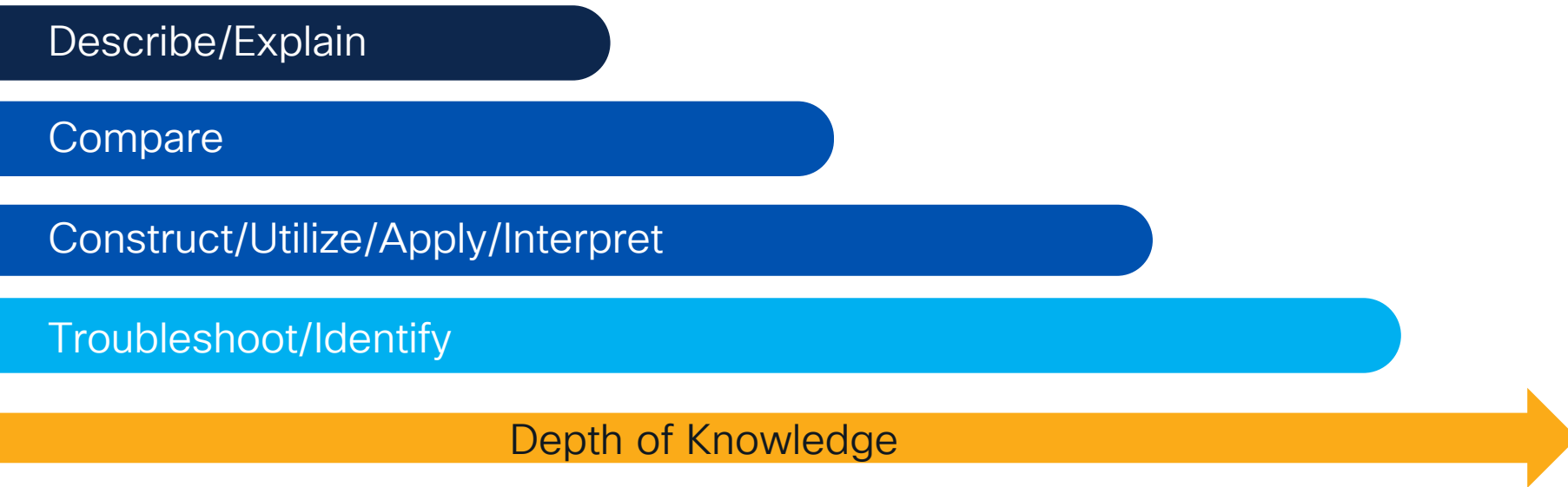
Describe/Explain

Compare

Construct/Utilize/Apply/Interpret

Troubleshoot/Identify

Depth of Knowledge



Types of questions



Multiple choice



Drag and drop



Lablets are coming!

Describe question

Example: Describe the functionality of these IP Services: DHCP, DNS, NAT, SNMP, NTP

Which protocol synchronises the clock between computer systems?

- A. NTP
- B. NAT
- C. DNS
- D. SNMP

Describe question

Example: Describe the functionality of these IP Services: DHCP, DNS, NAT, SNMP, NTP

Which protocol synchronises the clock between computer systems?

- A. NTP
- B. NAT
- C. DNS
- D. SNMP

Compare question

Example: Compare common API styles (REST and RPC)

What is the difference between REST and RPC APIs?

- A. REST APIs are stateless, and RPC APIs are stateful.
- B. REST APIs are vendor-specific, and RPC APIs are vendor-neutral.
- C. REST APIs are...
- D. REST APIs are...

OR like this

Drag and drop the characteristics from the left to the API style on the right.	
Options	Categories
stateful	Rest APIs
stateless	stateless
uses XML	uses XML
uses YAML	RPC APIs
	stateful
	uses YAML

Compare question

Example: Compare common API styles (REST and RPC)

What is the difference between REST and RPC APIs?

- A. REST APIs are stateless, and RPC APIs are stateful.
- B. REST APIs are vendor-specific, and RPC APIs are vendor-neutral.
- C. REST APIs are...
- D. REST APIs are...

OR like this

Drag and drop the characteristics from the left to the API style on the right.	
Options	Categories
stateful	Rest APIs
stateless	stateless
uses XML	uses XML
uses YAML	RPC APIs
	stateful
	uses YAML

How to prepare

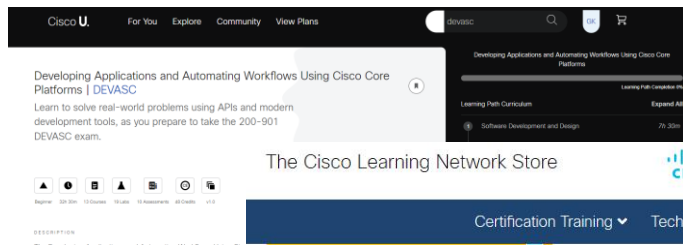
Describe/Explain

Compare

Construct/Utilize/Apply/Interpret

Troubleshoot/Identify

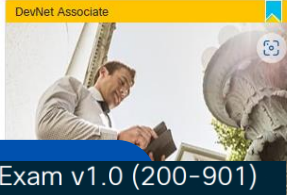
Depth of Knowledge



The Cisco Learning Network Store



Certification Training Technology Training All Training



DEVASC Preparation Bundle

Continuing Education Credits: 48



Access Duration: 180 days

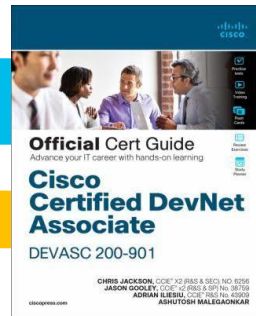
DevNet Associate Exam v1.0 (200-901)

DevNet Associate Exam v1.0 (DEVASC 200-901) is a 120-minute exam associated with the DevNet Associate - Developer Certification. This exam tests a candidate's knowledge of all the associate level in software development and design, understanding and using APIs, application development and security, and infrastructure and automation on Cisco platforms. The course, Developing Applications and Automating Workflows using Cisco Core Platforms, helps candidates to prepare for this exam.

Sign up for updates

Exam overview

15% 1.0 Software Development and Design	
Exam Topics	Study Material
1.1 Compare data formats (XML, JSON, YAML)	<p>These resources are meant to supplement your learning experience and exam preparation. They are NOT designed to serve as a complete self-study program, but intended only as a suggested starting point. Login to access these materials</p> <ul style="list-style-type: none">Setting up your Linux (Ubuntu) workstation as a development environmentSetting up your Windows workstation as a development environmentSetting up your macOS workstation as a development environment
1.2 Describe parsing of common data format (XML, JSON, YAML) to Python data structures	
1.3 Describe the concepts of test-driven development	
1.4 Compare software development methods (agile, lean, waterfall)	



cisco Live!

Construct question

Usually a code snippet with missing parts.

Example: Construct a Python unit test

Drag and the drop the code on the snippet to complete the Python unit test. Not all options are used.

```
import   
import logging  
  
log = logging.getLogger(__name__)  
log.setLevel(logging.DEBUG)  
  
 uniTest(unittest.TestCase):  
    def setUp(self):  
        log.info('I am doing the setUp')  
  
    def test_one(self):  
        log.info('one')  
  
    @classmethod  
    def tearDownClass(self):  
        log.info('tearDown class')  
  
     test_two(self):  
        log.info('two')  
  
    @classmethod  
    def setUpClass(self):  
        log.info('set up Class')
```

Construct question

Usually a code snippet with missing parts.

Example: Construct a Python unit test

Drag and the drop the code on the snippet to complete the Python unit test. Not all options are used.

```
import 
import logging

log = logging.getLogger(__name__)
log.setLevel(logging.DEBUG)

 uniFest(unittest.TestCase):
    def setUp(self):
        log.info('I am doing the setUp')

    def test_one(self):
        log.info('one')

    @classmethod
    def tearDownClass(self):
        log.info('tear down class')

     test_two(self):
        log.info('two')

    @classmethod
    def setUpClass(self):
        log.info('set up Class')
```

class

unittest

log

def

def

How to prepare

Describe/Explain

Compare

Construct/Utilize/Apply/Interpret

Troubleshoot/Identify

Depth

cisco *Live!*

The Cisco Learning Network Store



Certification Training Technology Training All Training

DevNet Associate

DEVASC Preparation Bundle

Continuing Education Credits: 48



Access Duration: 180 days

Cisco U. For You Explore Com

Developing Applications and Automatin

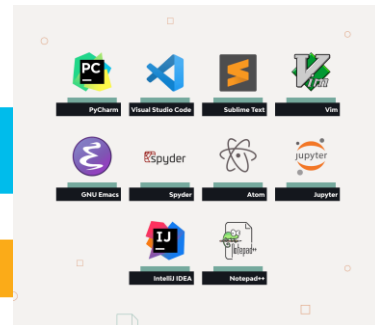
Platforms | DEVASC

Learn to solve real-world problems using APIs and modern development tools, as you prepare to take the 200-901 DEVASC exam.



REGISTRATION

The DevOps Applications and Automation Workflows Utilize Cisco Core Platforms



Troubleshoot question

Something is broken and we need to fix it.

Example: Troubleshoot a problem given the HTTP response code, request and API documentation

```
import requests
import json
url = "https://192.168.1.10:443/restconf/data/ietf-interfaces:interfaces"
payload = json.dumps({
    "ietf-interfaces:interface": {"name": "Loopback200", "description":
    "External LOOPBACK200 for VTI Tunnels", "type": "iana-if-type:softwareLoopback",
    "enabled": True, "ietf-ip:ipv4": {"address": [{"ip": "84.16.101.150",
    "netmask": "255.255.255.255"}]}}})
headers = {
    'Authorization': 'Basic e3t1c2VybmFtZXM19Ont7cGFzc3dvcmR9fQ==',
    'Accept': 'application/yang-data+json',
    'Content-Type': 'application/yang-data+json'}
response = requests.request("POST", url, headers=headers, data=payload)
print(r.status_code)
print(response.text)
```

Refer to the exhibit. A network engineer has developed a script to automate the provision of newly added devices on the network. In the lab environment where the script was tested, it was running with no issues. When applied to the customer side, it returned a **404 error**. What needs to be changed on the script to fix the error?

- A. The IP address on the script must be set to the customer's range.
- B. The device that needs to be provisioned must connect to the network.
- C. The script must be written in Yang instead of JSON for the devices to listen.
- D. The server running the script in a different range than the device to provision.

Troubleshoot question

Something is broken and we need to fix it.

Example: Troubleshoot a problem given the HTTP response code, request and API documentation

```
import requests
import json
url = "https://192.168.1.10:443/restconf/data/ietf-interfaces:interfaces"
payload = json.dumps({
    "ietf-interfaces:interface": {"name": "Loopback200", "description":
    "External LOOPBACK200 for VTI Tunnels", "type": "iana-if-type:softwareLoopback",
    "enabled": True, "ietf-ip:ipv4": {"address": [{"ip": "84.16.101.150",
    "netmask": "255.255.255.255"}]}}})
headers = {
    'Authorization': 'Basic e3t1c2VybmFtZXM19Ont7cGFzc3dvcmR9fQ==',
    'Accept': 'application/yang-data+json',
    'Content-Type': 'application/yang-data+json'}
response = requests.request("POST", url, headers=headers, data=payload)
print(r.status_code)
print(response.text)
```

Refer to the exhibit. A network engineer has developed a script to automate the provision of newly added devices on the network. In the lab environment where the script was tested, it was running with no issues. When applied to the customer side, it returned a **404 error**. What needs to be changed on the script to fix the error?

- A. The IP address on the script must be set to the customer's range.
- B. The device that needs to be provisioned must connect to the network.
- C. The script must be written in Yang instead of JSON for the devices to listen.
- D. The server running the script in a different range than the device to provision.

How to prepare

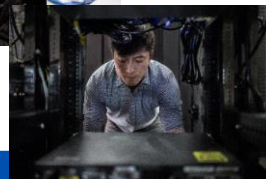
Describe/Explain

Compare

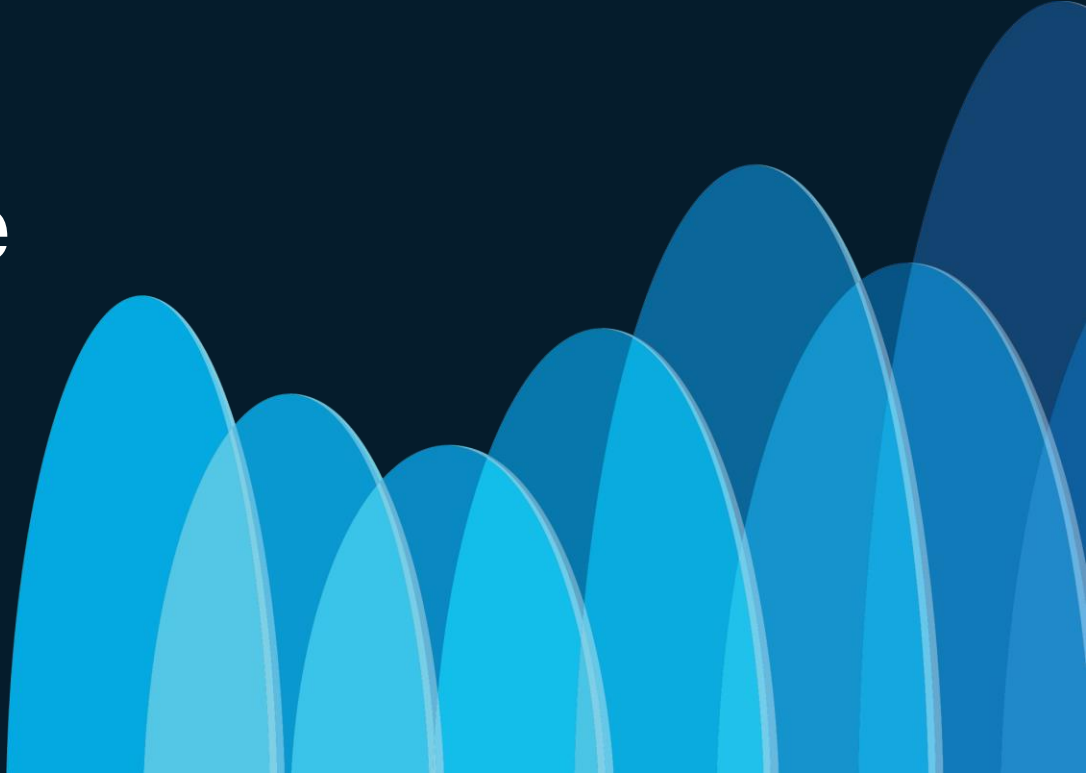
Construct/Utilize/Apply/Interpret

Troubleshoot/Identify

Depth of Knowledge



Exam vs Real life



What will be covered

- 1.8 Utilize common version control operations with Git
- 2.9 Construct a Python script that calls a REST API using the requests library
- 3.9 Construct code to perform a specific operation based on a set of requirements and given API reference documentation such as these:
 - 3.9.a Obtain a list of network devices by using Meraki, Cisco DNA Center, ACI, Cisco SD-WAN, or NSO
- 4.11 Utilize Bash commands (file management, directory navigation, and environmental variables)



Sample Code:

- Scan QR code

or

- <https://github.com/qsnyder/brkcrt-2080>

What will be covered

- 1.8 Utilize common version control operations with Git
- 2.9 Construct a Python script that calls a REST API using the requests library
- 3.9 Construct code to perform a specific operation based on a set of requirements and given API reference documentation such as these:
 - 3.9.a Obtain a list of network devices by using Meraki, Cisco DNA Center, ACI, Cisco SD-WAN, or NSO
- 4.1.1 Utilize Bash commands (file management, directory navigation, and environmental variables)

Utilize question

1.8 Utilize common version control operations with Git

1.8.a Clone

```
git  ssh://user@example.com/path/to/application.git
```

Refer to the exhibit. An engineer is working on a new feature for an existing application. The application is already live and the development must take place locally on the engineer's machine. Which Git command must the engineer use to create a copy of the application on the local machine?

- A. clone
- B. copy
- C. pull
- D. add

Utilize question

1.8 Utilize common version control operations with Git

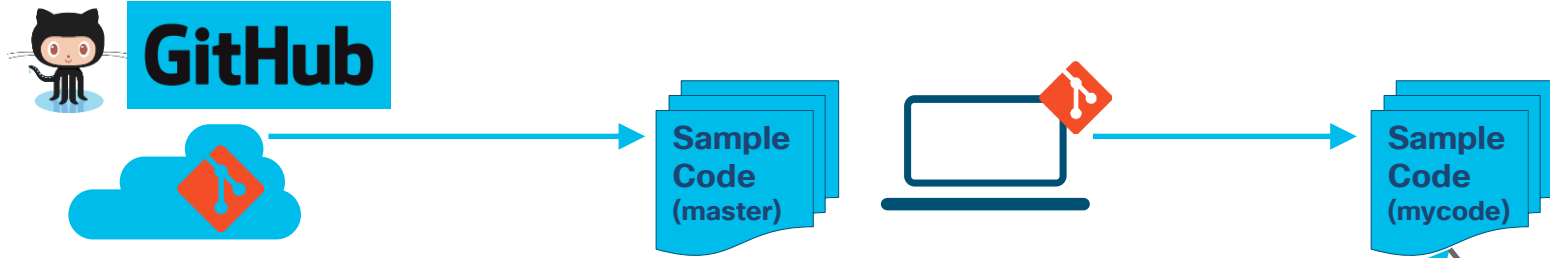
1.8.a Clone

```
git  ssh://user@example.com/path/to/application.git
```

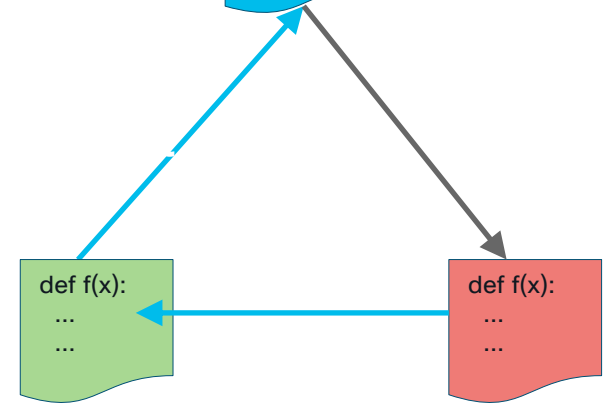
Refer to the exhibit. An engineer is working on a new feature for an existing application. The application is already live and the development must take place locally on the engineer's machine. Which Git command must the engineer use to create a copy of the application on the local machine?

- A. clone
- B. copy
- C. pull
- D. add

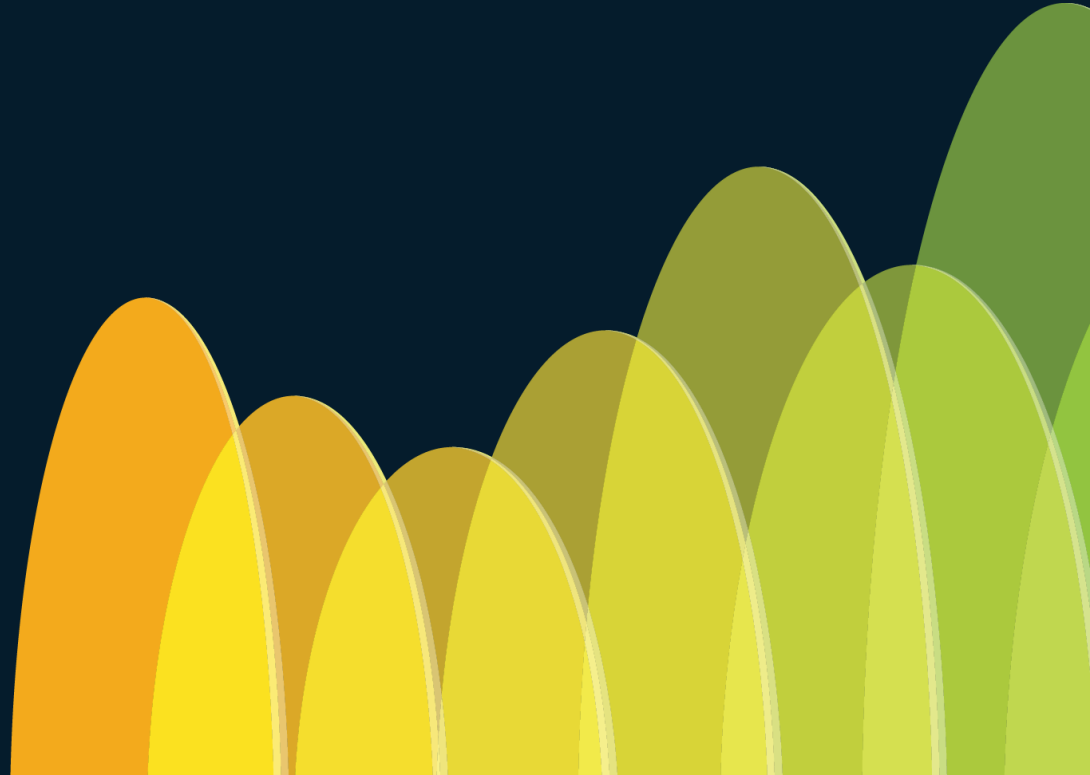
DevNet Sample-Code Workflow



Step	Action	Git Command
1.	Clone the Remote Repository	<code>git clone url</code>
2.	Create and Checkout a Local Branch	<code>git checkout -b new-branch-name</code>
3.	Incrementally Commit Changes	<code>git add filename</code> <code>git commit -m "Commit message"</code>



Git Hands-on Walkthrough



What will be covered

- 1.8 Utilize common version control operations with Git
- **2.9 Construct a Python script that calls a REST API using the requests library**
- 3.9 Construct code to perform a specific operation based on a set of requirements and given API reference documentation such as these:
 - 3.9.a Obtain a list of network devices by using Meraki, Cisco DNA Center, ACI, Cisco SD-WAN, or NSO
- 4.1.1 Utilize Bash commands (file management, directory navigation, and environmental variables)

Construct question

2.9 Construct a Python script that calls a REST API using the requests library

Drag and drop the code from the bottom onto the box where the code is missing to retrieve a list of organizations configured in the Meraki organization. Not all options are used.

```
import   
MERAKI_API_KEY = '6bec40cf957de430a6f1f2baa056b99a4fac9ea0'  
base_url = "https://api.meraki.com/api/v1"  
endpoint = "/organizations"  
header = {"X-Cisco-Meraki-API-Key": MERAKI_API_KEY}  
orgs = requests. (url=base_url+endpoint, headers=header)  
 = json.loads(orgs.text)  
pprint(orgs)
```

orgs

get

patch

requests

org

json

Construct question

2.9 Construct a Python script that calls a REST API using the requests library

Drag and drop the code from the bottom onto the box where the code is missing to retrieve a list of organizations configured in the Meraki organization. Not all options are used.

```
import   
MERAKI_API_KEY = '6bec40cf957de430a6f1f2baa056b99a4fac9ea0'  
base_url = "https://api.meraki.com/api/v1"  
endpoint = "/organizations"  
header = {"X-Cisco-Meraki-API-Key": MERAKI_API_KEY}  
orgs = requests.(url=base_url+endpoint, headers=header)  
 = json.loads(orgs.text)  
pprint(orgs)
```

orgs

get

patch

requests

org

json

What will be covered

- 1.8 Utilize common version control operations with Git
- 2.9 Construct a Python script that calls a REST API using the requests library
- 3.9 Construct code to perform a specific operation based on a set of requirements and given API reference documentation such as these:
 - 3.9.a Obtain a list of network devices by using Meraki, Cisco DNA Center, ACI, Cisco SD-WAN, or NSO
- 4.1.1 Utilize Bash commands (file management, directory navigation, and environmental variables)

Construct question

3.9.a Obtain a list of network devices by using Meraki, Cisco DNA Centre, ACI, Cisco SD-WAN, or NSO SDWAN

Drag and drop the code from the bottom onto the box where the code is missing to retrieve a list of devices in the Meraki organization. Not all options are used.

```
import requests
base_url = "https://api.meraki.com/api/v1"
[ ] = {"X-Cisco-Meraki-API-Key": MERAKI_API_KEY}
def get_device_list():
    for org in orgs:
        endpoint = "/organizations/{}/devices".format(org['id'])
        print([ ])
        resp = requests.get(url=base_url + endpoint, headers=header)
        [ ] = json.loads(resp.text)
        pprint(device)
if __name__ == "__main__":
    get_device_list()
```

header

device

auth

endpoint

orgs

list

Construct question

3.9.a Obtain a list of network devices by using Meraki, Cisco DNA Centre, ACI, Cisco SD-WAN, or NSO SDWAN

Drag and drop the code from the bottom onto the box where the code is missing to retrieve a list of devices in the Meraki organization. Not all options are used.

```
import requests
base_url = "https://api.meraki.com/api/v1"
[ ] = {"X-Cisco-Meraki-API-Key": MERAKI_API_KEY}
def get_device_list():
    for org in orgs:
        endpoint = "/organizations/{}/devices".format(org['id'])
        print([ ])
        resp = requests.get(url=base_url + endpoint, headers=header)
        [ ] = json.loads(resp.text)
        pprint(device)
if __name__ == "__main__":
    get_device_list()
```

header

device

auth

endpoint

orgs

list

What will be covered

- 1.8 Utilize common version control operations with Git
- 2.9 Construct a Python script that calls a REST API using the requests library
- 3.9 Construct code to perform a specific operation based on a set of requirements and given API reference documentation such as these:
 - 3.9.a Obtain a list of network devices by using Meraki, Cisco DNA Center, ACI, Cisco SD-WAN, or NSO
- **4.11 Utilize Bash commands (file management, directory navigation, and environmental variables)**

Utilize question

4.11 Utilize Bash commands

```
$virtualenv -p python3 <desired-path>  
$source <desired-path>/bin/
```

Refer to the exhibit. An engineer needs to complete the creation and enable a Python virtual environment. Which code needs to be placed on the box where the code is missing?

- A. activate
- B. enable
- C. allow
- D. power

Utilize question

4.11 Utilize Bash commands

```
$virtualenv -p python3 <desired-path>  
$source <desired-path>/bin/
```

Refer to the exhibit. An engineer needs to complete the creation and enable a Python virtual environment. Which code needs to be placed on the box where the code is missing?

- A. activate
- B. enable
- C. allow
- D. power

Virtual Environment

- You can code in Python without using virtual environments.
- The minute you update modules or Python itself, you run the risk of breaking your apps
- A virtual environment allows you to lock in the components and modules you use for your app into a single “package.”

Enabling Python VirtualEnv

- Directory Structure
- Usually associated with a Project
- An isolated environment for installing and working with Python Packages

```
$ python3 -m venv myvenv
```

```
$ source myvenv/bin/activate
```

✓ The activation script will modify your prompt.

✓ Inside a virtual environment, your interpreter will always be `python``

`source environment-name/bin/activate`

Complete Your Session Evaluations



Complete a minimum of 4 session surveys and the Overall Event Survey to claim a **Cisco Live T-Shirt**.



Complete your surveys in the **Cisco Live mobile app**.



Continue your education

- Visit the Cisco Stand for related demos
- Book your one-on-one Meet the Expert meeting
- Attend the interactive education with DevNet, Capture the Flag, and Walk-in Labs
- Visit the On-Demand Library for more sessions at www.CiscoLive.com/on-demand



Thank you

CISCO *Live!*

#CiscoLiveAPJC

CISCO *Live!*

GO BEYOND

#CiscoLiveAPJC