



300-635^{Q&As}

Automating and Programming Cisco Data Center Solutions (DCAUTO)

Pass Cisco 300-635 Exam with 100% Guarantee

Free Download Real Questions & Answers **PDF** and **VCE** file from:

<https://www.geekcert.com/300-635.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 HTTP methods are supported by the Cisco Nexus REST API? (Choose two)

- A. PUT
- B. POST
- C. DELETE
- D. UPDATE
- E. CONNECT

Correct Answer: BC

QUESTION 2

Refer to the exhibit.



```
from acitoolkit.acitoolkit import (
    AppProfile, BridgeDomain, Context,
    EPG, Session, Subnet, Tenant
)

def create_tenant():
    session = Session(
        "https://apic", "admin", "ciscopsdt"
    )
    session.login()
    my_tenant = Tenant("DevNet_Tenant")
    my_vrf = Context("DevNet_VRF", my_tenant)
    my_bd = BridgeDomain("DevNet_BD", my_tenant)
    my_bd.add_context(my_vrf)
    my_subnet = Subnet("DevNet_Subnet", my_bd)
    my_subnet.set_scope("public")
    my_subnet.set_addr("10.10.10.1/24")
    my_app = AppProfile("DevNet_App", my_tenant)
    my_epg = EPG("DevNet_EPG", my_app)
    my_epg.add_bd(my_bd)
    session.push_to_apic(
        my_tenant.get_url(),
        my_tenant.get_json())

if __name__ == '__main__':
    create_tenant()
```

Which two actions does this Python code perform with the Cisco ACI? (Choose two.)

- A. It creates a subnet "DevNet_Subnet" inside VRF "DevNet_VRF" located in ACI tenant "DevNet_Tenant" and sets the scope to "private".
- B. It creates a subnet "DevNet_Subnet" inside AppProfile "DevNet_App" located in ACI tenant "DevNet_Tenant" and sets the network address to "10.10.10.1/24".
- C. It creates an EPG "DevNet_EPG" inside AppProfile "DevNet_App" located in ACI tenant "DevNet_Tenant" and link



the EPG with BridgeDomain "DevNet_BD".

D. It creates a subnet "DevNet_Subnet" inside VRF "DevNet_VRF" located in ACI tenant "DevNet_Tenant" and sets the network address to "10.10.10.1/24".

E. It creates an EPG "DevNet_EPG" inside VRF "DevNet_VRF" located in ACI tenant "DevNet_Tenant" and link the EPG with BridgeDomain "DevNet_BD".

Correct Answer: CE

QUESTION 3

Which two benefits of using network configuration tools such as Ansible and Puppet to automate data center platforms are valid? (Choose two.)

- A. consistency of systems configuration
- B. automation of repetitive tasks
- C. ability to create device and interface groups
- D. ability to add VLANs and routes per device
- E. removal of network protocols such as Spanning Tree

Correct Answer: AB

QUESTION 4

Refer to the exhibit.

**Switch configuration**

```
!Command: show running-config
!
feature hsrp
!
ip access-list allow_http_traffic
 10 permit tcp any any eq www
!
vrf context management
 ip route 0.0.0.0/0 192.168.151.2
!
interface mgmt0
 ip address 192.168.251.129 255.255.255.0
 vrf member management
```

Ansible playbook

```
---
- name: Vlan Provisioning
  hosts: nxos
  gather_facts: no

  vars:
    nxos_provider:
      username: "{{ un }}"
      password: "{{ pwd }}"
      transport: nxapi
      host: "{{ inventory_hostname }}"

  tasks:

    - name: CREATE VLANS AND ASSIGN A NAME, USING VLAN_ID
      nxos_vlan:
        vlan_id: "{{ item.vlan_id }}"
        name: "{{ item.name }}"
        provider: "{{ nxos_provider }}"
      with_items:
        - vlan_id: 2
          name: Native
        - vlan_id: 15
          name: Web
        - vlan_id: 20
          name: App
        - vlan_id: 30
          name: DB
```

Playbook output

```
$ ansible-playbook playbook.yml
```

```
PLAY [Vlan Provisioning] *****
*****

TASK [CREATE VLANS AND ASSIGN A NAME, USING VLAN_ID] *****
*****
failed: [192.168.252.129] (item={'vlan_id': 2, 'name': 'Native'}) => {"ansible_facts": {'discovered_interpreter_python': "/usr/bin/python"}, "ansible_loop_var": "item", 'changed': false, 'item': 'name': 'Native', "vlan_id": 2}, "msg": "Request failed: <urlopen error [Errno 61] Connection 'refused'>" "status": -1, "url": "http://192.168.251.129:80/ins"}
```

The exhibit shows a Cisco NX-OS switch configuration, an Ansible playbook, and the output of running this playbook. The playbook failed due to error "msg\\' \\Request failed \\', \\status\\' -1, \"url\" \"http://192.168.251.129:80/ins\"".

Which Cisco NX-OS configuration command resolves this failure?

- A. feature nxapi
- B. http-server enabled
- C. interface mgmt0; ip access-group allow_http_traffic in
- D. feature http

Correct Answer: C

Reference: https://www.cisco.com/c/en/us/td/docs/switches/datacenter/nexus5000/sw/configuration/nxos/41_3/b_Copy_of_b_Cisco_Nexus_5000_Series_NXOS_Software_Configuration_Guide/Copy_of_b_Cisco_Nexus_5000_Series_NX-



OS_Software_Configuration_Guide_chapter22.pdf

QUESTION 5

Refer to the exhibit.

```
from ucsmsdk.ucshandle import UcsHandle

handle = UcsHandle(hostname, username, password, port=80)
handle.login()
for blade in handle.query_classid("computeBlade",
                                filter_str= ):
    print(blade)
handle.logout()
```

When Cisco UCS is configured, which filter must be added to the str_filter string to retrieve all discovered compute blades with two CPUs?

- A. (num_of_cpus eq 2) and (discovery eq complete)
- B. (num_of_cpus. \\x2\ type-sum") and (discovery, \\complete1, type=\\'sum\\')
- C. (num_of_cpus eq