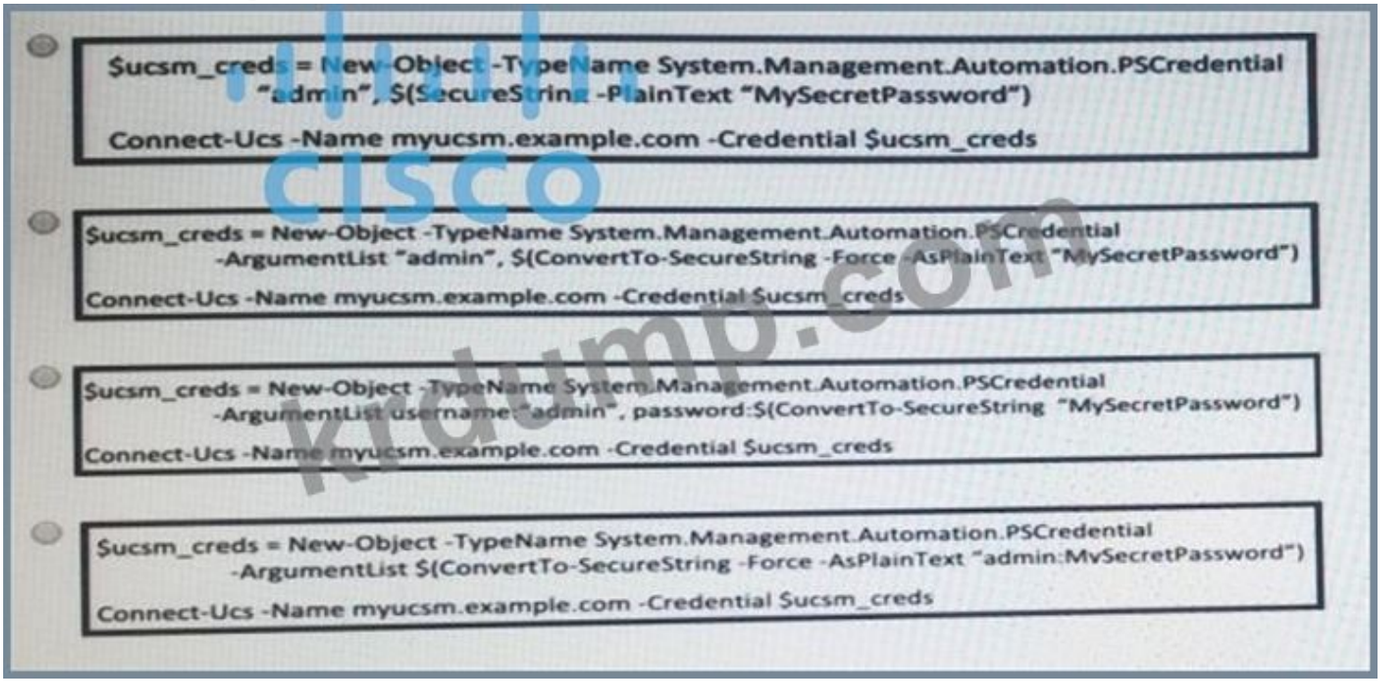
Cisco.300-635.v2022-02-03.q70

□□□□:	300-635
□□□□:	Automating Cisco Data Center Solutions (DCAUTO)
□□□:	Cisco
□□ □□ □□□:	70
□□:	v2022-02-03
# □□ □:	934
# □□ □□□:	700
https://www.krdump.com/Cisco.300-635.v2022-02-03.q70.html	

NEW QUESTION: 1
Kubernetes □□□ Cisco ACI□ □□□ □ □□□ Kubernetes □□□□□□ □□□ □□□□ □ □□?
A. □□□□□□□ ACI EPG □□□ □□□□□ □□□.
B. □□□□□□ □□□ □□□ □□□ □□□□□□ □□□□ □□ □□□ Cisco ACI□□ Kubemetes □□□ □□□□ □□□□.
C. □ □□□ □□□□ □□ □□□ □□□ □□□□.
D. □□□□□ □□□□ Cisco APIC □□ IP□ □□□□□ □□□.
Answer: C (LEAVE A REPLY)

NEW QUESTION: 2
Cisco UCS Manager □□□ □□□□ Cisco UCS PowerTool □□□ □□□□□?

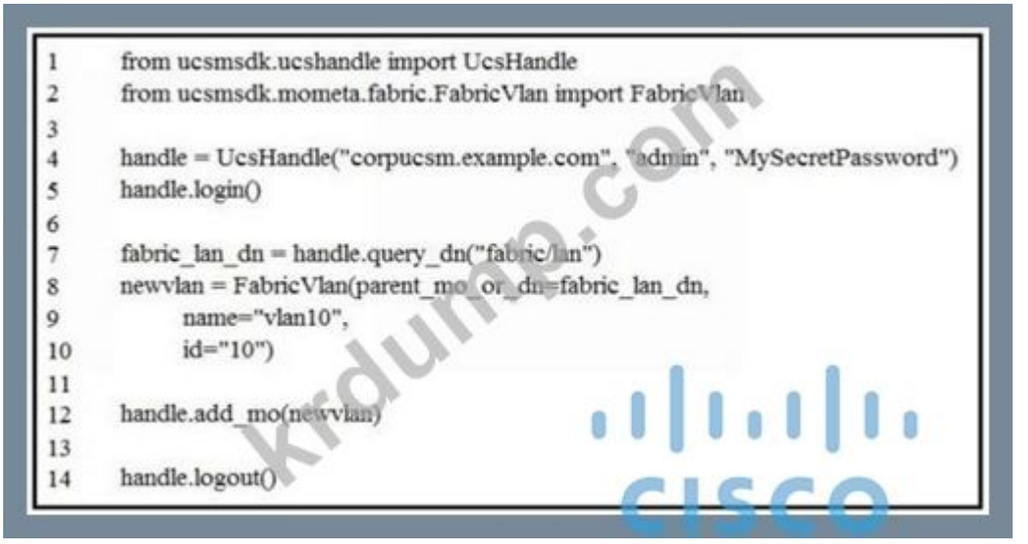


A. □□

B. □□
C. □□
D. □□
Answer: B (LEAVE A REPLY)
□□:
https://blogs.cisco.com/developer/cisco-ucs-automation-part2-a-step-by-step-guide-to-connecting-and-disconnecting-using-ucs-powertool

NEW QUESTION: 3
Cisco UCS Director □□□□ □□□ □□ □□□ □□□□□?
A. □□ □□□□ □□□ Python □□□□□ □□□□ □□
B. □□□ □□ □□□□□ □□□ □□□ □□□ □□ □□□□□ □□□□, jar □ □□□ □□ □ □□□ □□□□ □□
C. Cisco UCS Director□ □□□ □□ □□□□□ □□□□ □
D. □□□ □□ □□□□ □□□ □□□□ □□ □□□ □□□□, Bash □ □□□ □□ Python □ □□ □□□□ □□
Answer: B (LEAVE A REPLY)
□□: □□□ □□ □□□

NEW QUESTION: 4
□□□ □□□□□.



□□□ Cisco UCS□□ VLAN 10□ □□□ □ □□□ □□ □□ □□□ □□□□□?
A. 8□□ 9□□ □□ \□ □□□ □□□.
B. 13□□□ "handle.commit()"□ □□□□□ □□□.
C. 4□□□□ Transport 443 □□□ □□□□□ □□□.
D. 3□□ query_dn□ □□ □□□□□ □□□□ □□□.
Answer: C (LEAVE A REPLY)
□□: □□□ □□ □□□

NEW QUESTION: 5
Cisco NX-OS □□□□□ □□□□ □ □□ □□□□□ □□□ □□□□□? (2□ □□)
A. □□
B. DHCP
C. PNP
D. □
E. iPXE
Answer: B,D (LEAVE A REPLY)

NEW QUESTION: 6
□□□□□ Cisco Nexus 9000 □□□ □□□□ □□□□ □□□□. □□□ □□□□□ □□ Cisco NX-OS□□ Bash□ □□□□□ □□□ □□□□□?
A. bash □□
B. bash □□□
C. bash-shell □□
D. bash-shell □□
Answer: (SHOW ANSWER)

NEW QUESTION: 7
Cisco NX-OS□ □□□□ Cisco Nexus 9500 □□□ □□□□□ □□□□ □□ Docker □□ □□ □□ □□□□ □□□ □□□□□?
A. □□□ □□ □□□ □□□□□ Docker □□□ □□□□□.
B. □□ □□ □ dockerpart □□□ □□□□ □□ □□□□□□ □□□□□.
C. □□□ □□□□ □□ dockerpart □□□ □□ □□□□□□ □□□□ □□□□□.
D. □□□□ □□ □□□ □□□□.
Answer: C (LEAVE A REPLY)
□□: □□□ □□ □□ □□ □□□□

NEW QUESTION: 8
□□ □ □□□ ACI □□□□□ □□□ □□ □□ □□□□ Ansible □□□□ □□□ □□□□□?

A. □□

B. □□

C. □□

D. □□

Answer: C (LEAVE A REPLY)

NEW QUESTION: 9

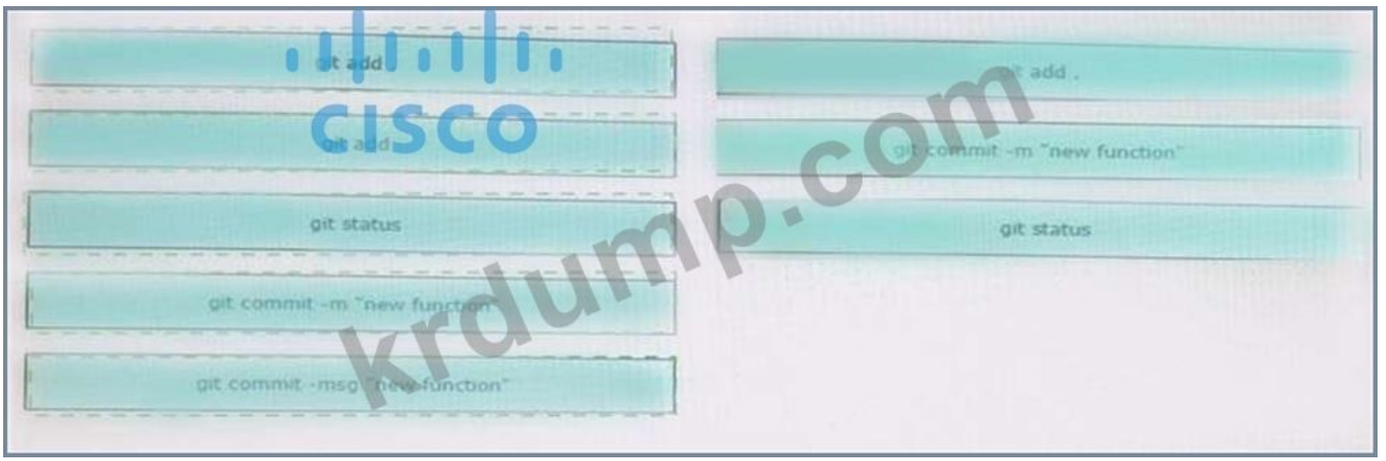
□□ □ Cisco Intersight REST API □□□ □□ □□□□ □□ □□□ □□□□ □ □□ □□□ □ □□□□? (2□□ □□□□□.)

A. □□□□ REST API □□□ RSA □□ □□ □□ □□□□ Cisco Intersight Web □□□□ □□ □□□□□.
B. REST API □□□□ □□□ □□□ □ □□□ base64 □□□ □□□ □□□□ □□□□.
C. □□□ □□□ □□ □□ □□ □□ □□□ □□□ SHA256 □□□ □□□□ □□ API □ ID□ □□□□□.
D. REST API □□ □□□ □□□ SHA384 □□□ □□□□ □□ API □ ID□ □□□□□.
E. Cisco Intersight □ □□□□ API □ ID□ □□ RSA □□ □□ □□□□ □□□ □□□ □□□ □□.
Answer: (SHOW ANSWER)

NEW QUESTION: 10
ACI MIT □□ □□□ □□ □ □□□ □□□□□? (2□□ □□□□□.)
A. URL
B. DN
C. □
D. RN
E. NI
Answer: B,D (LEAVE A REPLY)

NEW QUESTION: 11
myfunc.py□□ □□□ □□□□□□□. □ □□□ □□ Git □□□□□□ □□□□□□ □□□□ □□□□ □□□ □□□ □□□ □□□□ □□□ □□□□ □□□ □□□□. □□ □□□ □□□ □ □□ □□□□.

Answer:



□□

□□ □□ .

git commit -m "□ □□"

□□ □□

NEW QUESTION: 12

GitHub□ Cisco bigmuddy-network-telemetry-collector□ □□□ □ □□□ □□□ □□□□ □□ □ □□□□□?

A. --print-all

B. --□□

C. --□□

D. --□□□

Answer: D (LEAVE A REPLY)

NEW QUESTION: 13

show install all impact kickstart bootflash:n5000-uk9- kickstart.7.0.7.N.1.1.bin system bootflash:n5000-uk9.7.0.7.N1.1a.bin □□□ □□□□ □□□ □ □□ □□ □□□□□?

A. □□□ Cisco NX-OS □□□□ □□ □□□□□□ □□□□□ □□

B. □□□ Cisco NX-OS □□□□ □□ ISSU□ □□□□□ □□

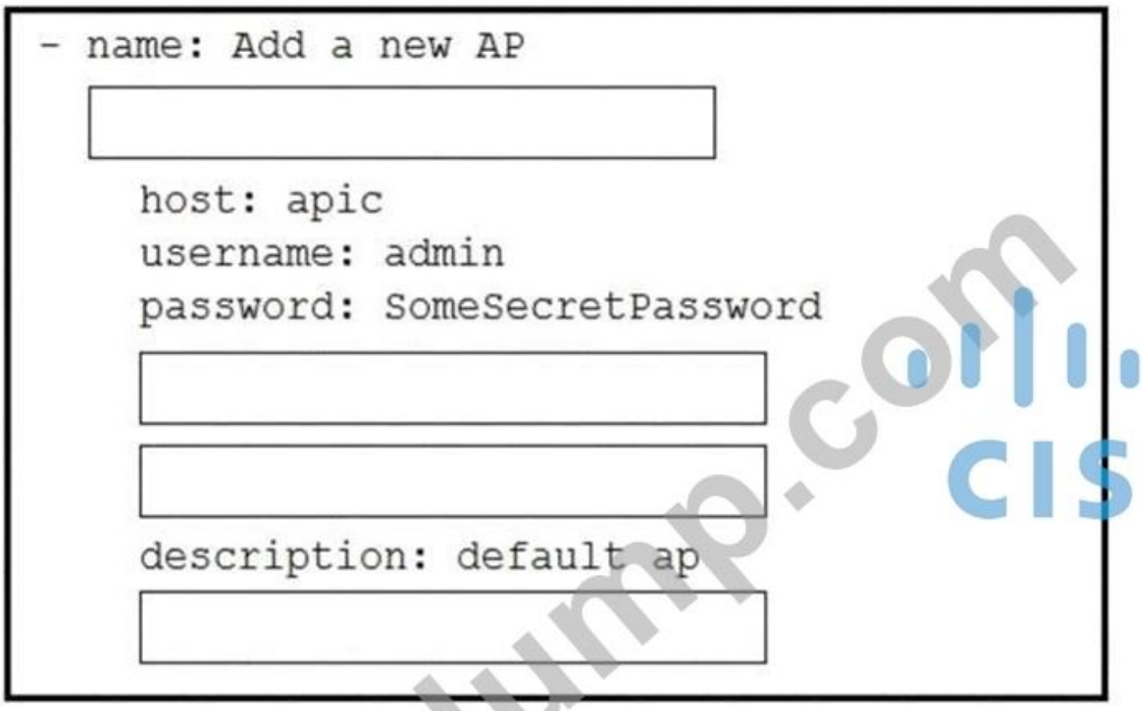
C. □□□ □□□□ □□□□ □□ □□□ □□□□□ □□

D. □□□ □□□ □□□□ □□□□ □□□□ □□□□□ □□

Answer: (SHOW ANSWER)

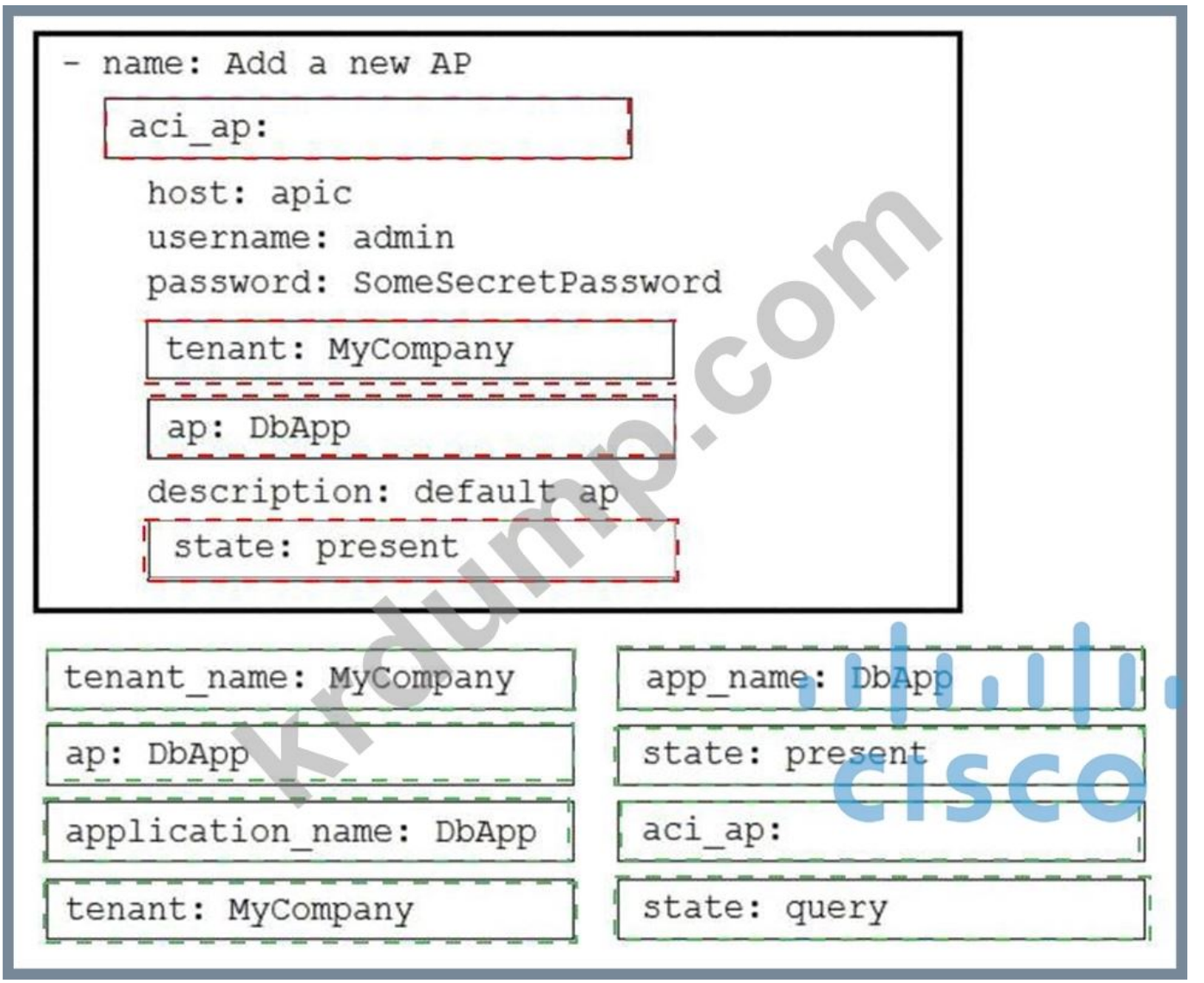
NEW QUESTION: 14

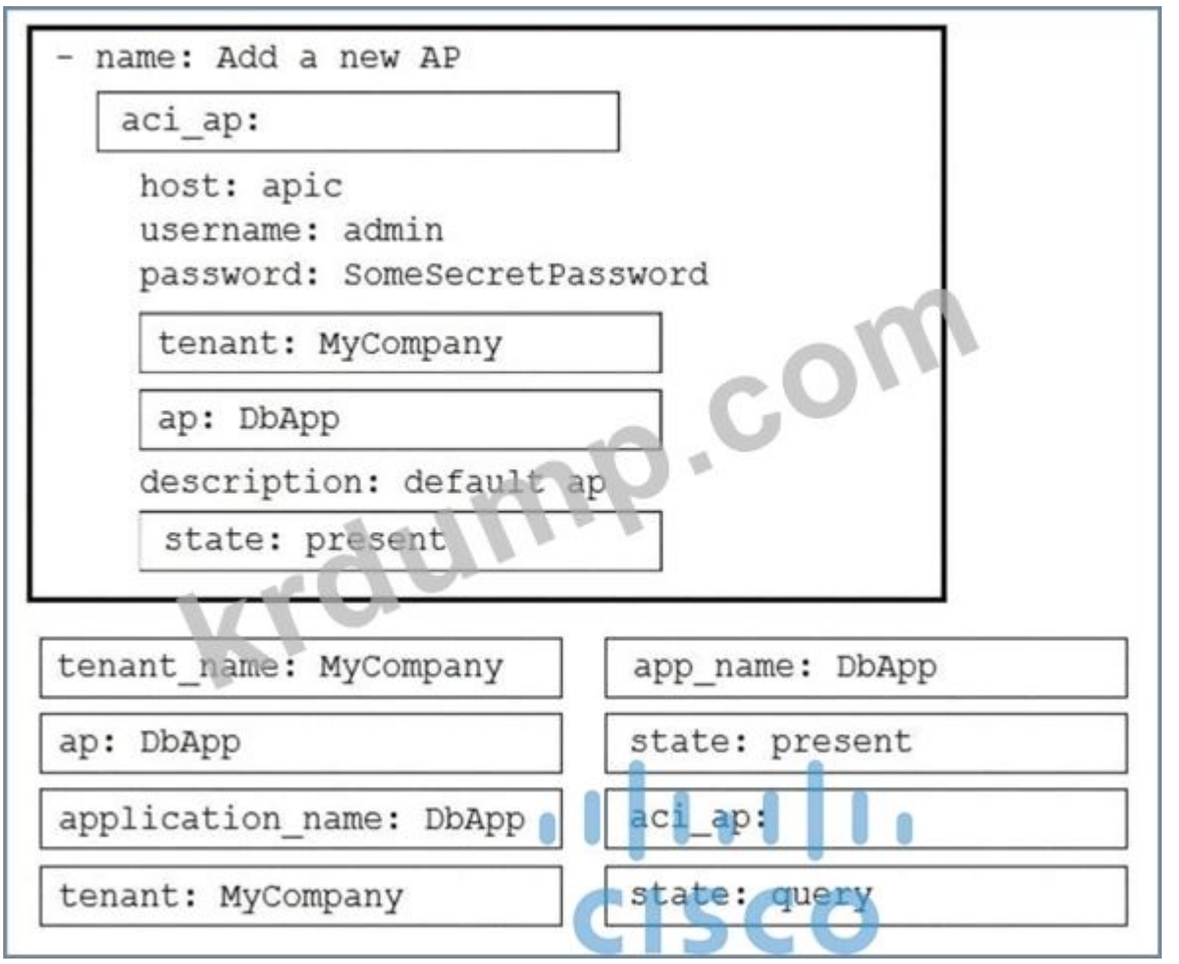
Ansible ACI □□□ □□□□ "DbApp"□□□ □ □□□□□□ □□□□ □□□□□ □□□□ □ □□ YAML □□ □□□ Ansible □□□□ □□ □□□ □□□□ □□□ □□□□. □□ □□□ □ □□□ □□ □□□□.



tenant_name: MyCompany	app_name: DbApp
ap: DbApp	state: present
application_name: DbApp	aci_ap:
tenant: MyCompany	state: query

Answer:


- name: Add a new AP
aci_ap:
host: apic
username: admin
password: SomeSecretPassword
tenant: MyCompany
ap: DbApp
description: default ap
state: present
tenant_name: MyCompany
ap: DbApp
application_name: DbApp
tenant: MyCompany
app_name: DbApp
state: present
aci_ap:
state: query



NEW QUESTION: 15
DCNM REST API□ □ □□ □□□ □□□□□? (2□□ □□□□□.)
A. □□ □□ □□ □ □□ □□□ □□ dcnm-token □□□ □□□□□.
B. □□□ □□□ □□□□□ □□□□□ □□ □□ □□□ □□□□□.
C. □□□ 11.0(1)□□ HTTP □□
D. □□□ □ □□ □□
E. Classic LAN, LAN Fabric, Media Controller □ SAN □□ □□□ □□□□□.
Answer: A,E (LEAVE A REPLY)
□□: □□□ □□ □□□

NEW QUESTION: 16
CUCM□ □□□ □□ □□□ □□ □□□ □□□ □□□ □ □□□□?
A. CISCO TAC□□□ OS□ □□□□ □ □□□□.
B. □□
C. CLI
D. GUI □ CLI
E. GUI
Answer: D (LEAVE A REPLY)

NEW QUESTION: 17
gRPC□ □□ □ □□ □□□□ □□ □□? (2□ □□)
A. IETF □□□□□.
B. HTTPS□ □□ □□□□□.
C. □□ □□ □□□□□□□□.
D. IETF □□□□□.
E. SSH□ □□ □□
Answer: B,D (LEAVE A REPLY)

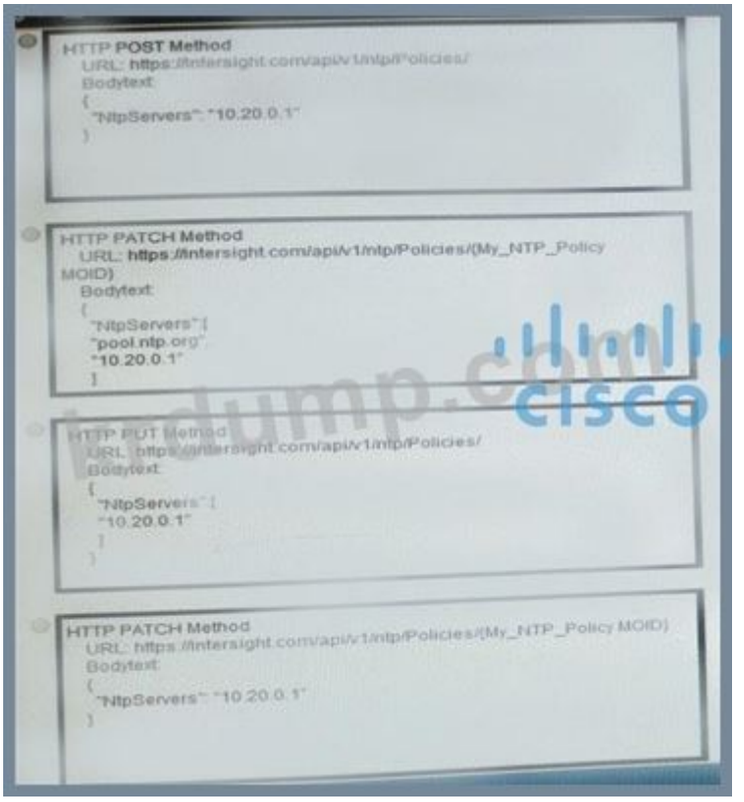
NEW QUESTION: 18
Python □□□□ □□ □□ □□□□ □□□□ □□□□ GitHub□□ Cisco bigmuddy-network-telemetry-collector□ □□□□ □□□ □□□□□?
A. □□ □□ □□ □□
B. □□□□ □□ □□
C. telemetry_receiver.py --destination <□□> -uri <url>
D. telemetry_receiver.py --ip-address <addr> -port <port>
Answer: D (LEAVE A REPLY)

NEW QUESTION: 19
Cisco Nexus 7000□□ □□□□ □□□ □ □□ 128□□ □□ □□□ □□ □□ □□ 6 □□□□ □□□□ □ □□□ □□ □□ □□□ □□□□ □□□ □□ 6 □□□□ □□□ □□□□ □□□ □□□□□?
A. □□□# □□□ □□□□ □□□
B. □□□# □□□ □□ type6
C. □□□# □ □□ □ ASCII
D. switch(config)# □□ □□ □□□ aes
Answer: A (LEAVE A REPLY)
□□/□□: http://www.cisco.com/c/en/us/td/docs/switches/datacenter/sw/6_x/nx-os/security/configuration/guide/b_Cisco_Nexus_7000_NX-OS_Security_Configuration_Guide__Release_6-x/ b_Cisco-NX_00 OS_Security_Configuration_Guide__Release_6-x_chapter_010101.html

NEW QUESTION: 20

□□□□:

□□□ □□□□□. Cisco Instersight□□ □□ NTP □□ □ □□ "pool.ntp"□ □□□□ My_NTP_Policy□□ NTP □□ □□□ □□□□ □□□□. My_NTP_Policy □□ □□□ □□ NTP □□(10.20.0.1)□ □□□□ Cisco Intersight API □□□ □□□□□?



A. □□

B. □□

C. □□

D. □□

Answer: C (LEAVE A REPLY)

NEW QUESTION: 21

SNMP□ □□ □□ □□ □□□ □ □□ □□□□ □□□□□? (2□□ □□□□□.)

A. □□ □□ □□ □□□ □□ □□□ □□□□□.

B. SNMP□ □□ □□□ □□□□□.

C. SNMP□ □□ □□□ □□□ □□□□□.

D. □□ □□ □□ □□□ □ □□□ □□□□□.

E. SNMP□ □ □□□ □□□□□.

Answer: (SHOW ANSWER)

NEW QUESTION: 22

Ansible □□□ □□□□ Cisco ACI□□ 3□□ EPG□ □□□□ □□□ □□□□□? □□□□ □ □ □□ □□ aci_epg □□ □□□ □□□□□□□.

A)

```
aci_epg:
   epg: "{{ item.epg }}"
 loop:
   - item epg:epg1
   - item epg:epg2
   - item epg:epg3
```

B)

```
aci_epg:
   epg: "{{ item.epg }}"
 loop:
   - epg:epg1
   - epg:epg2
   - epg:epg3
```

C)

```
aci_epg:
   epg: "{{ item.epg }}"
 loop:
   item: "{{ epg1 }}"
   item: "{{ epg2 }}"
   item: "{{ epg3 }}"
```

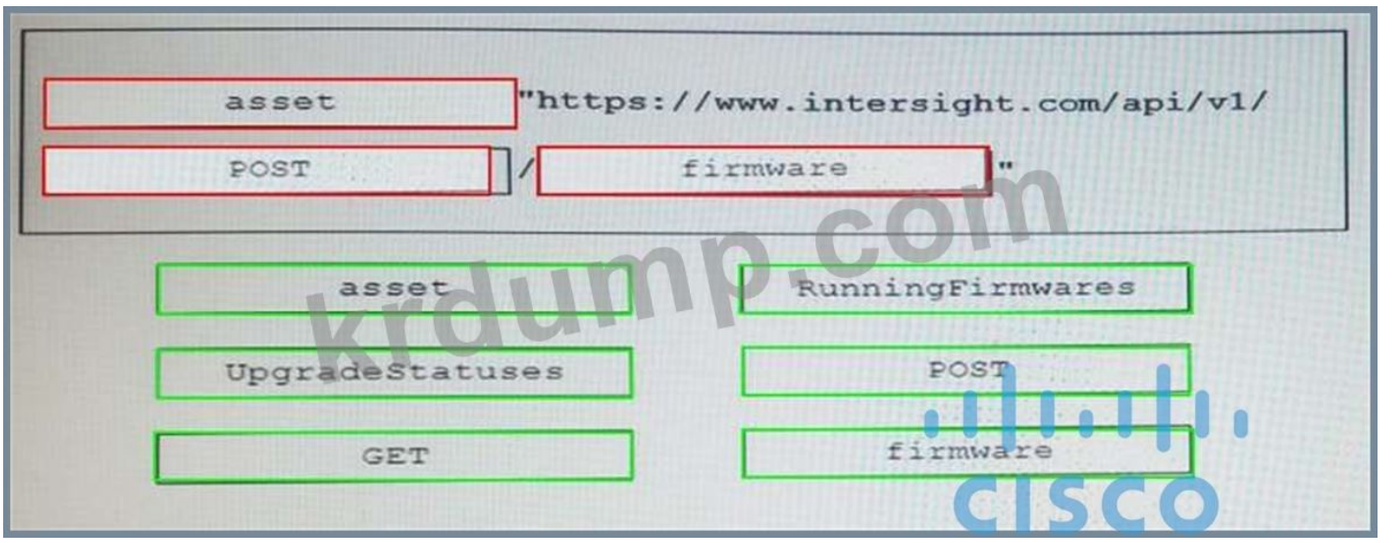
D)

```
aci_epg:
  epg: "{{ item.epg }}"
loop:
  epg: "{{ epg1 }}"
  epg: "{{ epg2 }}"
  epg: "{{ epg3 }}"
```

A. □□
B. □□
C. □□
D. □□
Answer: C (LEAVE A REPLY)

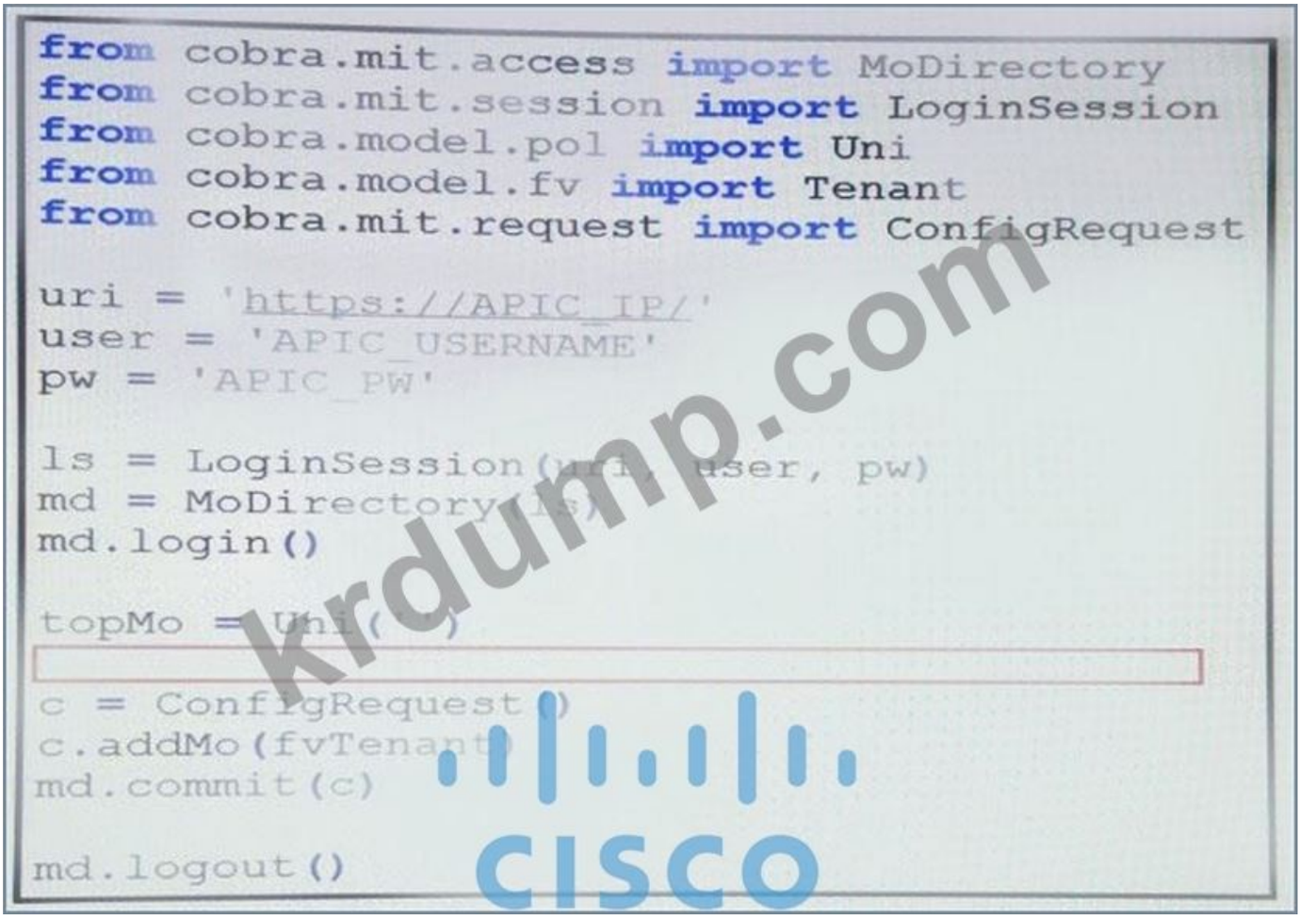
NEW QUESTION: 23
□□□ □□□ □□ Cisco intersight API□□ Cisco UCS □□□ □□ □□□□ □□□□□ □□□ □□□□□. □□ □□□ □□□□ □□ □□□□.

Answer:



NEW QUESTION: 24

□□□ □□:



□□□ □□: □□□ APIC □□□□□□ □ □□□□□ □□□ □ □□□□ Cobra SDK□ □□ Cisco□□ □ □□□□ □□□□ □□□.

□ □□□□ □□□□ □□□ □□□ □□ □□□ □□□□ □□□?

A. □□□ = □□□(topMo, name='Cisco')

B. □□□ = NewTenant(□□='Cisco')

C. □□□ - □□□(topMo, name-Cisco')

D. □□□ = □□□('Cisco')

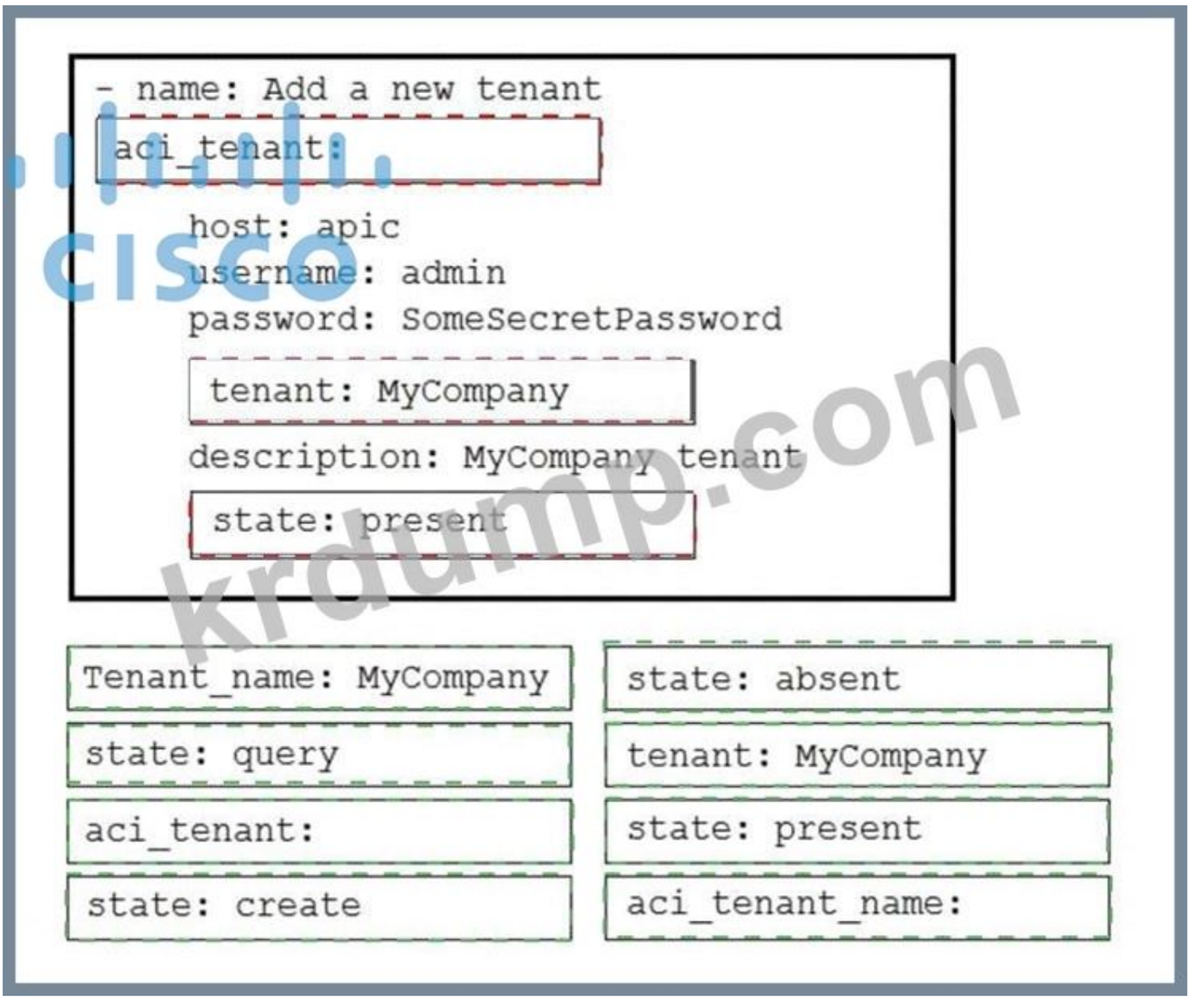
Answer: A (LEAVE A REPLY)

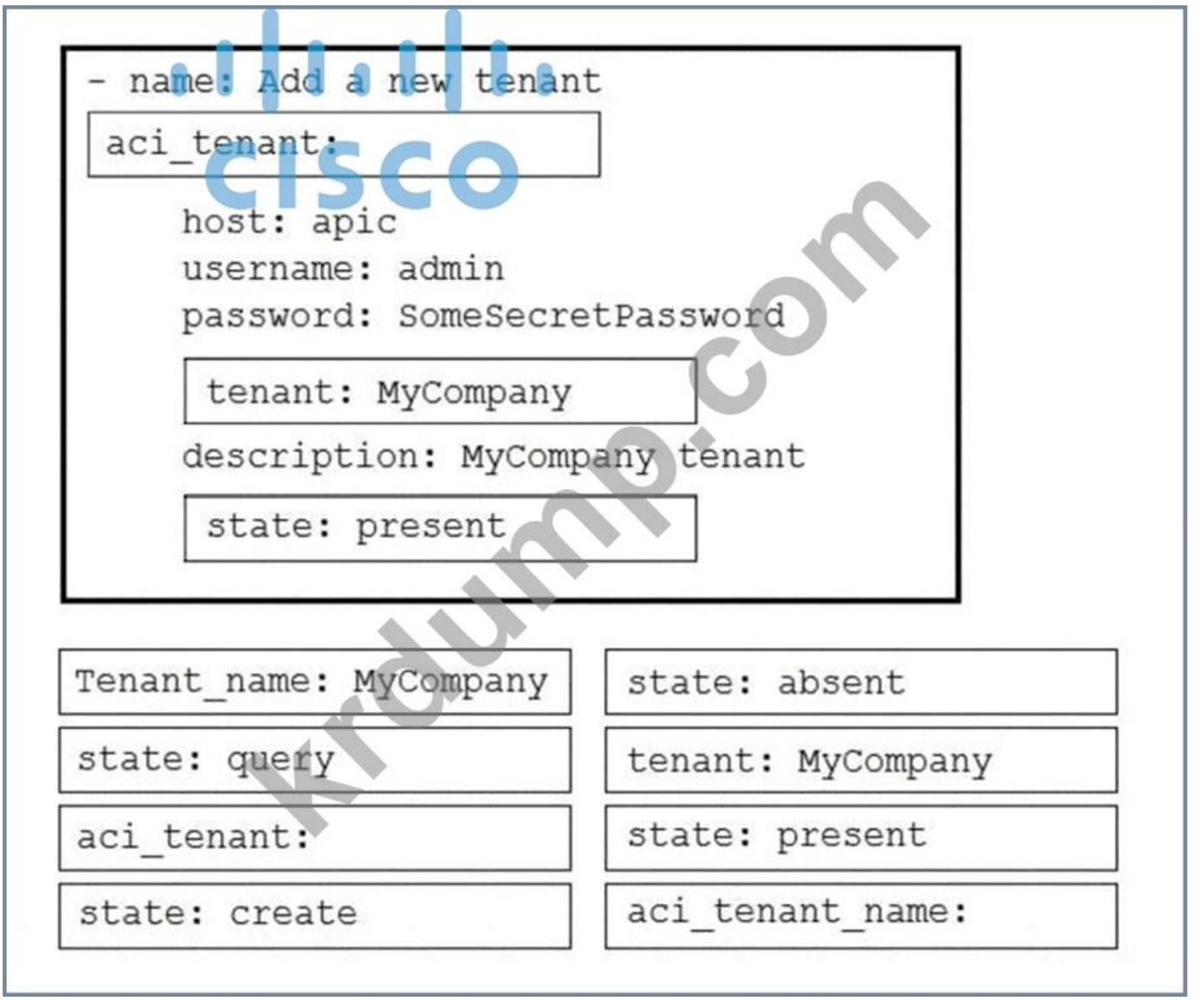
NEW QUESTION: 25

□□□ □□□ □□ □ □□□□ □□□□ Ansible □□□□□ □□□□□. □□ □□□ □□□□ □□ □□□□.

Tenant_name: MyCompany	state: absent
state: query	tenant: MyCompany
aci_tenant:	state: present
state: create	aci_tenant_name:

Answer:


- name: Add a new tenant
aci_tenant:
host: apic
username: admin
password: SomeSecretPassword
tenant: MyCompany
description: MyCompany tenant
state: present
Tenant_name: MyCompany
state: query
aci_tenant:
state: create
state: absent
tenant: MyCompany
state: present
aci_tenant_name:



* aci_tenant
* □□□ : □□□□□
* □□: □□

NEW QUESTION: 26

NX-API CLI JSON-RPC □□□□□□ □□□□ SVI□ □□□□ □ □□ Python □□□ □□ □ □□ □□□ □□□□□?

(2□□ □□□□□.)

```
requests.post(url, data=json.dumps(payload), headers={'content-type':
'application/json-rpc'},auth=(username, password))
```

```
requests.post(url, data=json.dumps(payload), headers={'content-type':
'application/json'},auth=(username, password))
```

```
payload = {
      "jsonrpc": "2.0", "method": "cli_conf",
      "params": {
        "command": "conf t; interface vlan " + id,
        "version": 1},
      "id": 1
}
```

```
payload = [
    {
      "jsonrpc": "2.0", "method": "cli",
      "params": {"cmd": "interface vlan " + id, "version": 1},
      "id": 1
    },
    {
      "jsonrpc": "2.0", "method": "cli",
      "params": {"cmd": "interface vlan " + id, "version": 1},
      "id": 2
    }
]
```

```
payload = {
      "jsonrpc": "2.0", "method": "cli_conf",
      "params": {"cmd": "interface vlan " + id, "version": 1},
      "id": 1
}
```


A. ☐☐

B. ☐☐

C. ☐☐

D. ☐☐

Answer: A,C (LEAVE A REPLY)

NEW QUESTION: 27

☐☐☐ ☐☐:

```
switch(config)# telemetry
switch(config-telemetry)# sensor-group 100
switch(config-tm-sensor)# path sys/intf/phys-[eth1/1] depth 0
switch(config-tm-sensor)# destination-group 100
switch(config-tm-dest)# ip address 1.2.3.4 port 50004
switch(config-tm-dest)# ip address 1.2.3.4 port 50005
switch(config-tm-sensor)# destination-group 200
switch(config-tm-dest)# ip address 5.6.7.8 port 50001 protocol HTTP encoding JSON
switch(config-tm-dest)# ip address 1.4.8.2 port 50003
switch(config-tm-dest)# subscription 100
switch(config-tm-sub)# snsr-grp 100 sample-interval 10000
switch(config-tm-sub)# dst-grp 100
switch(config-tm-sub)# dst-grp 200
```

□□□□ □□□□□□. □□□ □□ 1/1□ □□ □□ □□□□ □□□□ □□□□ □□□ □□ □ □□□□?

A. 10□□□ 4□□ □□ □□□□

B. 10000□□□ 4□□ □□ □□□□

C. 10000□□□□□□□ 4□□ □□ □□□□

D. 100□□□□□ 4□□ □□ □□□□

Answer: C (LEAVE A REPLY)

NEW QUESTION: 28

Switch(config)# □□□ □□ - 60□ □□ □ 180□ □□ 10□

Switch(config)# □□□ □□ □□ □□□ □□□ acl

□□□ □□□□□. □ □□□ □□□□ Cisco NX-OS □□□□ □□□ □□□ □□ □□□ □□ □□□□ □□ □□ □□?

A. 180□ □□□ □□□ □□□ 10□ □□□□ ACL□ □□□□ □□□□□.

B. □□□ □□□ 10□ □□ □□□□ ACL □□ □□□□ □□□□□.

C. ACL□ □□□□ 180□ □□□ □□□ □□□ 10□ □□□□ □□□□□.

D. 180□ □□ ACL□ □□□□□ □□□ □□□ 10□ □□□□ □□ □□□□ □□□□□.

Answer: C (LEAVE A REPLY)

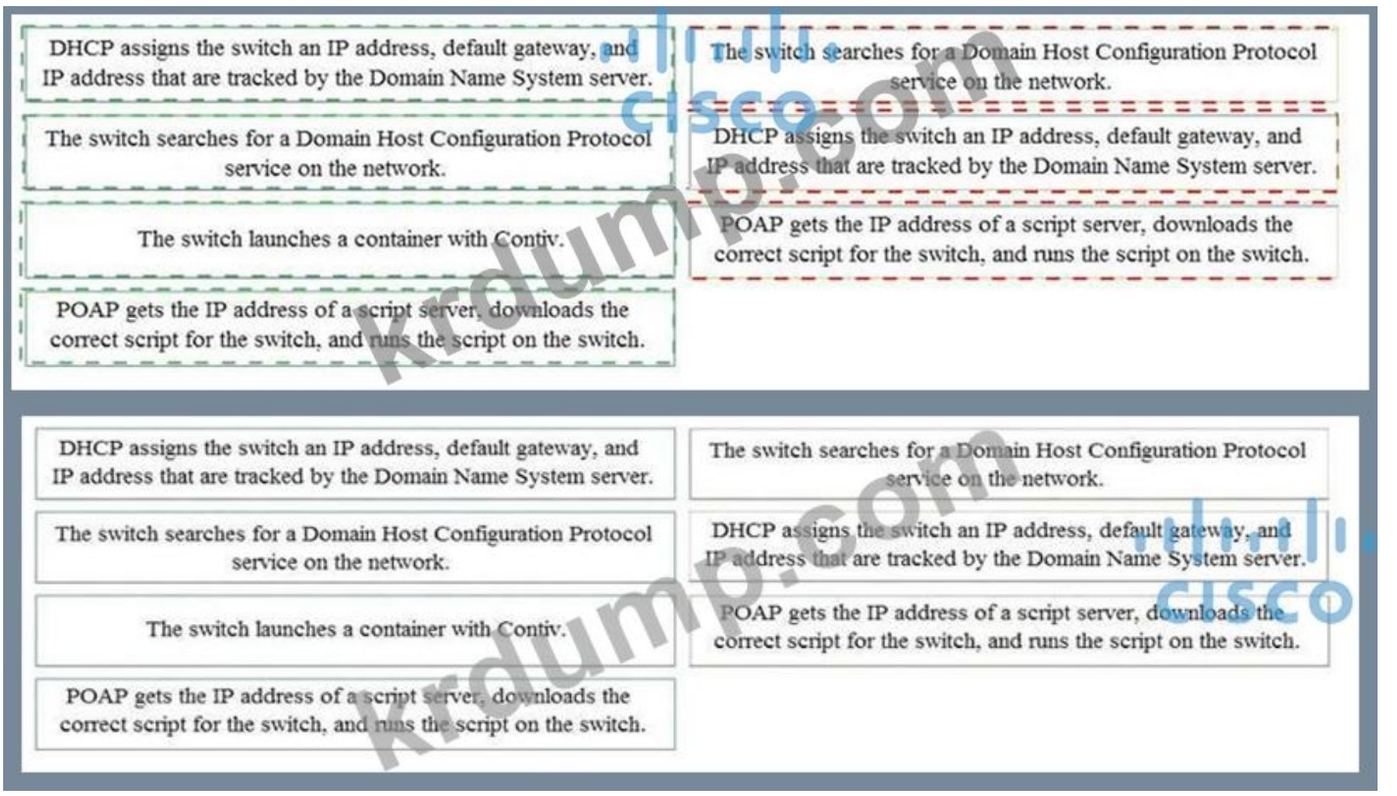
"loginquiet-mode access-class acl "□□□ □□□ □□□ □□ □□□ □□ □□□ □□□ □□□ □ □□□ □□□□□ □ □□□ □ □ □□□□. □□□ acl □□□□ □□□□ □□ □□□ □□ □□ □ □□ □□□□ cisco □□□ □□□□ □ □□□□.

NEW QUESTION: 29

□□□□ □□□ □ startup-config □□□ □□ □□□□. □□ □□□ □□□□ □□□□ □□ Power-On Auto Provisioning□ □□□ □□□ □□□□ □□□□ □□□ □□□ □□□ □□□□. □□ □□□ □□□□ □□ □□□□.

DHCP assigns the switch an IP address, default gateway, and IP address that are tracked by the Domain Name System server.	step 1
The switch searches for a Domain Host Configuration Protocol service on the network.	step 2
The switch launches a container with Contiv.	step 3
POAP gets the IP address of a script server, downloads the correct script for the switch, and runs the script on the switch.	

Answer:



NEW QUESTION: 30

□□ □□

```
mo_dir = cobra.mit.access.MoDirectory(cobra.mit.session.LoginSession(apic_url, username, password))
mo_dir.login()
cq = cobra.mit.access.ClassQuery('fvCEp')
cq.subtree = 'full'
objlist = mo_dir.query(cq)
for mo in objlist:
  print "MAC: " + mo.mac + " " + "IP: " + mo.ip
```

□ ACI Cobra Python □□□ □□□ □□ □□□ □□□□□?

A. □□ LLDP □□ MAC □ IP □□□ □□□□□.

B. □□ Cisco Discovery Protocol □□ MAC □ IP □□□ □□□□□.

C. □□ □□□□□ MAC □ IP □□□ □□□□□.

D. □□ APIC MAC □ IP □□□ □□□□□.

Answer: C (LEAVE A REPLY)

□□:

https://www.cisco.com/c/en/us/td/docs/switches/datacenter/aci/apic/sw/1-x/Operating_ACI/guide/b_Cisco_Operating_ACI/b_Cisco_Operating_ACI_appendix_011.html

NEW QUESTION: 31

DCNM API□ □□□□ □ □□ □□□ □□□□□? (2□ □□)

A. DCNM□ □□ □□□ GUI□ □□ □□□□ □□□.

B. □□ API □□□ DCNM GUI□ □□□□ □□□ □ □□□□.

C. DCNM□□ API□ □□□□□ □□□□□ □□□□□.
D. DCNM□ REST □□ API□ □□□□□.
E. DCNM□ XML □□ SOAP API□ □□□□□.
Answer: D (LEAVE A REPLY)

NEW QUESTION: 32
□□□ □□□□□.

```
1  dnQuery = DnQuery('uni/tn-companyXYZ')
2  dnQuery.subtree = 'parent'
3  tenantMo = moDir.query(dnQuery)
4  defaultBDMo = tenantMo.BD['default']
```

□□ □□ □□□ □□□□□ □□ □□□ □□□□ □□□ □□□□□?
A. 2□□ □□□□□.
B. 2□□ □□□ □□□ □□□□□.
C. 1□□□ dnQuery□ classQuery□ □□□□□.
D. 2□□□ □□□ □□□□ □□□□□.
Answer: D (LEAVE A REPLY)

NEW QUESTION: 33
□□□ □□□□□.

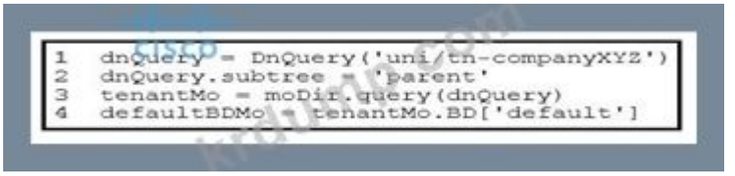
```
https://APIC_IP/api/class/l1PhysIf.xml?query-target-filter=eq(l1PhysIf.speed,"10G")
```

ACI APIC□ □□ □ API GET □□□ □□ □□□□ □□ □□? (2□□ □□□□□.)
A. API □□□ □□ □□□□□ □□□ □□□□.
B. API □□□ JSON□□ □□□□□□.
C. API □□□ XML□ □□□□□□.
D. API □□□ □□ □□□□ □□□ □□□□.
E. API □□□ APIC□□ □□□ 10G □□□□□□ □□□□□.
Answer: (SHOW ANSWER)

NEW QUESTION: 34
gRPC□ □□ □ □□ □□□□ □□ □□? (2□ □□)
A. HTTPS□ □□ □□

B. IETF □□□□□.
C. SSH□ □□ □□
D. IETF □□□□□.
E. □□ □□ □□□□□□□□□.
Answer: (SHOW ANSWER)

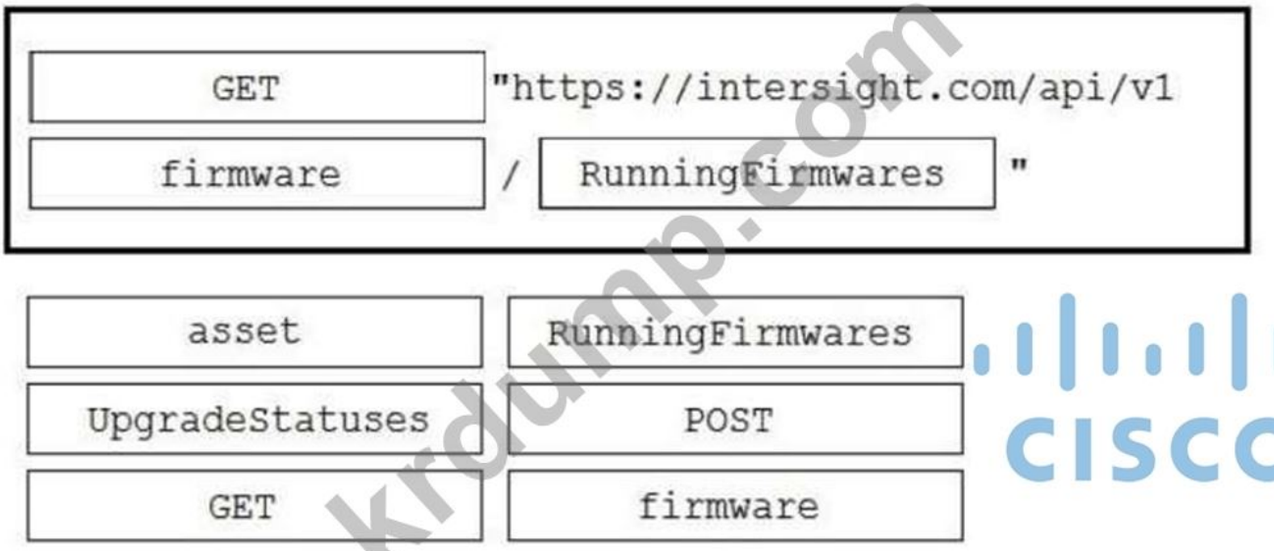
NEW QUESTION: 35
□□□ □□□□□.



□□ □□ □□□ □□□□□ □□ □□□ □□□□ □□□ □□□□□?
A. 2□□ □□□ □□□ □□□□□.
B. 2□□□ □□□ □□□□ □□□□□.
C. 1□□□ dnQuery□ classQuery□ □□□□□.
D. 2□□ □□□□□.
Answer: B (LEAVE A REPLY)

NEW QUESTION: 36
□□□ □ □□
□□□ □□□ □□ Cisco Intersight API□□ Cisco UCS □□□ □□ □□□□ □□□□□ □□□ □□□□□. □□ □□□ □□□□ □□ □□□□.
□□ □ □□:

Answer:



□□: □□□ □□ □□□

□□/□□: https://developer.cisco.com/codeexchange/github/repo/CiscoUcs/intersight-python/

NEW QUESTION: 37

□□□ □□:

□□□□ Cisco NX-OS □□□ □□□ Ansible □□□□ □ □ □□□□ □□□ □□□ □□□□ □. □□□□□ □□ "msg'□ □□ □□□□□□. "□□ □□ <urlopen □□ [Errno 61] □□□ □□ □□□□□.>\ 'status* -1 " url" "http://192.168.251.129:80/ins" □ □□□ □□□□ Cisco NX-OS □□ □□□ □□□□□?

A. □□□□□ mgmt0; ip □□□ □□ allow_http_traffic in

B. nxapi □□

C. □□ http

D. http-□□ □□□

Answer: A (LEAVE A REPLY)

NEW QUESTION: 38
□□□ □□□□□.

```
def add_tenant():
    token = apic_login.aaaLogin()
    for tenant in range (1,10):
        try:
            response = requests.post(
                url=constant.APIC_URL +"/api/node/mo/uni/tn-exam%s.json" % (tenant),
                headers={
                  "Cookie": "APIC-cookie=" + token,
                  "Content-Type": "application/json; charset=utf-8",
                },
                data=json.dumps({
                  "fvTenant": {
                    "attributes": {
                      "status": "created",
                      "dn": "uni/tn-exam%s" % (tenant),
                      "name": "exam%s" % (tenant),
                      "rn": "tn-exam%s" % (tenant)
                    },
                    "children": [
                    ]
                  }
                })
                )
                print('Response HTTP Status Code: {status_code}'.format(
                  status_code=response.status_code))
                print('Response HTTP Response Body: {content}'.format(
                  content=response.content))
            except requests.exceptions.RequestException:
                print('HTTP Request failed')
add_tenant()
```


□ ACI □□□□□ □□□ □ □ □□□□□ □□□ □□ □□□ □□□□□?

A. 10□□ □□□ □□□ □ □□□□□.

B. 9□□ □□□ □□□□□.

C. □□□ □□□□□□.

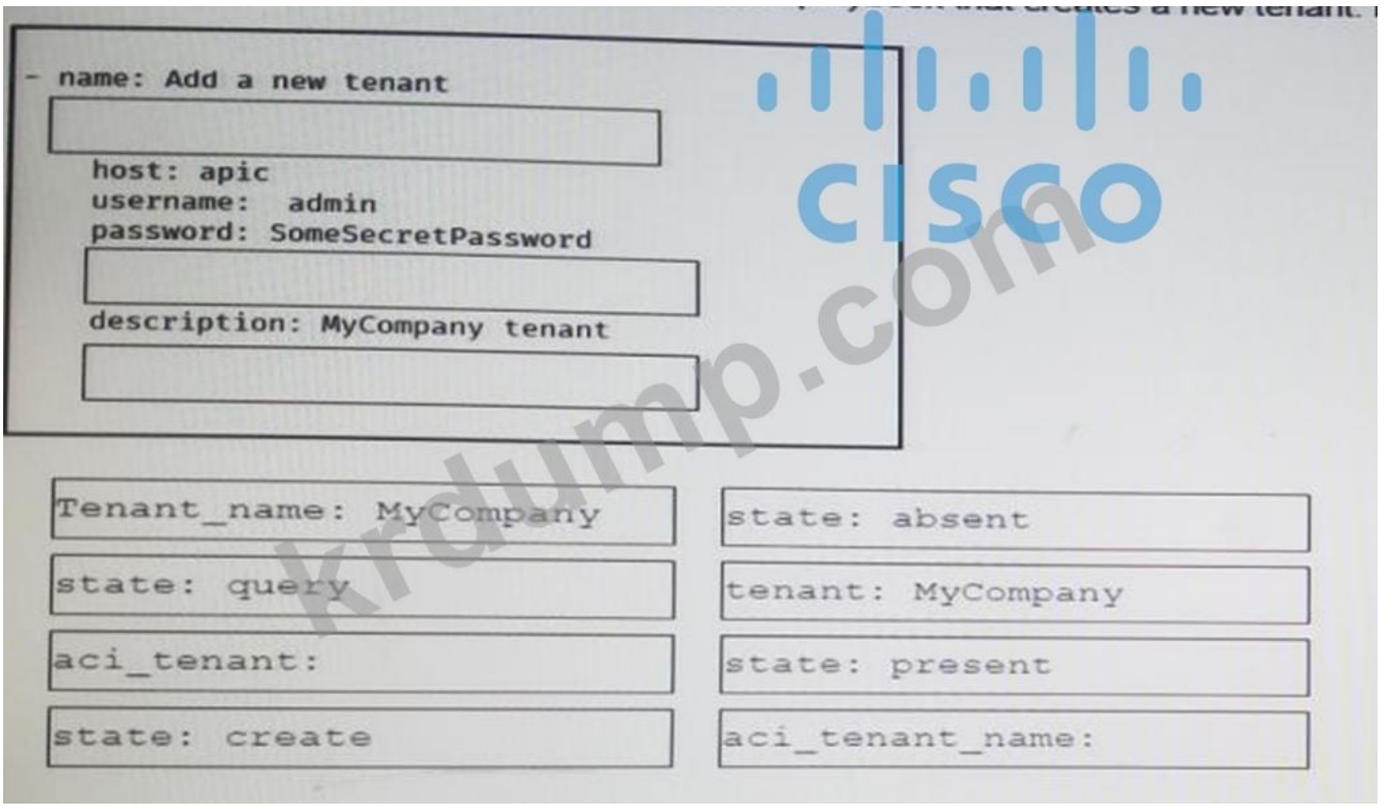
D. 10□□ □□□ □□□□□.

Answer: D (LEAVE A REPLY)
□□: □□□□ □□ □□□ □□ □□□□

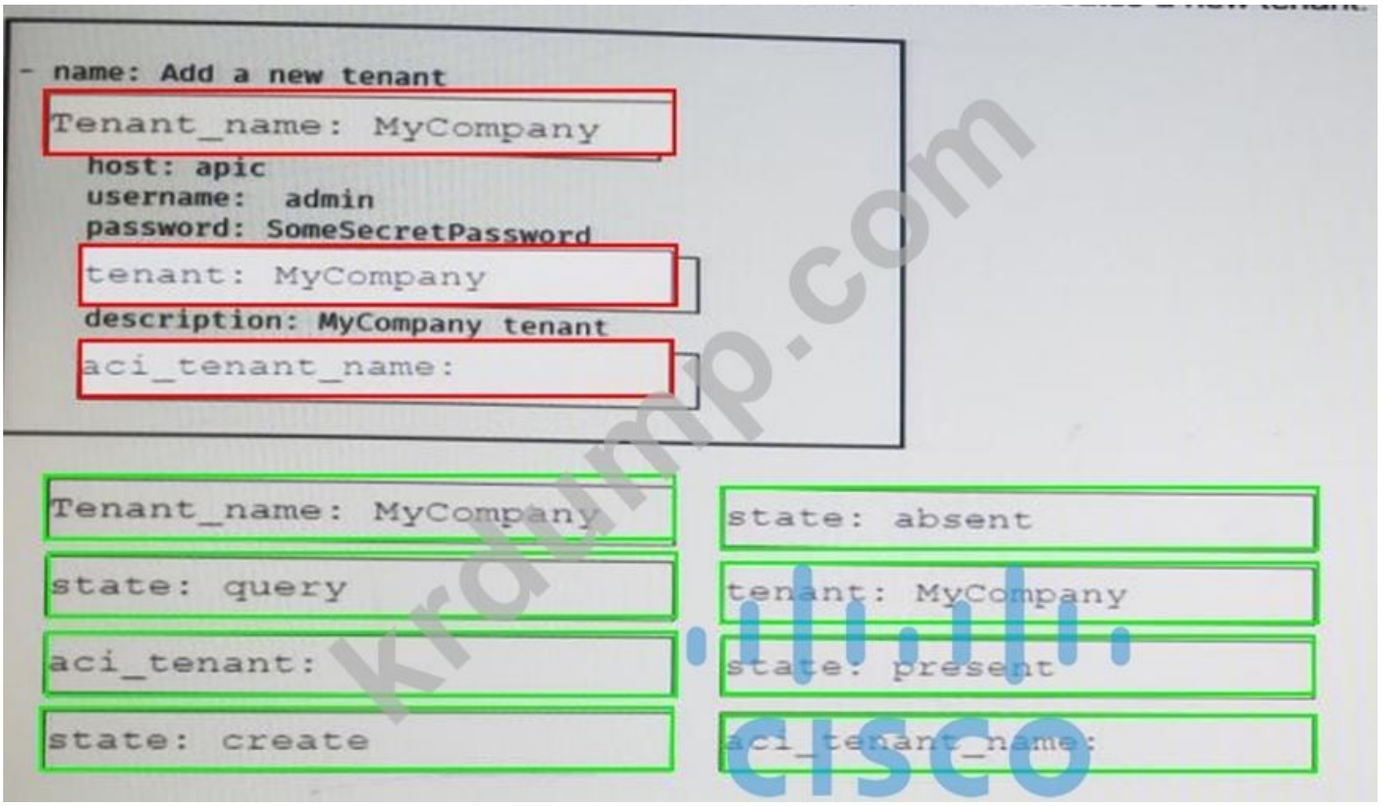
NEW QUESTION: 39
□□□□ Cisco CUCM(Unified Communications Manager)□ Cisco VCS □□□□ □□ □□□ SIP □□□□ □□□□. Cisco VCS□ □□□ Cisco TelePresence System EX60□□ CUCM□ □ □□ Cisco IP Phone 9971□ □□□ □□ □□ □□□□. □□□ □□□ □□□ □□ □ □□□□ □□□□. □ □□□ □□□□□ □□□ □□□□ □□□?
A. SIP □□□ □□.
B. SIP □□□□ □□ □□□□□.
C. Cisco VCS□ CUCM □□□□□.
D. SIP □□□ □ □□ □□ □□.
Answer: D (LEAVE A REPLY)

NEW QUESTION: 40
□□ □□ □□ □□□ □□□ □□□□□?
A. □□□□□□ □□□□□ □□□□ □□□□□.
B. □□□□□□ □□□□ □□□□□□□.
C. □□□□□□ □□□□ □□□□□ □□□□
D. □□□□□ □□□□ □□□□ □□□□
Answer: C (LEAVE A REPLY)

NEW QUESTION: 41
□□□ □□□ □□ □ □□□□ □□□□ Ansible □□□□□ □□□□□. □□ □□□ □□□□ □□ □□□□.



Answer:



NEW QUESTION: 42

□□ □□□ □□ CUCM □□ □□□□ 8851 □□□ □ 8851 □□ □□□□ □□□□□. □□□ □□□□ Cisco 8841□ □□□□□ □□ □□ □□□□ □□□□□?

A. 8841 □□ □□ □□□

B. □□ □□ □□□

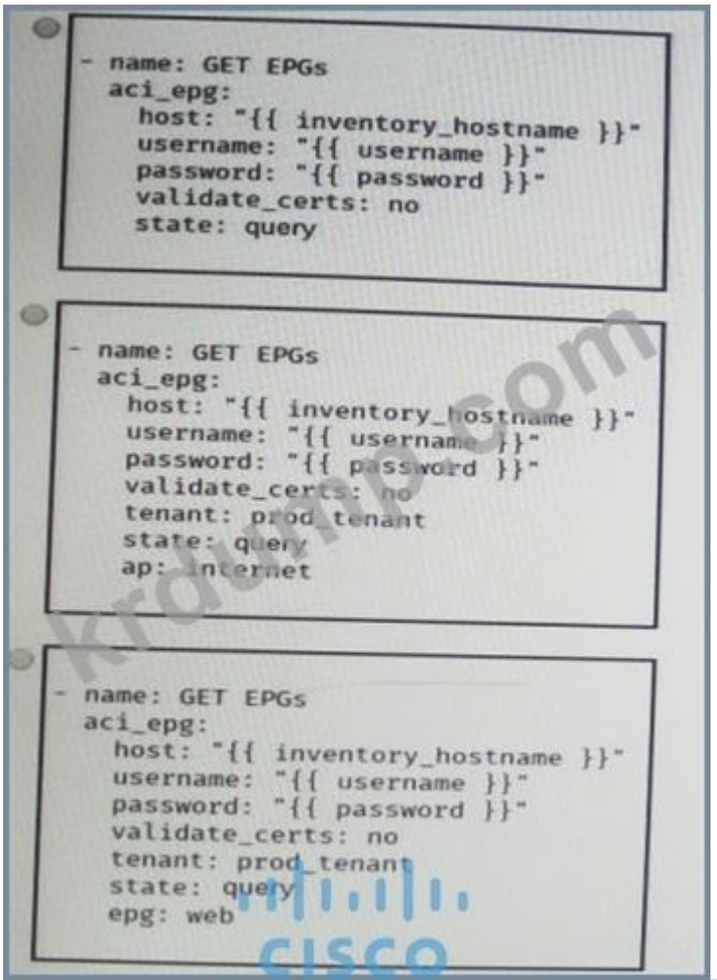
C. 8851 □□ □□□

D. 8851 □□ □□ □□□

Answer: (SHOW ANSWER)

NEW QUESTION: 43

□□ □ □□□ ACI □□□□□ □□□ □□ □□ □□□□ Ansible □□□□ □□□ □□□□□?



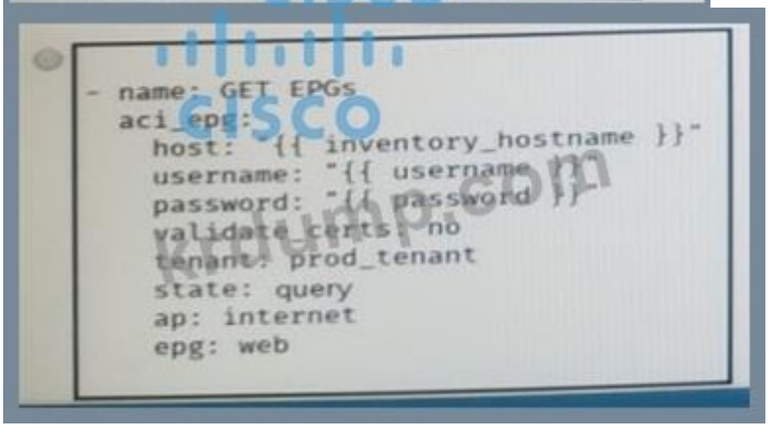
```
- name: GET EPGs
  aci_epg:
    host: "{{ inventory_hostname }}"
    username: "{{ username }}"
    password: "{{ password }}"
    validate_certs: no
    state: query
```



```
- name: GET EPGs
  aci_epg:
    host: "{{ inventory_hostname }}"
    username: "{{ username }}"
    password: "{{ password }}"
    validate_certs: no
    tenant: prod_tenant
    state: query
    ap: internet
```



```
- name: GET EPGs
  aci_epg:
    host: "{{ inventory_hostname }}"
    username: "{{ username }}"
    password: "{{ password }}"
    validate_certs: no
    tenant: prod_tenant
    state: query
    epg: web
```



```
- name: GET EPGs
  aci_epg:
    host: "{{ inventory_hostname }}"
    username: "{{ username }}"
    password: "{{ password }}"
    validate_certs: no
    tenant: prod_tenant
    state: query
    ap: internet
    epg: web
```

A. □□

B. □□

C. □□

D. □□

Answer: (SHOW ANSWER)

NEW QUESTION: 44
Cisco Intersight API□ □□□ □ □□□ □□□□ □ □□ □□□ □□□□□? (2□□ □□□□ □.)
A. cisco.com □□□ □□ □□□ □□ □□□ Cisco Intersight Web Service□ □□□□□.
B. □ API □□ □□ □□□ □□□ □ □□□ □□□ □□□ □ □□□□.
C. Secret Key□ API Key □□ □□□ □□ □□
D. API Key□ Key ID□ Secret Key□ □□
E. API Key□ keyId□ sessionCookie□ □□
Answer: C,D (LEAVE A REPLY)

NEW QUESTION: 45
Ansible □ Puppet□ □□ □□□□ □□ □□□ □□□□ □□□ □□ □□□□ □□□□ □ □□ □ □□ □ □□ □□□ □□□□□? (2□□ □□□□□.)
A. □□□ VLAN □ □□ □□ □□
B. □□□ □□□ □□□
C. Spanning Tree□ □□ □□□□ □□□□ □□
D. □□ □ □□□□□ □□□ □□□□ □□
E. □□ □□□ □□□
Answer: B,E (LEAVE A REPLY)

NEW QUESTION: 46
□□□ □□□□□.

```
from cobra.mit.access import MoDirectory
from cobra.mit.session import LoginSession

uri = 'https://APIC_IP/'
user = 'APIC_USERNAME'
pw = 'APIC_PW'

ls = LoginSession(uri, user, pw)
md = MoDirectory(ls)
md.login()

list = md.lookupByClass('fvAp')

for element in list:
    print(str(element.dn))

md.logout()
```

ACI Cobra SDK□ □□□□ Python □□□□□ □□□□□□□. □ □□□□□ □□□ □□ □□ □ □□□□□?
A. □□ □□□□□□ □□□□ □□ □□□ □□□□□.
B. □□ □□□□ □□ □□□ □□□□□.
C. □□ □□□□□□ □□□□□ □□□□ □□□ □□□□□.
D. □□ □□□□ □□ □□□ □□□□□.
Answer: (SHOW ANSWER)

NEW QUESTION: 47
Cisco UCS Director□ REST API□ □□□□ □ □□□□ □□ □□□ □□□□□?
A. □□□((□□□ □□))
B. HTTP □□ □□
C. RestAuth: ((□□□□ □□ □□))
D. X-Cloupia-Request-Key: ((□□□□ □□ □□))
Answer: D (LEAVE A REPLY)
□□:
https://www.cisco.com/c/en/us/td/docs/unified_computing/ucs/ucs-director/rest-api-cookbook/6-6/cisco-ucs-director-REST-API-cookbook- 66/cisco-ucs-director-REST-API-□□-66_chapter_010.html

NEW QUESTION: 48
Cisco Nexus 7700 □□□ □□□□ □□□□□□ □□□□. □□□□ □□□□ □□ □□ □□□ □□□ □□□□□. □□ □□□ □□□□□ □□□ □ □□ □□□□ □□□□ □□□. □□ □□ □□□ □□□□ □□□?
A. □□
B. □ □□ □□ □ □□
C. □□□ □□□
D. □□□□
Answer: (SHOW ANSWER)
□□/□□: http://www.cisco.com/c/en/us/td/docs/switches/datacenter/sw/5_x/nx-os/system_management/configuration/guide/sm_nx_os_cg/sm_7rollback.html

NEW QUESTION: 49
□□□ □□□□□.

□□□□□ □□□ □□□ □□□□ □□ □□□□□ □□□ □□ □□□ □□ □□□ □□□□ □.

□□□ □□□ □□□□□?

A. □□ □□□□ "□□ □□ □□ CM □□□□ □□"□ □□□□ □□□□□.

B. CUCM □□□ □ □□□ □□□ □□□□□□□.

C. CUCM□ □□□ □□□□ □□ □□□□□□□.

D. □ □□ □□□ "□□ □□ □□"□ □□□□□□□.

Answer: B (LEAVE A REPLY)

NEW QUESTION: 50

DCNM API□ □□□□ □ □□ □□□ □□□□□? (2□ □□)

A. DCNM□ XML □□ SOAP API□ □□□□□.

B. DCNM□ API□ □□□□□ □□□□□ □□□□□.

C. □□ API □□□ DCNM GUI□ □□□□ □□□ □ □□□□.

D. DCNM□ REST □□ API□ □□□□□.

E. DCNM□ □□ □□□ GUI□ □□ □□□□ □□□.

Answer: (SHOW ANSWER)

NEW QUESTION: 51

VM □ □□□□ □□□ □□□□ □□□ □□□ □□ □□ □□□□□ □□□□ Kubernetes □□ □□ □□□□ □□□□□□ □□□□□?

A. Ingress CNI □□□□

B. ACI CNI □□□□

C. Contiv CNI □□□□
D. Calico CNI □□□□
Answer: D (LEAVE A REPLY)

NEW QUESTION: 52
ACI □□□□□ □□□ □□□□□?
A. MitName:□□□
B. □□□ □□:□□□
C. :□□□
D. □□□: □□□ □□
Answer: (SHOW ANSWER)

NEW QUESTION: 53
ACI APIC API □□□□ □□ □□□ □□ □□ □□□ □□□□ □□□□□?
A. CSV
B. JSON
C. YAML
D. XML
Answer: A (LEAVE A REPLY)

NEW QUESTION: 54
□□□ □□:

□□□ □□□□□. □ □□□ □□□□□ Cisco NX-OS □□□□ □□ □□ □□ □□□ □□□ □ □□□?
A. bash-shell □□□
B. bash-shell □□□
C. □□□ bash-shell
D. bash-shell □□
Answer: D (LEAVE A REPLY)

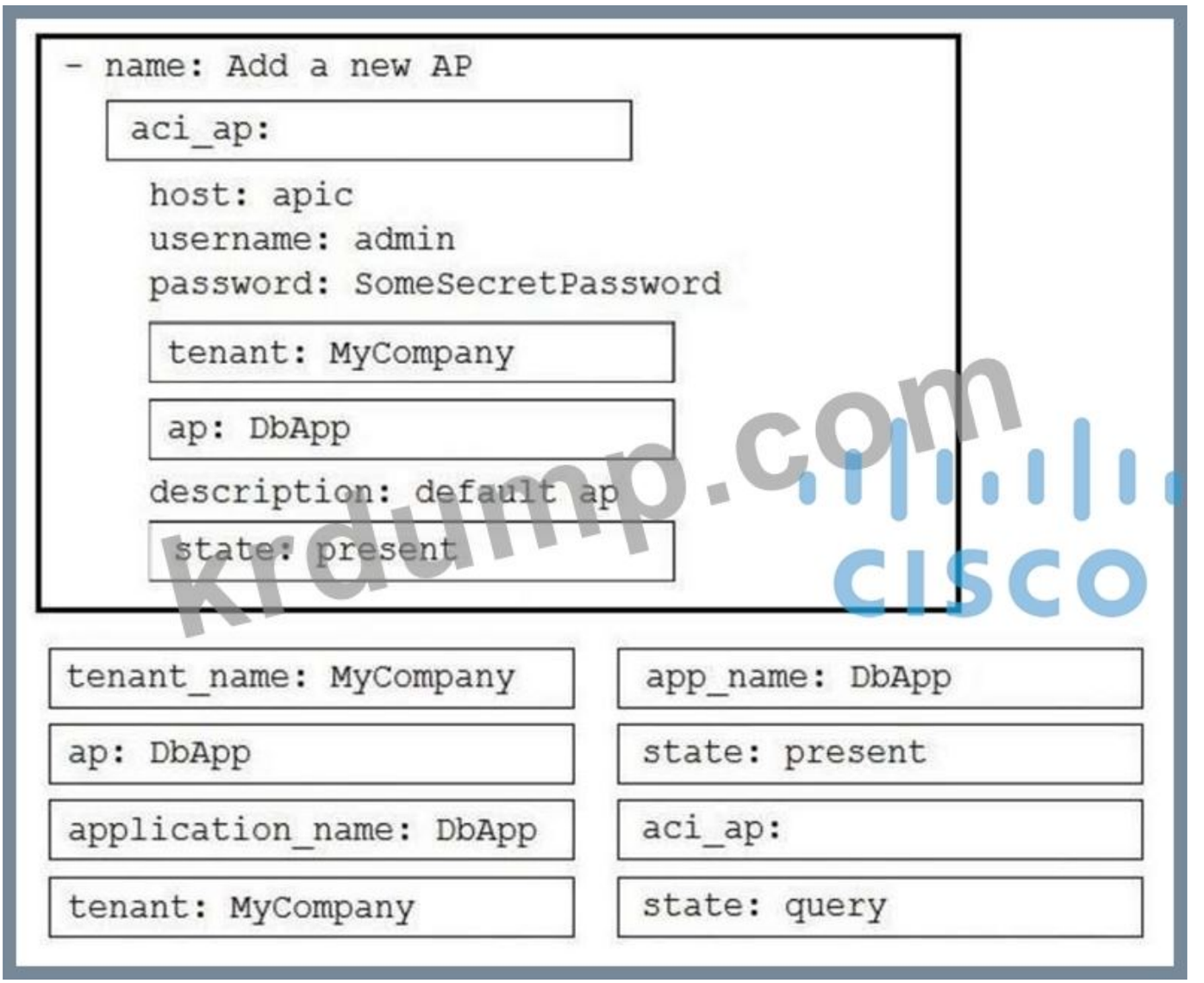
NEW QUESTION: 55
□□□ □ □□

Ansible ACI □□□ □□□□ "DbApp"□□□ □ □□□□□□ □□□□ □□□□□ □□□□ □ □□ YAML □□ □□□ Ansible □□□□ □□ □□□ □□□□ □□□ □□□□. □□ □□□ □ □□□ □□ □□□□.

□□ □ □□:

tenant_name: MyCompany	app_name: DbApp
ap: DbApp	state: present
application_name: DbApp	aci_ap:
tenant: MyCompany	state: query

Answer:



□□: □□□□ □□ □□□ □□ □□□□

□□/□□:

https://docs.ansible.com/ansible/latest/modules/aci_ap_module.html

NEW QUESTION: 56

□□□ □□:

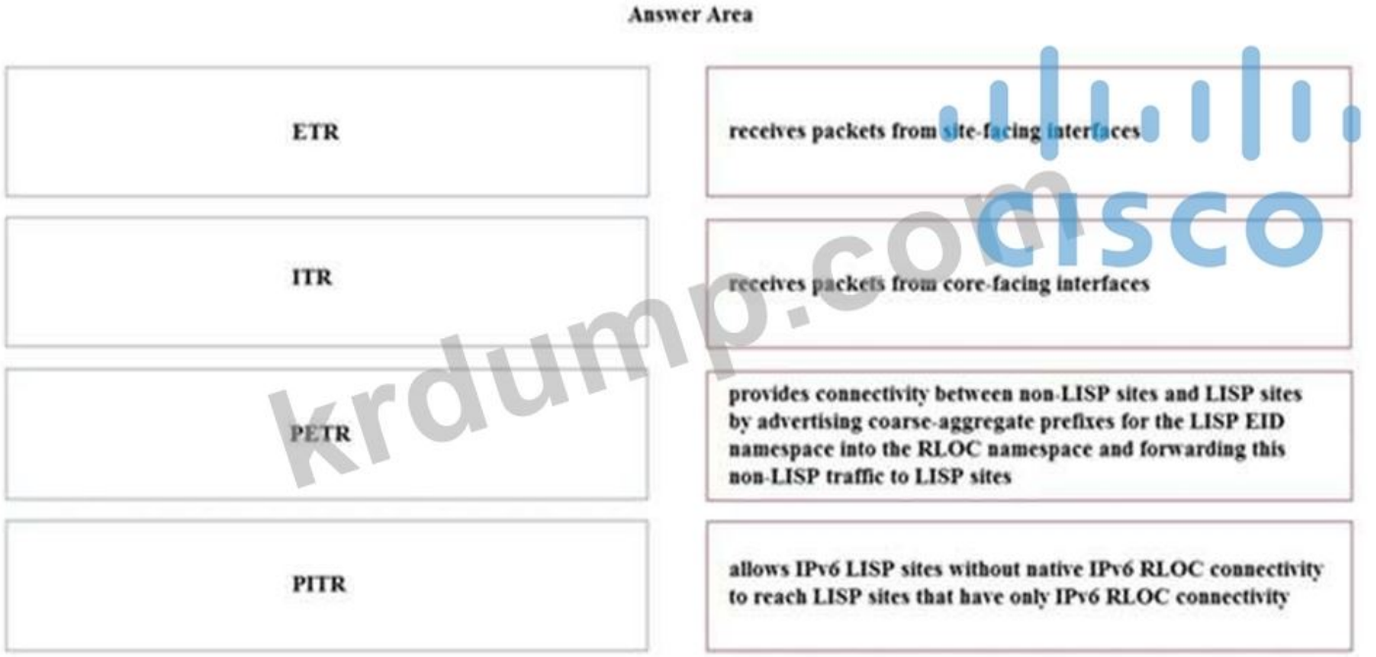
□□□ □□□□□. □ □□□ □□□□□ Cisco NX-OS □□□□ □□ □□ □□ □□□ □□□ □ □□□?

A. □□□ bash-shell
B. bash-shell □□□
C. bash-shell □□
D. bash-shell □□□
Answer: A (LEAVE A REPLY)

NEW QUESTION: 57
□□□ □□□□□.

□□□ □□□ □□ 1/1□ □□ □□□□ □□□□ □□□ □□ □□□□□□□?
A. 10000□□□□□□□ 4□□ □□ □□□□
B. 10□□□ 4□□ □□ □□□□
C. 100□□□□□ 4□□ □□ □□□□
D. 10000□□□ 4□□ □□ □□□□
Answer: B (LEAVE A REPLY)

NEW QUESTION: 58
□□□ □ □□
□□□ LISP □□□ □□□□ □□□ □□□□ □□□ □□□□.
□□ □ □□:



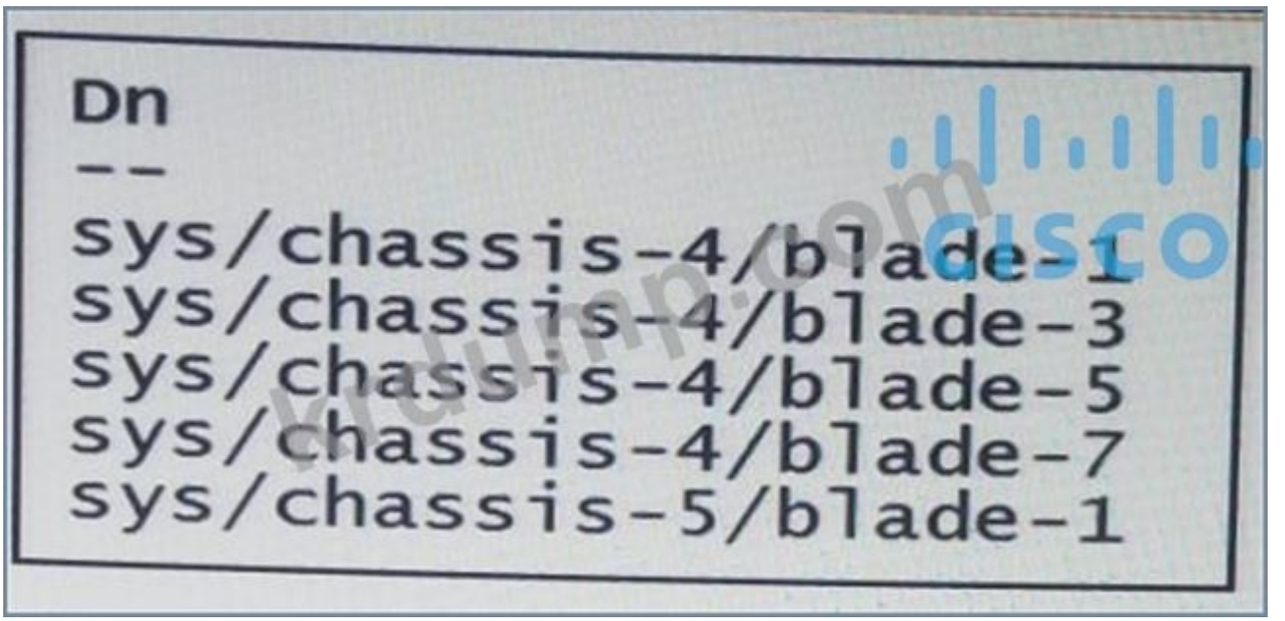
Answer:

□□:

□□: https://www.cisco.com/c/en/us/td/docs/switches/datacenter/sw/nx-os/lisp/configuration/guide/b_NX-OS_LISP_Configuration_Guide/b_NX-OS_LISP_Configuration_Guide_chapter_01.html

NEW QUESTION: 59

□□□ □□:



□ □□□ □□□□ □ Cisco UCS PowerTool □□□ □□□□□? (2□ □□)

A. Get-UcsBlade | □□ □□ Dn

B. Get-UcsSystems | □□ □□ Dn

C. Get-Ucs □□ | □□ □□ Dn

D. Get-UcsRackUnit | □□ □□ Dn

E. Get-UcsRackSystems | □□ □□ Dn

Answer: C,D (LEAVE A REPLY)

NEW QUESTION: 60

□□ □ □□□ API □□□ □□ □□□□ □□ □□?

A. □□ API □□□ □□□ □□□ □□□ □□□ □□□□□.

B. □□□ □□□ □□□ □□□ □□□□□ □ □□□□□.

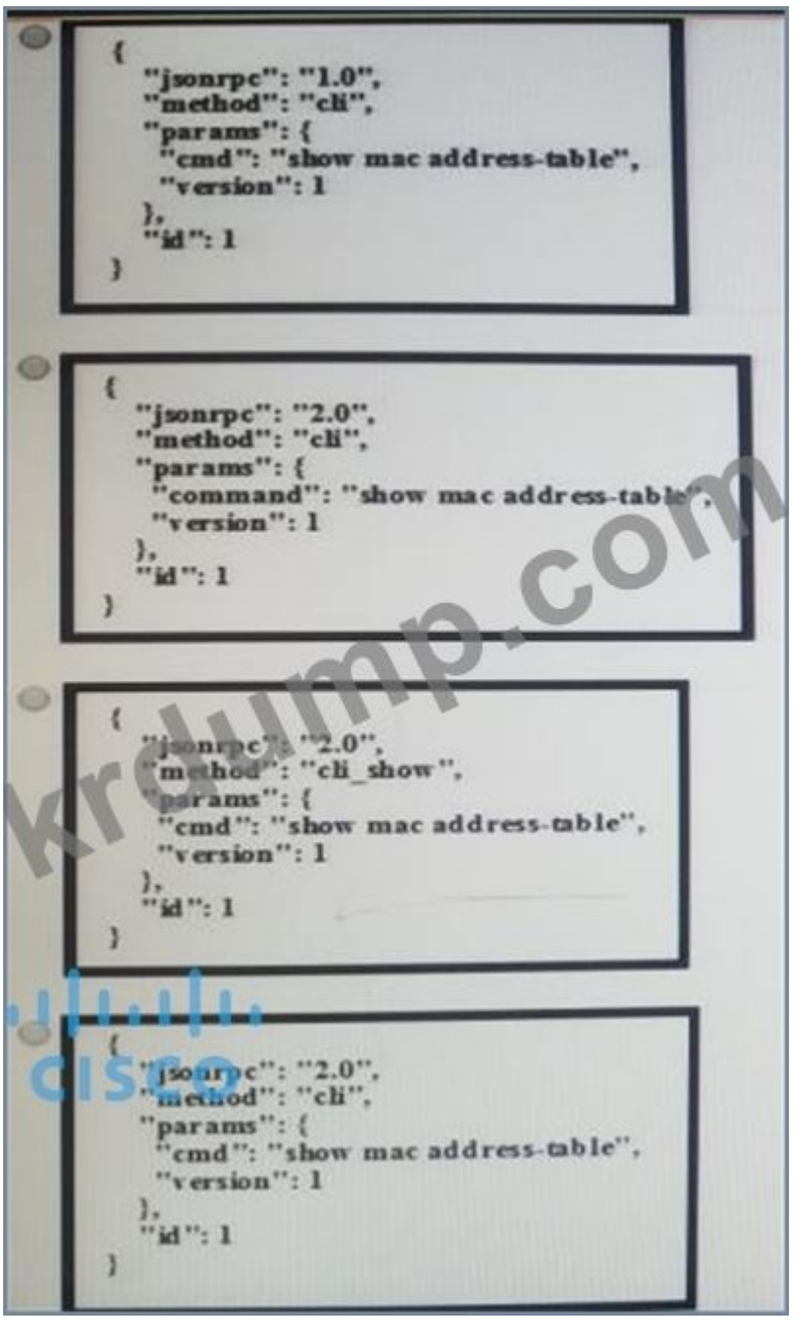
C. □□□ □□□ □□□□□ □□□ □ □□ □□□□□ □□□□□.

D. □□ API □□□ □□ □□□ □□□ □□□□ □□□.

Answer: C (LEAVE A REPLY)

NEW QUESTION: 61

□□ NX-API □□□ MAC □□ □□□□ □□□□□?



A. □□

B. □□

C. □□

D. □□

Answer: A (LEAVE A REPLY)

300-635 □□ □□□ □□□□□ □□ DumpTop □□ □□□□ □□□ 300-635 □□! DumpTop □ □□ **300-635** □□ □□□ □□□□□□, DumpTop 300-635 □□ □□□ □□□ □□□□□ □□□ □□□□□□□. □□□□ □□□ □□□□ □□ DumpTop 300-635 □□□

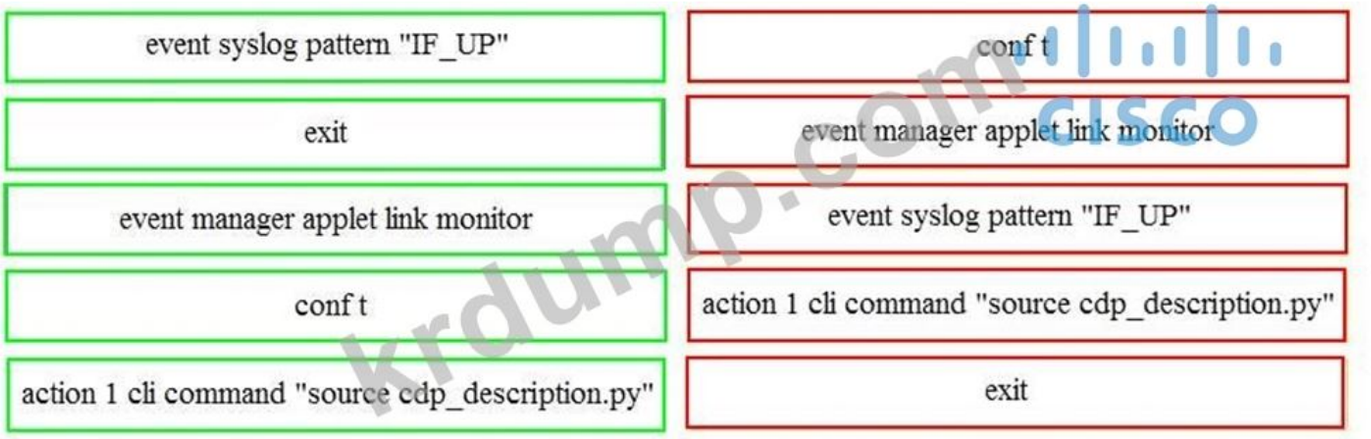
□□□□□. https://www.dumptop.com/Cisco/300-635-dump.html (**76** Q&As Dumps, **30%OFF** **Special Discount: KrDump**)

NEW QUESTION: 62
□□□ "WEST15"□ □□ □□ □□□□ "WEST15" □□□□ "LXT14"□ □□□□□ □□□. □□ □□ □□ □□□ "Vmhost-west 15-01"□ "VMHOST-LXT 14-01"□ □□□□ □□□. Python □□ □□□□ Cisco Intersight REST API□ □□□□ □□□ □□□□□□ □□ API □□□ □□□ □ □□ □□□ "WEST 15" □□□□ □□ □□ □□□□ □□□□ □ □□□□ □ □□ GET API □ □□ □□□□□? "sp_name" □□□ □□□ □□ □□□□ □□□ □□□□ □□□ □□□□□. (2□□ □□□□□.)
A. ('WEST15')□□ https://intersight.com/api/v1/server/Profiles?$select=Name&filter=Name □□ □□
B. GET https://intersight.com/api/v1/server/Profiles?$select=Name&$filter=startswith(□□, 'WEST15')
C. GET https://intersight.com/api/v1/server/Profiles?$select=Namefilter=contains(□□, 'WEST15')
D. BODY = {"□□": sp_name.format('WEST15,'LXT14')}
E. BODY = {" □□": sp_name.replace('WEST15','LXT14')}
Answer: (SHOW ANSWER)

NEW QUESTION: 63
□□□ "WEST15"□ □□ □□ □□□□ "WEST15" □□□□ " LXT14"□ □□□□□ □□□. □ □ □□ □□ □□□ "Vmhost-west 15-01"□ "VMHOST-LXT 14-01"□ □□□□ □□□. Python □ □□□□□ Cisco Intersight REST API□ □□□□ □□□ □□□□□□ □□ API □□□ □□□ □□□ □□□ "WEST 15" □□□□ □□ □□ □□□□ □□□□ □ □□□□ □ □□ GET API □□□ □□□□□? "sp_name" □□□ □□□ □□ □□□□ □□□ □□□□ □□□ □□□□ □. (2□□ □□□□□.)
A. https://infrsight.com/apiAv1/servirtProfiles?$select=Name&filter=□□('WEST15') □□□□
B. BODY = {" □□": sp_name.replace(WEST15','LXT14)}
C. https://infrsight.com/apiAv1/servirtProfiles?$select=Name&filter=contains □□□□(□□, 'WEST15')
D. BODY = {"□□": sp_name.format('WEST15,'LXT14')}
E. GET https://intersight.com/api/v1server/Profiles?$select=Name&$filter=startswith(□□, 'WEST15')
Answer: E (LEAVE A REPLY)

NEW QUESTION: 64
Cisco Nexus □□□ □□□□□□ □□□□□ □□ □□□ □□□ □□□□□ EEM Python □□ □ □□□□ □□□□□ □□□ □□□ □□□□□ □□□.
□□□□□ □□□□ □□ □□□ □□□□ □□□□ □□□ □□□ □□□ □□□□.

Answer:



□□:

conf

□□□ □□□ □□□ □□ □□□

□□□ □□□ □□ □□ "IF_UP"

□□ 1 cli □□ "□□ cdp_description.py"

□□

□□:

https://www.slideshare.net/CiscoDevNet/automating-with-nxos-lets-get-started

NEW QUESTION: 65

□□□ □□□□□.

```
[admin@guestshell ~]$ pwd
/home/admin
[admin@guestshell ~]$
[admin@guestshell ~]$
[admin@guestshell ~]$ more deltacounter.py
#!/isan/bin/python

from cli import *
import sys, time

ifName = sys.argv[1]
delay = 2
count = 5
cmd = 'show interface ' + ifName + ' counters'

out = json.loads(clid(cmd))
rxuc = int(out['TABLE_rx_counters']['ROW_rx_counters'][0]['eth_inucast'])
rxmc = int(out['TABLE_rx_counters']['ROW_rx_counters'][1]['eth_inmcast'])
rxbc = int(out['TABLE_rx_counters']['ROW_rx_counters'][1]['eth_inbcast'])
txuc = int(out['TABLE_tx_counters']['ROW_tx_counters'][0]['eth_outucast'])
txmc = int(out['TABLE_tx_counters']['ROW_tx_counters'][1]['eth_outmcast'])
txbc = int(out['TABLE_tx_counters']['ROW_tx_counters'][1]['eth_outbcast'])
print ('row rx_ucast rx_mcast rx_bcast tx_ucast tx_mcast tx_bcast')
print ('===================================')
print ('    %8d %8d %8d %8d %8d %8d'  % (rxuc, rxmc, rxbc, txuc, txmc, txbc))
print ('===================================')

i = 0
while (i < count):
    time.sleep(delay)
    out = json.loads(clid(cmd))
    rxucNew = int(out['TABLE_rx_counters']['ROW_rx_counters'][0]['eth_inucast'])
    rxmcNew = int(out['TABLE_rx_counters']['ROW_rx_counters'][1]['eth_inmcast'])
    rxbcNew = int(out['TABLE_rx_counters']['ROW_rx_counters'][1]['eth_inbcast'])
    txucNew = int(out['TABLE_tx_counters']['ROW_tx_counters'][0]['eth_outucast'])
    txmcNew = int(out['TABLE_tx_counters']['ROW_tx_counters'][1]['eth_outmcast'])
    txbcNew = int(out['TABLE_tx_counters']['ROW_tx_counters'][1]['eth_outbcast'])
    i += 1
    print ('%-3d %8d %8d %8d %8d %8d %8d'  % \
      (i, rxucNew - rxuc, rxmcNew - rxmc, rxbcNew - rxbc, txucNew - txuc, txmcNew - txmc,

[admin@guestshell ~]$
```

□ □□□□□ deltacounters.py□□ □□ □□ Cisco NX-OS □□□ □□□ □□□□ □□□ □ □□□□ □□□ □□□□.

□ □□□□□ □□□□□ □□□□ Cisco NX-OS □□□ □□□□□?

A. □□□ /home/admin/bootflash:deltacounters.py ethemet1/1

B. guestshell □□ python /home/admin/deltacounter.py ethernet1/1

C. python bootflash:deltacounters.py ethernet1/1 □□

D. guestshell □□ python /home/admin/deltacounter.py ethernet1/1

Answer: (SHOW ANSWER)

NEW QUESTION: 66

□□□ □□□□□.

```
https://APIC_IP/api/class/l1PhysIf.xml?query-target-filter=eq(l1PhysIf.speed,"10G")
```

ACI APIC□ □□ □ API GET □□□ □□ □□□□ □□ □□? (2□□ □□□□□.)

A. API □□□ XML□ □□□□□□.

B. API □□□ JSON□□ □□□□□□.

C. API □□□ □□ □□□□□ □□□ □□□□.

D. API □□□ □□ □□□□ □□□ □□□□.

E. API □□□ APIC□□ □□□ 10G □□□□□□ □□□□□.

Answer: A,D (LEAVE A REPLY)

NEW QUESTION: 67

□□□ □□□□□.

```
from cobra.mit.access import MoDirectory
from cobra.mit.session import LoginSession
from cobra.model.pol import Uni
from cobra.model.fv import Tenant
from cobra.mit.request import ConfigRequest

uri = 'https://APIC_IP/'
user = 'APIC_USERNAME'
pw = 'APIC_PW'

ls = LoginSession(uri, user, pw)
md = MoDirectory(ls)
md.login()

topMo = Uni('')
[                                    ]
c = ConfigRequest()
c.addMo(fvTenant)
md.commit(c)

md.logout()
```


□□□ APIC □□□□□□ □ □□□□□ □□□ □ □□□□ Cobra SDK□ □□ Cisco□□ □ □ □□□ □□□□ □□□.

□ □□□□ □□□□ □□□ □□□ □□ □□□ □□□□ □□□?

A. fvTenant = NewTenant(□□='Cisco')

B. □□□ = □□□(topMo. name='Cisco')

C. fvTenant = □□□(topMo, □□='Cisco')

D. fvTenant = □□□('Cisco')

Answer: (SHOW ANSWER)

□□: □□□□ □□ □□□ □□ □□□□

NEW QUESTION: 68

□□ □ □□□ ACI □□□□□ □□□ □□ □□ □□□□ Ansible □□□□ □□□ □□□□□?

A. □□

B. □□

C. □□

D. □□

Answer: (SHOW ANSWER)

NEW QUESTION: 69

□□□ □□□□ □□□ □□ □□□□□ □□ □□ □□□□□ □□□□□ □□ □ □□□□ □ □□ □□□□ □□□.

□ □ □□□□ □□ □□ □□□□□ □□□ □□□□ □□□□?

A. □□ □□ txt □□□ □ □□□□ □□□□□.

B. □ □□ □□□ □□□□ □ □□□□ pip □□□ □□□□□.

C. □□ □□□□ □□□ □□□□ □ □ □□□□ pip □□□ □□□□□.

D. □ □□ □□□ □□□ □ □□□□ pip □□□ □□□□□.

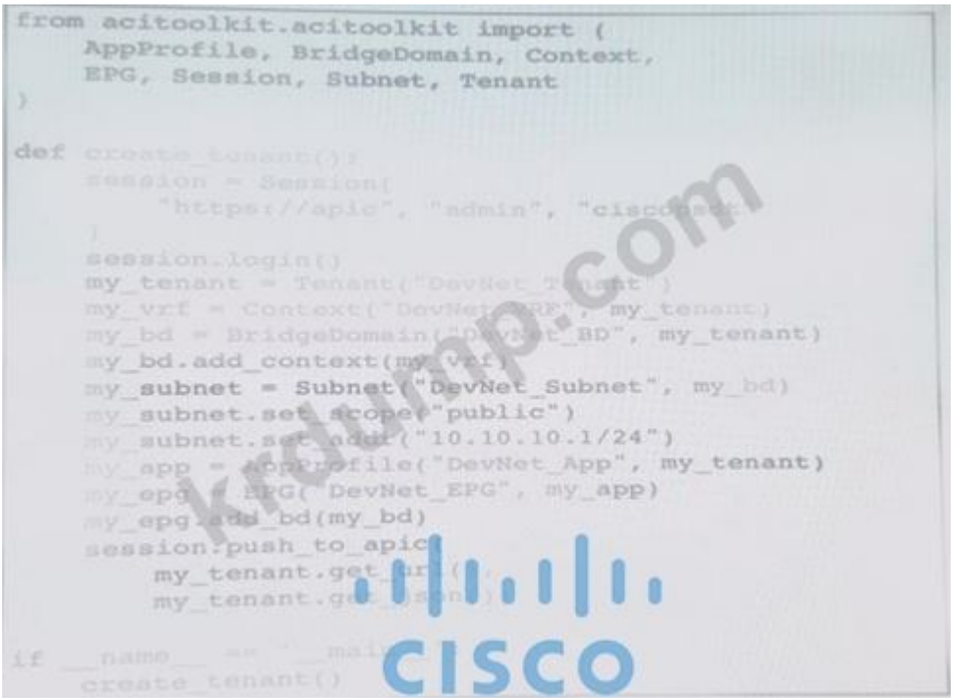
Answer: (SHOW ANSWER)

□□:

□□□ □□

NEW QUESTION: 70

□□□ □□:



□□□□ □□□□□□. □ Python □□□ Cisco ACI□□ □□ □ □□ □□□ □□□□□? (2□ □□) ACI □□□ "DevNet_Tenant"□ □□□ VRF "DevNet_VRF" □□□ "DevNet_Subnet" □□ □□ □□□□ □□□ "private"□ □□□□□.

A. ACI □□□□ □□□ AppProfile "DevNet_App" □□□ "DevNet_Subnet" □□□□ □□□□□. "DevNet_Tenant" □ □□□□ □□□ "10.10.10 1/24"□ □□□□□.

B. ACI □□□ "DevNet_Tenant"□ □□□ VRF "DevNet_VRF" □□□ EPG "DevNet_EPG"□ □ □□□ BridgeDomain "DevNet BD"□ EPG□ □□□□□.

C. ACI □□□□ □□□ VRF "DevNet_VRF" □□□ "DevNet_Subnet" □□□□ □□□□□. "DevNet_Tenant" □ □□□□ □□□ "10.10 10.1/24"□ □□□□□.

D. ACI □□□□ □□□ AppProfile "DevNet_App" □□□ EPG "DevNet_EPG"□ □□□□□.

"DevNet_Tenant" □ EPG□ BridgeDomain "DevNet_BD"□ □□
Answer: C,D (LEAVE A REPLY)