



# 300-635<sup>Q&As</sup>

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.pass4itsure.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**

What are two capabilities of the DCNM REST API? (Choose two.)

- A. uses the dcnm-token header for requests authentication after initial basic authentication
- B. uses basic authentication without encoding for username and password
- C. supports HTTP until release 11.0(1)
- D. uses bearer key authorization
- E. is separated into Classic LAN, LAN Fabric, Media Controller, and SAN Management categories

Correct Answer: CD

Reference:

<https://developer.cisco.com/docs/data-center-network-manager/#!cisco-dcnm-classic-lan-rest-api> Check for /rest/login

---

**QUESTION 2**

An engineer is implementing a Cisco Nexus 9000 Series Switch. To automate the configuration, which command enables Bash on a Cisco NX-OS?

- A. run bash-shell
- B. enable bash
- C. run bash
- D. feature bash-shell

Correct Answer: D

---

**QUESTION 3**

When Cisco Nexus switches are implemented, which command requires the method key to perform a POST operation using the NXAPI-CLIJSON-RPC interface?

- A. cli
- B. cli\_conf
- C. cli\_show
- D. cli\_post

Correct Answer: B

---

**QUESTION 4**

Using the NX-API CLI JSON-RPC interface, which two Python data structure and requests call create an SVI? (Choose two.)

- A. 

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

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

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

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

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

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



E. Option E

Correct Answer: AD

Sandbox

### QUESTION 5

DRAG DROP

Drag and drop the code to complete an Ansible playbook that creates a new tenant. Not all options are used.

Select and Place:

```
- name: Add a new tenant
[ ]

  host: apic
  username: admin
  password: SomeSecretPassword
  [ ]
  description: MyCompany tenant
  [ ]
```

Tenant\_name: MyCompany

state: absent

state: query

tenant: MyCompany

aci\_tenant:

state: present

state: create

aci\_tenant\_name:

Correct 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
```

```
state: create
```

```
state: absent
```

```
aci_tenant_name:
```

Reference: [https://docs.ansible.com/ansible/2.4/aci\\_tenant\\_module.html](https://docs.ansible.com/ansible/2.4/aci_tenant_module.html)

[300-635 Practice Test](#)

[300-635 Study Guide](#)

[300-635 Braindumps](#)