ITTEST

QUESTION & ANSWER

Guías de estudio precisos, Alta tasa de paso!

Ittest ofrece información actualizada de forma gratuita en un año!

Exam : **300-635**

Title : Automating and Programming Cisco Data Center Solutions (DCAUTO)

Version : DEMO

1.Which two application isolation options are available when Kubernetes is deployed with the ACI CNI plug-in? (Choose two.)

A. Process Isolation

B. Namespace Isolation

C. Cluster Isolation

D. Server Isolation

E. VM Isolation

Answer: B,C

2.Refer to the exhibit.

```
event manager applet eem-correlate
event syslog tag one pattern "copy bootflash:.* running-config.*"
event syslog tag two pattern "copy run start"
event syslog tag three pattern "hello"
tag one or two or three happens 1 in 120
action 1.0 reload module 1
```

What results from the EEM script that runs on a Cisco Nexus 9000 switch?

A. Module 1 is reloaded if one of the specified console patterns occurs within 120 seconds.

B. Module 1 is reloaded if one of the specified syslog patterns occurs within 120 seconds.

C. Module 1 is reloaded if all of the specified syslog patterns occur within 120 seconds.

D. The message "reload module 1" is printed if one of the specified syslog patterns occurs within 120 seconds

Answer: D

3.Refer to the exhibit

```
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
```

Which action does the execution of this ACI Cobra Python code perform?

A. It prints all LLDP neighbor MAC and IP addresses

B. It prints all Cisco Discovery Protocol neighbor MAC and IP addresses

C. It prints all endpoint MAC and IP addresses

D. It prints all APIC MAC and IP addresses

Answer: C

Explanation:

Reference: 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.ht ml

4.A server profile with the "WEST15" in its name must have the string "WEST15" changed to "LXT14". For example, server profile "Vmhost-west 15-01" would need to be changed to "VMHOST-LXT 14-01".

Using the Cisco Intersight REST API in a Python script, which two GET API requests are used to retrieve

just the server profile with the string "WEST 15" in the name and the correct body for the API request to update the name? Assume the variable "sp_name" contains the name of the retrieved server profile. (Choose two.)

A. GET https://intersight.com/api/v1/server/Profiles?$select=Namefilter=contains (Name,'WEST15')
B. GET https://intersight.com/api/v1/server/Profiles?$select=Name&filter=Name in ('WEST15')
C. BODY = {"Name": sp_name.format('WEST15,'LXT14')}
D. GET https://intersight.com/api/v1/server/Profiles?$select=Name&$filter=startswith(Name, 'WEST15')
E. BODY = {" Name": sp_name.replace('WEST15','LXT14')}

Answer: B,E

Explanation:

Reference:

https://developer.cisco.com/codeexchange/github/repo/CiscoUcs/intersightpython/

https://intersight.com/apidocs/introduction/query/#referencing-properties-by-name

5.DRAG DROP

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

```
- 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:

Answer:

```
- name: Add a new tenant
  aci_tenant:

    host: apic
    username: admin
    password: SomeSecretPassword

    tenant: MyCompany

    description: MyCompany tenant

    state: present
```

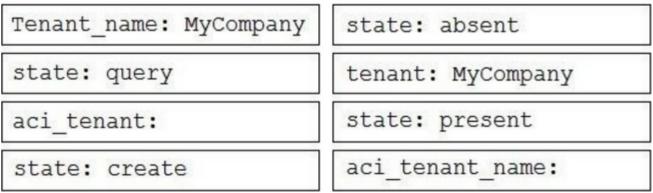|	
Tenant_name: MyCompany	state: absent
state: query	tenant: MyCompany
aci_tenant:	state: present
state: create	aci_tenant_name:

Explanation:

aci_tenant tenant : MyCompany state: present