350-901^{Q&As}

Developing Applications Using Cisco Core Platforms and APIs (DEVCOR)

Pass Cisco 350-901 Exam with 100% Guarantee

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

https://www.pass4itsure.com/350-901.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

An organization manages a large cloud-deployed application that employs a microservices architecture. No notable issues occur with downtime because the services of this application are redundantly deployed over three or more data center regions. However, several times a week reports are received about application slowness. The container orchestration logs show faults in a variety of containers that cause them to fail and then spin up brand new.

Which action must be taken to improve the resiliency design of the application while maintaining current scale?

A. Update the base image of the containers.

B. Test the execution of the application with another cloud services platform.

C. Increase the number of containers running per service.

D. Add consistent "try/catch(exception)" clauses to the code.

Correct Answer: A

QUESTION 2

Refer to the exhibit.

```
name: VRFs
ios_vrf:
    vrfs: "{{ local_vrfs }}"
    state: present
    purge: yes
```

The YAML represented is using the ios_vrf module. As part of the Ansible playbook workflow, what is the result when this task is run?

A. VRFs not defined in the host_vars file are removed from the device.

B. VRFs not defined in the host_vars file are added to the device, and any other VRFs on the device remain.

C. VRFs defined in the host_vars file are removed from the device.

D. VRFs are added to the device from the host_vars file, and any other VRFs on the device are removed.

Correct Answer: D

QUESTION 3

```
module: ietf-interfaces
  +--rw interfaces
  |  +--rw interface* [name]
  |  +--rw name                       string
  |  +rw description?                 string
  |  +--rw type                       identityref
  |  +--rw enabled?                   boolean
  |  +--rw link-up-down-trap-enable?  enumeration {if-mib}?
  +--ro  interfaces-state
     +--ro interface* [name]
        +--ro name             string
        +--ro type             identityref
        +--ro admin-status     enumeration {if-mib}?
        +--ro oper-status      enumeration
        +--ro last-change?     yang:date-and-time
        +--ro if-index         int32 {if-mib}?
        +--ro phys-address?    yang:phys-address
        +--ro higher-layer-if* interface-state-ref
        +--ro lower-layer-if*  interface-state-ref
        +--ro speed?           yang:gauge64
        +--ro statistics
                    +--ro discontinuity-time    yang:date-and-time
                    +--ro in-octets?            yang:counter64
                    +--ro in-unicast-pkts?      yang:counter64
                    +--ro in-broadcast-pkts?    yang:counter64
                    +--ro in-multicast-pkts?    yang:counter64
                    +--ro in-discards?          yang:counter32
                    +--ro in-errors?            yang:counter32
                    +--ro in-unknown-protos?    yang:counter32
                    +--ro out-octets?           yang:counter64
                    +--ro out-unicast-pkts?     yang:counter64
                    +--ro out-broadcast-pkts?   yang:counter64
                    +--ro out-multicast-pltas?  yang:counter64
                    +--ro out-discards?         yang:counter32
                    +--ro out-errors?           yang:counter32
```

```python
import requests
url = ("https://ios-xe-mgmt.cisco.com:9443/restconf/data/ietf-interfaces:" +
    "interfaces/interface=GigabitEthernet2")

headers = {
    'Accept': "application/yang-data+json",
    'Authorization': "Basic cm9vdDpEpEX1ZheSFfMTAm",
    'Content-Type': "application"
}

response = requests.request(rest_operation, url, data=payload,
    headers = headers, verify=False)

print (response.text)
```

Refer to the exhibits. An interface named "GigabitEthernet2" has been configured on a Cisco IOS XE device. Using RESTCONF APIs as defined by the ietf-interfaces@2014-05- 08.yang model, which two combinations of "rest_operation" and "payload" must be added to the Python script to set the "description" to "Configured by RESTCONF"? (Choose two.)

A.
```
rest_operation = "PATCH"

payload = " {\n      \"ietf-interfaces:interface\": {\n
    \"name\": \"GigabitEthernet2\", \n
    \ "description\": \"Configured by RESTCONF\" \n
    }\n}"
```

B.
```
rest_operation = "PUT"

payload = " {\n      \"ietf-interfaces:interface\": {\n
    \"name\": \"GigabitEthernet2\", \n
    \ "description\": \"Configured by RESTCONF\" \n
    }\n}"
```

C.
```
rest_operation = "PUT"

payload = "{\n \"ietf=interfaces:interface\": {\n
    \"name\": \"GigabitEthernet2\",\n
    \"description\": \"Configured by RESTCONF\", \n
    \"type\": \"iana-if-type:ethernetCsmacd\", \n
    \"enabled\" true,\n \"ietf-ip:ipv4\": {\n
        \"address\": [\n{\n\"ip\": \"10.255.255.1\",
        \n"netmask\": \"255.255.255.0\"\n  }\n
    ]\n         }\n         }\n}"
```

D.
```
rest_operation = "POST"

payload = " {\n      \"ietf-interfaces:interface\": {\n
    \"name\": \"GigabitEthernet2\", \n
    \ "description\": \"Configured by RESTCONF\" \n
    }\n}"
```

E.
```
rest_operation = "POST"

payload = "{\n \"ietf=interfaces:interface\": {\n
    \"name\": \"GigabitEthernet2\",\n
    \"description\": \"Configured by RESTCONF\", \n
    \"type\": \"iana-if-type:ethernetCsmacd\", \n
    \"enabled\" true,\n \"ietf-ip:ipv4\": {\n
        \"address\": [\n{\n\"ip\": \"10.255.255.1\",
        \n"netmask\": \"255.255.255.0\"\n  }\n
    ]\n         }\n         }\n}"
```

A. Option A

B. Option B

C. Option C

D. Option D

E. Option E

Correct Answer: AB

QUESTION 4

A team is developing a cloud-native application for network monitoring and management of various devices. An increased growth rate of users is expected. The solution must be easily managed and meet these requirements:

able to use dependencies easy disposability flexible configuration

Which application design approach must be used?

A. waterfall model

B. 12-factor app framework

C. object-oriented programming

D. agile software development

Correct Answer: B

Explanation: This framework is designed to provide a consistent set of practices and principles to ensure applications can be easily deployed and managed in the cloud. It utilizes a microservices architecture which allows applications to be broken up into smaller, more manageable components. In addition, the 12-factor App Framework makes use of dependencies, flexible configuration and disposable services, making it an ideal choice for this type of application.

QUESTION 5

DRAG DROP

Drag and drop the steps from the left into the order on the right to host a Docker-contained application on a Cisco Catalyst 9000 Series Switch.

Select and Place:

Activate and start the hosted application.	step 1
Upload the containerized application to a repository, save the container as a .tar file, and prepare and format the USB drive.	step 2
Configure the application that hosts vNIC on the switch.	step 3
Download the application to usbflash1 and configure the relevant VLAN interface on the switch.	step 4
Design, develop, and dockerize the application.	step 5
Install the application on the switch.	step 6

Correct Answer:

	Design, develop, and dockerize the application.
	Upload the containerized application to a repository, save the container as a .tar file, and prepare and format the USB drive.
	Download the application to usbflash1 and configure the relevant VLAN interface on the switch.
	Install the application on the switch.
	Configure the application that hosts vNIC on the switch.
	Activate and start the hosted application.