DUMPSARENA

# Developing Applications using Cisco Core Platforms and APIs (DEVCOR)

## Cisco 350-901

**Version Demo**

**Total Demo Questions: 20**

**Total Premium Questions: 472**

**Buy Premium PDF**

https://dumpsarena.co

sales@dumpsarena.co

sales@dumpsarena.co
dumpsarena.co

# Topic Break Down

| Topic | No. of Questions |
|---|---|
| Topic 1, New Update | 246 |
| Topic 2, Software Development and Design | 38 |
| Topic 3, Using APIs | 43 |
| Topic 4, Cisco Platforms | 53 |
| Topic 5, Application Deployment and Security | 49 |
| Topic 6, Infrastructure and Automation | 43 |
| Total | 472 |

## QUESTION NO: 1

What is an effective logging strategy according 10 the 12-factor app tenets?

**A.** Capture togs by the execution environment and route to a centralized destination

**B.** Tag and save togs in a local document database that has querying capabilities.

**C.** Back up log files in a high-availability remote cluster on the public cloud

**D.** Timestamp and save logs in a local time-series database that has querying capabilities

**ANSWER: A**

## QUESTION NO: 2

The UCS Python SDK includes modules for Service Profile template creation.

Which two UCS Service Profile template types are supported? (Choose two.)

**A.** initial-template

**B.** updating-template

**C.** abstract-template

**D.** attached-template

**E.** base-template

**ANSWER: A B**

## QUESTION NO: 3

An application has these characteristics:

• provide one service or function

• distributed database

• API gateway

• central repository for code

• configuration database

• uses session management

Which two design approaches contribute to the scalability of the application? (Choose two.)

**A.** session management in a stateless architecture

**B.** modular design iteration

**C.** distributed computing with tightly coupled components

**D.** built to scale based on a star topology

**E.** planned before the first device is deployed

**ANSWER: B C**

**QUESTION NO: 4**

Which Puppet manifest changes the NTP server and generates the traffic from VLAN 15?

A.

```
ntp_server { '172.30.200.11':
 ensure           => 'present',
 key              => 94,
 prefer           => true,
 minpoll          => 4,
 maxpoll          => 14,
 source_interface => 'Vlan 15',
}
```

B.

```
ntp_server {
 ip               => '172.30.200.11',
 ensure           => 'present',
 key              => 94,
 prefer           => true,
 minpoll          => 4,
 maxpoll          => 14,
 source_interface => '15',
}
```

C.

```
ntp_server {
 server           => '172.30.200.11',
 ensure           => 'present',
 key              => 94,
 prefer           => true,
 minpoll          => 4,
 maxpoll          => 14,
 source_interface => 'Vlan 15',
}
```

D.

```
ntp_server { '172.30.200.11':
 ensure           => 'present',
 key              => 94,
 prefer           => true,
 minpoll          => 4,
 maxpoll          => 14,
 vlan             => '15',
}
```

**A.** Option A

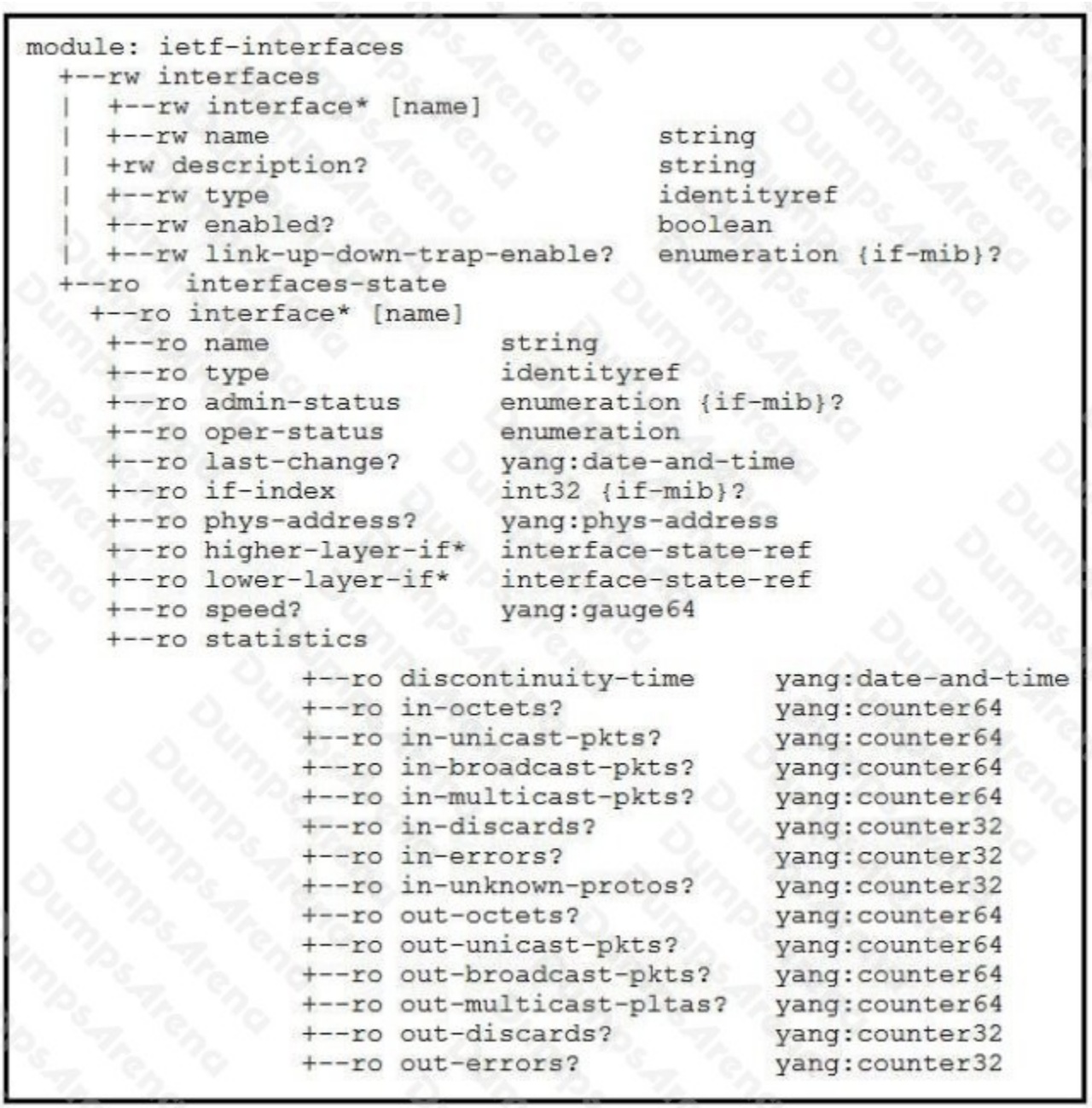
**B.** Option B

**C.** Option C

**D.** Option D

**ANSWER: C**

**QUESTION NO: 5**



```
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
```

```
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 cm9vdDpEX1ZheSFfMTAm",
     '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

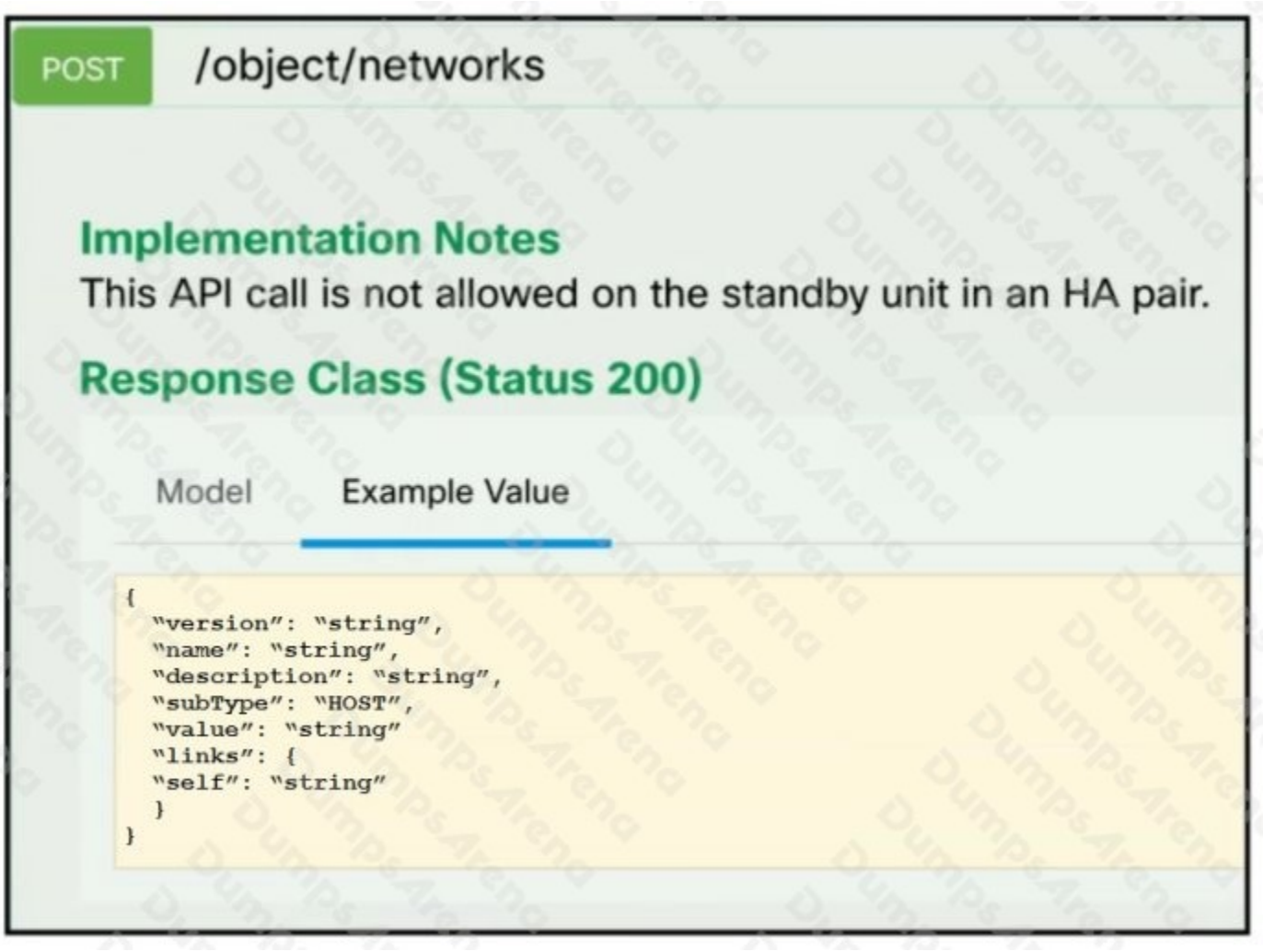
**C.** Option C

**D.** Option D

**E.** Option E

**ANSWER: A B**

**QUESTION NO: 6**




Refer to the exhibit. A developer must create a new network object named testnetwork by using the Cisco Firepower Device Management API. The script must also handle any exceptions that occur during the request and print out any resulting errors. Which script must be used?

A.

```
import requests, json
headers = { 'Content-type': 'application/json'}
data = {"name": "testnetwork",
"description": "Test Network", "subType" : "HOST",
"value": "192.168.1.1", "type" : "networkobject"}
try:
    response = requests.post(
        'https://firepower-server/object/networks',
        data=data)
except:
    print(error)
```

B.

```
import requests, json
headers = { 'Content-type': 'application/json'}
data = {"name": "testnetwork",
"description": "Test Network", "subType" : "HOST",
"value": "192.168.1.1", "type" : "networkobject"}
try:
    response = requests.post(
        'https://firepower-server/object/networks',
        data=data, headers=headers)
    response.raise_for_status()
except:
    print(error)
```

C.

```
import requests, json
headers = { 'Content-type': 'application/json'}
data = {"name": "testnetwork",
"description": "Test Network", "subType" : "HOST",
"value": "192.168.1.1", "type" : "networkobject"}
try:
    response = requests.post(
        'http://firepower-server/object/networks',
        data=json.dumps(headers), headers=data)
    response.raise_for_status()
except:
    print(error)
```

D.

```
import requests, json
headers = { 'Content-type': 'application/json'}
data = {"name": "testnetwork",
"description": "Test Network", "subType" : "HOST",
"value": "192.168.1.1", "type" : "networkobject"}
try:
    response = requests.post(
        'https://firepower-server/object/networks',
        data=json.dumps(data), headers=headers)
    response.raise_for_status()
except:
    print(error)
```

**A.** Option A

**B.** Option B

**C.** Option C

**D.** Option D

**ANSWER: A**

**QUESTION NO: 7**

What are two methods for sending bearer access tokens in resource requests to servers? (Choose two,)

**A.** in plaintext for user access

**B.** in the HTTP API schema

**C.** in the HTTP request entity-body

**D.** in the "Authorization" request header field

**E.** in the HTTP request URI

**ANSWER: C D**

**QUESTION NO: 8**

```
image: docker:19.03.1
services:
  - name: docker:19.03.1-dind

stages:
  - build_container
  - get_config

variables:
  DOCKER_DRIVER: overlay2
  DOCKER_TLS_CERTDIR: ""
  ANSIBLE_HOST_KEY_CHECKING: "False"

Build container and install Dependencies:
  stage: build_container
  before_script:
   - docker info
   - docker login registry.gitlab.com -u/$DOCKER_USERNAME/$DOCKER_REPOSITORY
"$DOCKER_PASSWORD"
script:
   - docker build . -t registry.gitlab.com/$DOCKER_USERNAME/$DOCKER_REPOSITORY
    - docker run -t -d - -rm - -name nettest registry.gitlab.com/
$DOCKER_USERNAME/$DOCKER_REPOSITORY
    - docker commit nettest registry.gitlab.com/$DOCKER_REPOSITORY
after_script:
 -
Connect to Cisco Sandbox and backup config:
 image: registry.gitlab.com/$DOCKER_USERNAME/$DOCKER_REPOSITORY
stage: get_config
 script:
 - ansible-playbook gather_and_process_configs.yml -i inventory
```

Refer to the exhibit. A developer wants to automatically deploy infrastructure for a containerized application. A .gitlab-ci.yml file must describe a pipeline that builds a container based on a supplied Dockerfile and executes an Ansible playbook on the configured container. What must be added where the code is missing to complete the script?

A.
```
docker assign nettest
registry.gitlab.com/DOCKER USERNAME/$DOCKER REPOSITORY
```

B.
```
docker info registry.gitlab.com/$DOCKER REPOSITORY
```

C.
```
docker push registry.gitlab.com/
$DOCKER USERNAME/$DOCKER REPOSITORY
```

D.
```
docker logout registry.gitlab.com
```

**A.** Option A

**B.** Option B

**C.** Option C

**D.** Option D

**ANSWER: C**

## QUESTION NO: 9

Which two countermeasures help reduce the risk of playback attacks? (Choose two.)

**A.** Store data in a NoSQL database.

**B.** Implement message authentication (HMAC).

**C.** Enable end-to-end encryption.

**D.** Remove stack traces from errors.

**E.** Use short-lived access tokens.

**ANSWER: B E**

## QUESTION NO: 10

Which two types of organization are subject to GDPR? (Choose two.)

**A.** only organizations that operate outside the EU

**B.** any organization that offers goods or services to customers in the EU

**C.** only organizations that have offices in countries that are part of the EU

**D.** any organization that operates within the EU

**E.** only organizations that physically reside in the EU

**ANSWER: B D**

**Explanation:**

Reference: https://www.cisco.com/c/en/us/products/security/comply-with-GDPR.html

## QUESTION NO: 11

```
response = requests.get(url)
if response.status_code != 200:
    error_message = "Unexpected HTTP Response code: {}".format(response.status_code)
    raise Exception(error_message)
data = response.json()
```

Refer to the exhibit. This snippet of a script has recently started exiting abnormally with an exception stating “Unexpected HTTP Response code: 429”.

Which solution handles rate limiting by the remote API?

A.
```
response = requests.get(url)
if response.status_code == 429:
    backoff_seconds = int(response.headers['Retry-After'])
    sleep(backoff_seconds)
elif response.status_code != 200:
    error_message = "Unexpected HTTP Response code: {}".format(response.status_code)
    raise Exception(error_message)
data = response.json()
```

B.
```
response = requests.get(url)
if response.status_code != 200 and response.status_code != 429:
    error_message = "Unexpected HTTP Response code: {}".format(response.status_code)
    raise Exception(error_message)
data = response.json()
```

C.
```
response = requests.get(url)
if response.status_code != 200 and response.status_code != 429:
    backoff_seconds = int(response.headers['Retry-After'])
    sleep(backoff_seconds)
    error_message = "Unexpected HTTP Response code: {}".format(response.status_code)
    raise Exception(error_message)
data = response.json()
```

D.
```
response = requests.get(url)
if response.status_code == 429:
    backoff_seconds = int(response.headers['Retry-After'])
    sleep(backoff_seconds)
    response = requests.get(url)
elif response.status_code != 200:
    error_message = "Unexpected HTTP Response code: {}".format(response.status_code)
    raise Exception(error_message)
data = response.json()
```

**A.** Option A

**B.** Option B

**C.** Option C

**D.** Option D

**ANSWER: D**

## QUESTION NO: 12

Into which two areas are AppDynamics APIs categorized? (Choose two.)

**A.** application-centric

**B.** analytics-events

**C.** database-visibility

**D.** platform-side

**E.** agent-side

**ANSWER: D E**

## QUESTION NO: 13

There is a requirement to securely store unique usernames and passwords. Given a valid username, it is also required to validate that the password provided is correct.

Which action accomplishes this task?

**A.** Encrypt the username, hash the password, and store these values.

**B.** Hash the username, hash the password, and store these values.

**C.** Encrypt the username, encrypt the password, and store these values.

**D.** Hash the username, encrypt the password, and store these values.

**ANSWER: A**

## QUESTION NO: 14

An enterprise refactors its monolithic application into a modem cloud-native application that is based on microservices. A key requirement of the application design is to ensure that the IT team is aware of performance issues or bottlenecks in the new application Which two approaches must be part of the design considerations" (Choose two.)

**A.** Periodically scale up the resources of the host machines when the application starts to experience high loads

**B.** Instrument the application code to gather telemetry data from logs, metrics or tracing

**C.** Adopt a service-oriented architecture to handle communication between the services that make up the application

**D.** Deploy infrastructure monitoring agents into the operating system of the host machines

**E.** Implement infrastructure monitoring to ensure that pipeline components interoperate smoothly and reliably

**ANSWER: B E**

## QUESTION NO: 15

What is a characteristic of a monolithic architecture?

**A.** It is an application with multiple independent parts.

**B.** New capabilities are deployed by restarting a component of the application.

**C.** A service failure can bring down the whole application.

**D.** The components are platform-agnostic.

## ANSWER: C

**Explanation:**

Reference: https://medium.com/koderlabs/introduction-to-monolithic-architecture-and-microservices-architecture-b211a5955c63

## QUESTION NO: 16

What is the function of dependency management?

**A.** separating code into modules that execute independently

**B.** utilizing a single programming language/framework for each code project

**C.** automating the identification and resolution of code dependencies

**D.** managing and enforcing unique software version names or numbers

## ANSWER: A

## QUESTION NO: 17 - (DRAG DROP)

Drag and drop the code from the bottom onto the box where the code is missing to retrieve and display wireless network hearth information. The wireless network devices are being managed by Cisco DNA Center and are compatible with REST API Not all options are used.


**ANSWER:**

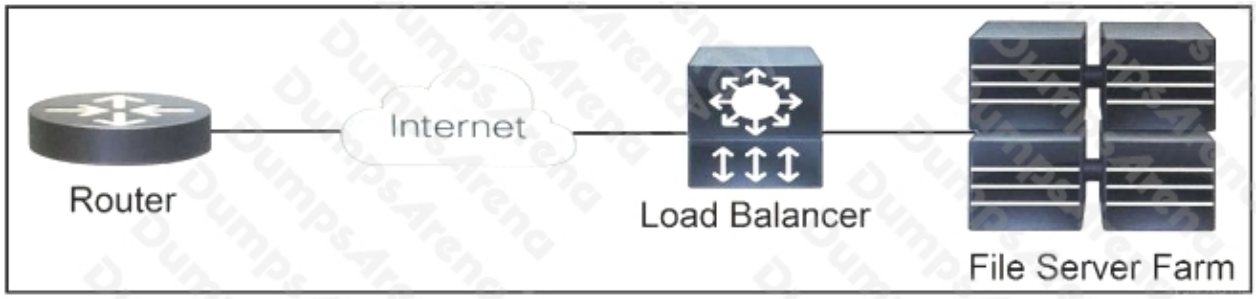

**Explanation:**

```
import requests
import json

BASE_URL = "https://sandboxdnac.cisco.com"
url = f"{BASE_URL}/dna/intent/api/v1/wireless-health"

token = 'eyJhbGc  ...  yJbN8'
headers = {
      'x-auth-token': token,
      'Content-Type': 'application/json',
      'Accept': 'application/json'
}

response = requests.request('GET', url, headers=headers)

networkHealth = json.loads(response.text)

for healthDist in networkHealth["healthDistirubution"]:
    if healthDist[ "category" ] == "Wireless" :
        print(json.dumps(healthDist))
```

api/v1/network-health"

"Distribution"

## QUESTION NO: 18




Refer to the exhibit. Which two functions are performed by the load balancer when it handles traffic originating from the Internet destined to an application hosted on the file server farm? (Choose two.)

**A.** Terminate the TLS over the UDP connection from the router and originate an HTTPS connection to the selected server.

**B.** Terminate the TLS over the UDP connection from the router and originate an HTTP connection to the selected server.

**C.** Terminate the TLS over the TCP connection from the router and originate an HTTP connection to the selected server.

**D.** Terminate the TLS over the TCP connection from the router and originate an HTTPS connection to the selected server.

**E.** Terminate the TLS over the SCTP connection from the router and originate an HTTPS connection to the selected server.

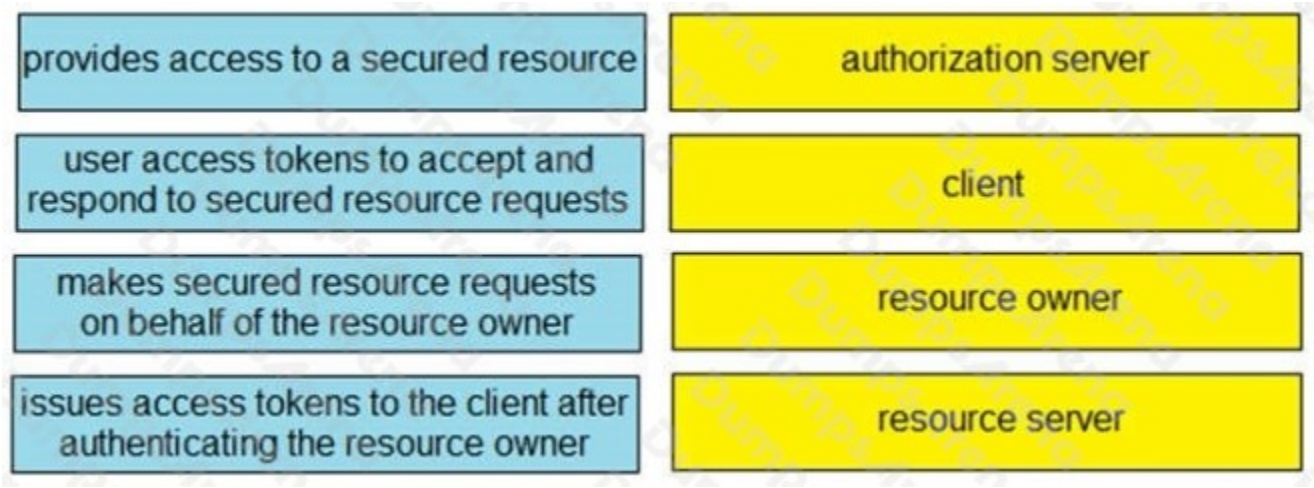
**ANSWER: D E**

## QUESTION NO: 19 - (DRAG DROP)

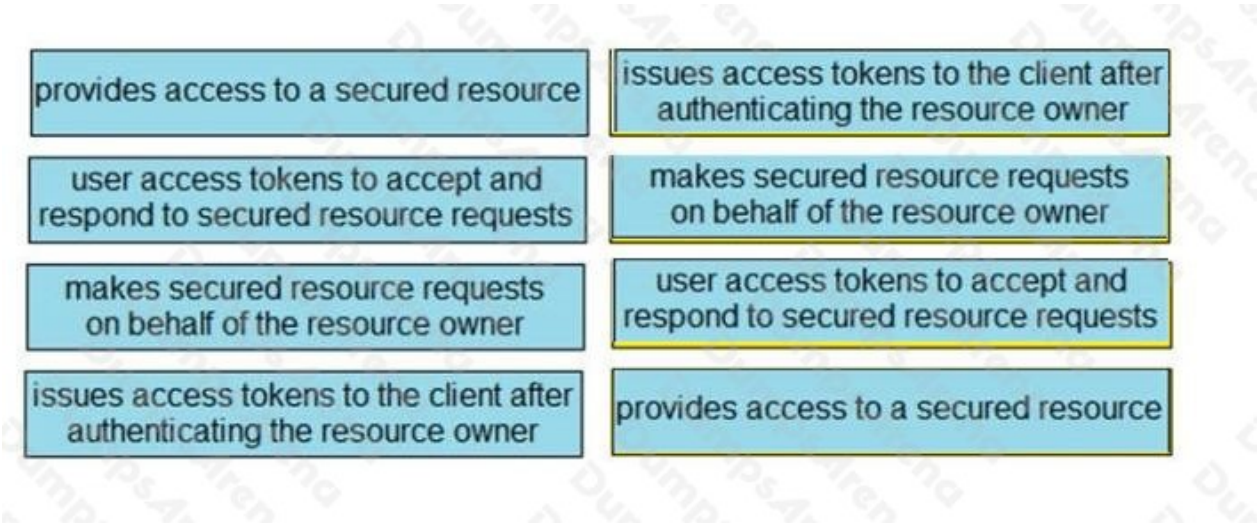
DRAG DROP

Drag and drop the descriptions from the left onto the related OAuth-defined roles on the right.

**Select and Place:**




**ANSWER:**




**Explanation:**

## QUESTION NO: 20

In the three-legged OAuth2 process, after the authorization server presents a form to the resource owner to grant access, what is the next step?

**A.** The resource owner authenticates and optionally authorizes with the authorization server.

**B.** The user who owns the resource initiates a request to the OAuth client.

**C.** If the resource owner allows access, the authorization server sends the OAuth client a redirection.

**D.** A form to allow or restrict access is submitted by the owner of the resource.

**ANSWER: C**

**Explanation:**

"If the resource owner grants access, the authorization server redirects the user's browser back to the client using the redirection URI provided earlier (in the request or during client (registration). The redirection URI includes an authorization code and any local state provided by the client earlier" "Assuming the resource owner grants access, the authorization server redirects the user-agent back to the client using the redirection URI provided earlier (in the request or during client (registration). The redirection URI includes an authorization code and any local state provided by the client earlier."