

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.leads4pass.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

A developer in a learn of distributed authors is working on code in an isolated Git branch named `\\update4a2f5464771`. The developer attempts to merge the new code with a branch named `\\dvcapp2a3f1564733\\` however the merge operation encounters a conflict error during the process. Which Git command must the developer use to exit from the merge process and return the branch to the previous state?

- A. `git merge -exit`
- B. `git merge -abort`
- C. `git abort --merge`
- D. `git abort -status`

Correct Answer: B

QUESTION 2

An application requires SSL certificates signed by an intermediate CA certificate. The crt files must be available to the application:

The root CA certificate is `root_certificate.crt`.

The intermediate CA certificate is `intermediate_certificate.crt`.

The application-specific SSL certificate is `crt_certificate.crt`.

Which Bash command outputs the certificate bundle as a `.pem` file?

- A.

```
cat root_certificate.crt intermediate_certificate.crt > certificate_bundle.pem
```
- B.

```
cat root_certificate.crt intermediate_certificate.crt crt_certificate.crt > certificate_bundle.pem
```
- C.

```
cat crt_certificate.crt intermediate_certificate.crt root_certificate.crt > certificate_bundle.pem
```
- D.

```
cat intermediate_certificate.crt root_certificate.crt > certificate_bundle.pem
```

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

D. Option D

Correct Answer: D

QUESTION 3

AppGigabitEthernet interface is used as data port for a container on a Cisco Catalyst 9000 Series Switch. Which two interface configuration options should be used? (Choose two.)

- A. trunk interface
- B. bridged virtual interface
- C. SPAN port
- D. management interface
- E. subinterface

Correct Answer: AC

QUESTION 4

Refer to the exhibit.

```
$ docker node ls
ID                               HOSTNAME          STATUS
AVAILABILITY                     MANAGER STATUS
cfeae09c992f *                   docker-west-01   Ready
Active                             Reachable
8c15d37dda1f                       docker-west-02   Ready
Active
0683c478497b                       docker-central-01  Ready
Active                             Leader
9a30a613f083                       docker-central-02  Ready
Active
0ac8a7b59d9a                       docker-east-01    Ready
Active                             Reachable
dcd6ecda93d3                       docker-east-02    Ready
Active
```

A Docker swarm cluster is configured to load balance services across data centers in three different geographical regions west central and east. The cluster has three manager nodes and three worker nodes A new service named cisco.devnet is being deployed. The service has these design requirements:

All containers must be hosted only on nodes in the central region The service must run only on nodes that are ineligible for the manager role

Which approach fulfills the requirements?

- A. Create a second swarm cluster that is hosted only in the central region.
- B. Create the service manually in the central region and set replicas to 0.
- C. Use placement constraints to control nodes to which the service can be assigned.
- D. Enable the control flag in the containers of the west and east regions to prevent the service from starting

Correct Answer: D

QUESTION 5

Refer to the exhibit.

```
Started POST "/users/auth/ldapmain/callback" for 172.17.4.98 at 2021-02-11 21:06:47 +0000
2021-02-11_21:06:49.80956 127.0.0.1 - - [11/Feb/2021:21:06:49 UTC] "GET /sidekiq HTTP/1.1" 200 57318
2021-02-11_21:06:49.80959 - -> /sidekiq
{"severity":"ERROR","timestamp":"2021-02-11T21:06:52.813Z","pid":20966,"progname":"omniauth", \
 "message":"(ldapmain) Authentication failure! ldap_error: Net::LDAP::Error, Connection timed out -
 user specified timeout"}
Processing by OmniauthCallbacksController#failure as HTML
Parameters: {"utf8"=>"✓", "authenticity_token"=>"[FILTERED]", "username"=>"user1",
"password"=>"[FILTERED]"}
Redirected to http://192.168.24.55/users/sign_in
Completed 302 Found in 119ms (ActiveRecord: 32.9ms | Elasticsearch: 0.0ms | Allocations: 40775)
{"method":"POST","path":"/users/auth/ldapmain/callback","format":"html","controller":
"OmniauthCallbacksController","action":"failure","status":302, \
 "location":"http://192.168.24.55/users/sign_in","time":"2021-02-11T21:06:52.934Z",
 "params":[{"key":"utf8","value":"✓"}, \
 {"key":"authenticity_token","value":"[FILTERED]"}, {"key":"username","value":"user1"},
 {"key":"password","value":"[FILTERED]"}], \
 "remote_ip":"172.17.4.98","user_id":null,"username":null,"ua":"Mozilla/5.0
(Macintosh; Intel Mac OS X 10_15_7; rv:68.0) Gecko/20100101 Firefox/84.0" \
 "redis_shared_state_read_bytes":109,"redis_shared_state_write_bytes":85,"db_count":5,\
 "db_write_count":1,"db_cached_count":0,"queue_duration_s":0.008279,"cpu_s":0.11,"db_duration_s":0.03294,
"view_duration_s":0.0,"duration_s":0.11892}
```

Recently, users have reported problems logging into an application with their usernames and passwords. The logs have captured an authentication attempt. Based on the messages and errors contained, what is the cause of the problem?

- A. The sign-in redirection is sending clients to the wrong server for SSO.
- B. There is a time synchronization issue between the application and LDAP.
- C. Users are providing incorrect credentials when logging in.
- D. The LDAP server used for authentication fails to respond to connection requests

Correct Answer: A

QUESTION 6

DRAG DROP

Refer to the exhibit. Drag and drop the correct parts of the Dockerfile from the left onto the item numbers on the right that match the missing sections in the exhibit to complete the Dockerfile to successfully build and deploy a container

running a Python application. Not all parts of the Dockerfile are used.

```
<item 1> python:3.6-alpine
<item 2> . .
<item 3> pip install -r requirements.txt
<item 4> 5001
<item 5> ["python", "app.py"]
```

Select and Place:

Answer Area

ENV	<item 1>
CMD	<item 2>
RUN	<item 3>
COPY	<item 4>
VOLUME	<item 5>
FROM	
WORKDIR	
EXPOSE	

Correct Answer:

Answer Area

ENV	FROM
	COPY
	RUN
	EXPOSE
VOLUME	CMD
WORKDIR	

QUESTION 7

DRAG DROP

A developer is creating a Python function that adds network engineers to a Webex room to troubleshoot after a monitoring alert. Drag and drop the code from the bottom onto the box where the code is missing in the Python function to add the engineers to the room. Not all options are used.

Select and Place:

```
def gather_all(token, room_id, teamMembers):  
    for member in teamMembers:  
        header = {"": "Bearer %s" % token,  
                 "Content-Type": "application/json"}  
  
        payload = {  
            "roomId": room_id,  
            "personEmail": member  
        }  
  
        response = requests.post("https://api.webexapi.com/v1/memberships",  
                                headers=header, , verify=True)  
  
        if response. !=  :  
            print("error inviting:" + member)
```

500	code	200
status_code	json=payload	Authorization

Correct Answer:

```
def gather_all(token, room_id, teamMembers):  
    for member in teamMembers:  
        header = {" Authorization": "Bearer %s" % token,  
                 "Content-Type": "application/json"}  
  
        payload = {  
            "roomId": room_id,  
            "personEmail": member  
        }  
  
        response = requests.post("https://api.webexapi.com/v1/memberships",  
                                headers=header,  json=payload, verify=True)  
  
        if response. status_code !=  200 :  
            print("error inviting:" + member)
```

500	code	

QUESTION 8

What is the gRPC Network Management Interface protocol?

- A. a unified management protocol for streaming telemetry and database logging
- B. a configuration management protocol for monitoring
- C. a protocol for configuration management and streaming telemetry

D. a logging protocol used across database servers

Correct Answer: C

Reference: <https://infohub.delltechnologies.com//enterprise-sonic-distribution-by-dell-technologies-lifecyclemanagement/grpc-network-management-interface>

QUESTION 9

Refer to the exhibit.

```
1 import requests
2 url = "https://api.meraki.com/api/v0/networks/2/clients"
3 payload = None
4 headers = {
5     "Content-Type": "application/json",
6     "Accept": "application/json",
7     "X-Cisco-Meraki-API-Key": "1b87b584fb5411eaadc10242ac120002"
8 }
9 response = requests.request('GET', url, headers=headers, data = payload)
10 print(response.text.encode('utf8'))
11
```

A Python script must list network clients in the Cisco Meraki API that have used a network with an ID of 2 The number of client entries per returned page is restricted to 1.000 according to the API specification Network 2 has 2.500 clients. What must be added where the code is missing to print the content of each response?

- A.

```
if response.links.get('next'):
    url = response.links['next']['url']
    response = requests.get(url)
    print(response.text)
```
- B.

```
url = "https://api.meraki.com/api/v0/networks/2/clients?startingAfter=1000"
response = requests.get(url)
print(response.text)
```
- C.

```
url = "https://api.meraki.com/api/v0/networks/2/clients?perPage=1000"
response = requests.get(url)
print(response.text)
```
- D.

```
if response.links.get('next'):
    url = links['url']
    response = requests.get(url)
    print(response.text)
```

A. B. C. D.

Correct Answer: A

QUESTION 10

DRAG DROP

An engineer is developing a web-based application that will be used as a central repository for the HR department. The application needs to authenticate user access and encrypt communication. Drag and drop the steps from the left into the order on the right to install an application-specific SSL certificate.

Select and Place:

Configure the application to use HTTPS.	step 1
Acquire the SSL certificate.	step 2
Install the SSL certificate.	step 3
Create a certificate signing request.	step 4

Correct Answer:

	Create a certificate signing request.
	Acquire the SSL certificate.
	Install the SSL certificate.
	Configure the application to use HTTPS.

QUESTION 11

A developer creates an application for a Cisco Catalyst 9000 switch in a Docker container. Which action must be taken to host the application on the switch?

- A. Copy the application code to a NETCONF file and upload the file to the switch
- B. Connect the switch to Cisco DNA Center and push the application through the platform.
- C. Use the Cisco IOxClient tool to export the application to a ZIP file and push the file to the switch
- D. Export the application as a TAR file and import the file to the switch

Correct Answer: D

QUESTION 12

Refer to the exhibit.

```
headers = ( _____ )
try:
    response = requests.get("https://sandboxnac.cisco.com/dna/intent/api/v1/wireless/profile",
                            headers=headers, verify=False)
except requests.exceptions.RequestException as cerror:
    print("Error processing request", cerror)
    sys.exit(1)
```

Which code snippet is required in the headers to successfully authorize wireless information from Cisco DNA Center?

- A. headers = { 'X-auth-token': 'fa8426a0-8eaf-4d22-8e13-7c1b16a9370c' }
- B. headers = { 'Authorization': 'Basic YWRtaW46R3JhcGV2aW5IMQ==' }
- C. headers = { 'Authorization': 'Bearer ASDNFALKJER23412RKDALSNKF' }
- D. headers = { 'Content-type': 'application/json' }

Correct Answer: A

QUESTION 13

DRAG DROP

A paginated endpoint in the API accepts in 3 special query parameters:

- `perPage`: The number of entries to be returned in the page (the current request)
- `startingAfter`: A token used by our server to indicate the starting "identifier" of the page (i.e. the data we return in this request will start immediately after the entry with this "identifier")
- `endingBefore`: A token used by our server to indicate the ending "identifier" of the page (i.e. the data we return in this request will end immediately before the entry with this "identifier")

The actual types of the `startingAfter` and `endingBefore` identifiers will vary depending on the API endpoint. However, they typically fall into 2 categories:

- **Timestamps**: The values of `startingAfter` and `endingBefore` are timestamps if we're paginating based on time. In other words, each entry returned in the response has some timestamp value associated with it, and each request returns a fixed number of these entries based on the value of the `perPage` parameter. We use timestamps as the "boundaries" between pages.
 - For example, the current page might contain entries with timestamps ranging from exactly 2 days ago to exactly 1 day ago. The previous page might be referred to by { `endingBefore: <2 days ago>` }, and the next page might be referred to by { `startingAfter: <1 day ago>` }.
 - For example, the current page might contain 5 entries with integer IDs ranging from 101 to 105 inclusive. The previous page might be referred to by { `endingBefore: 101` }, and the next page might be referred to by { `startingAfter: 105` }.

Refer to the exhibit Drag and drop the code from the bottom onto the box where the code is missing to query the last 10 Bluetooth clients seen by APs in their network using the Meraki Dashboard API Not all options are used.

Select and Place:

```
import requests
URL = 'https://api.meraki.com/api/v0/networks/NETWORK/'
r = requests. [ ] (URL + ' [ ] ?
  perPage= [ ] =0')
print( [ ] )
```

get	10fendingBefore	clients.mac
bluetoothClients	r.text	bluetooth

Correct Answer:

```
import requests
URL = 'https://api.meraki.com/api/v0/networks/NETWORK/'
r = requests. [ ] (URL + ' [ ] ?
  perPage= [ ] =0')
print( [ ] )
```

[]	[]	clients.mac
[]	r.text	bluetooth

QUESTION 14

```
{
  "version": "3.0",
  "secret": "supersecret",
  "type": "WiFi",
  "data": {
    "networkId": "L 000000000000391274",
    "observations": [
      {
        "locations": [],
        "ipv4": null,
        "ssid": null,
        "os": null,
        "": "cc:cc:66:58:85:23",
        "latestRecord": [
          {
            "time": "2020-10-19T10:23:21z",
            "nearestApMac": "aa:aa:22:56:2e:42",
            "nearestApRssi": "-62"
          }
        ]
      }
    ]
  }
}
```

Refer to the exhibit. The JSON response is received from the Meraki location API. Which parameter is missing?

- A. apMac
- B. clientMac
- C. clientId
- D. accesspoint

Correct Answer: B

Reference: <https://community.meraki.com/t5/Developers-APIs/Location-lat-lng-and-x-y-are-showing-similar-for-all-devices/td-p/65707>

QUESTION 15

DRAG DROP

Drag and drop the code from the bottom onto the box where the code is missing to create a new IKEv2 policy. Not all options are used.

Select and Place:

```

import requests, json

BASE_URL = 'https://ftd.example.com/api/fdm/latest'
url = 
token = 'eyJhbGc ... yJbN8'

payload = {
    'name': 'DEVNET_IKEV2',
    'enabled': True,
    'priority': 0,
    'type': ,
    'encryptionTypes': ['AES_GCM256', 'AES256']
}

headers = {
    'Authorization': ,
    'Content-Type': 'application/json'
}

response = requests.request(, url,
                            headers=headers, data=json.dumps(payload))
print(response.text.encode('utf8'))

```

<code>f'Bearer {token}',</code>	<code>'ikevtwopolicy'</code>
<code>'ikevtwopolicies'</code>	<code>f'{BASE_URL}/object/ikev2policy'</code>
<code>f'{BASE_URL}/object/ikev2policies'</code>	<code>"POST"</code>
<code>f'Basic {token}',</code>	

Correct Answer:

```

import requests, json

BASE_URL = 'https://ftd.example.com/api/fdm/latest'
url = f'{BASE_URL}/object/ikev2policies'
token = 'eyJhbGc ... yJbN8'

payload = {
    'name': 'DEVNET_IKEV2',
    'enabled': True,
    'priority': 0,
    'type': 'ikevtwopolicy'
    'encryptionTypes': ['AES_GCM256', 'AES256']
}

headers = {
    'Authorization': f'Bearer {token}',
    'Content-Type': 'application/json'
}

response = requests.request("POST", url,
                             headers=headers, data=json.dumps(payload))
print(response.text.encode('utf8'))

```

'ikevtwopolicies'	f'{BASE_URL}/object/ikev2policy'
f'Basic {token}',	

[Latest 350-901 Dumps](#)

[350-901 Exam Questions](#)

[350-901 Braindumps](#)