

# 100% Pass Quiz 2026 The Best 300-535: Automating and Programming Cisco Collaboration Solutions Reliable Test Braindumps



2026 Latest ValidVCE 300-535 PDF Dumps and 300-535 Exam Engine Free Share: <https://drive.google.com/open?id=1MTms7-3qomQJs0OOv7C9VZWWQRoXI6hx>

ValidVCE knows the importance of the Cisco 300-535 certification exam in the field of information technology. That is why it has prepared the remarkable Cisco 300-535 exam questions to help the aspirants pass it on the first go. The desiring candidates for the Cisco 300-535 certificate need help to find reliable 300-535 Exam Questions study material.

## Cisco 300-535 Exam Certification Details:

|                     |                                |
|---------------------|--------------------------------|
| Sample Questions    | Cisco 300-535 Sample Questions |
| Exam Price          | \$300 USD                      |
| Number of Questions | 55-65                          |
| Duration            | 90 minutes                     |
| Exam Code           | 300-535 SPAUTO                 |

>> 300-535 Reliable Test Braindumps <<

## 300-535 Latest Version | Valid Dumps 300-535 Ebook

In order to meet the demand of all customers and protect your machines network security, our company can promise that our 300-535 test training guide have adopted technological and other necessary measures to ensure the security of personal information they collect, and prevent information leaks, damage or loss. In addition, the 300-535 exam dumps system from our company can help all customers ward off network intrusion and attacks prevent information leakage, protect user machines network security. If you choose our 300-535 study questions as your study tool, we can promise that we will try our best to enhance the safety guarantees and keep your information from revealing, and your privacy will be protected well. You can rest assured to buy the 300-535 exam dumps from our company.

## Cisco Automating and Programming Cisco Collaboration Solutions Sample Questions (Q100-Q105):

### NEW QUESTION # 100

```
#!/usr/bin/env python

from ydk.models.openconfig.openconfig_interfaces import Interfaces
from ydk.errors import YError

def read_interfaces(crud_service, provider):

    intf_f = Interfaces()

    try:
        interfaces = crud_service.read(provider, intf_f)
        for interface in interfaces.interface:
            print(interface.name)
    except YError:
        print('An error occurred.')
```

Refer to the exhibit. When YDK is used to interact with Cisco routers, what is the purpose of passing `intf_f` into the `crud_service.read()` method?

- A. It locks the interfaces from modification by other active NETCONF sessions.
- B. It provides the data types of the `openconfig:interfaces` model to the router for dynamic configuration of the interfaces.
- C. The `Interfaces()` class acts as a NETCONF filter, which limits the data returned to that of the `openconfig:interfaces` YANG model.
- D. It passes default values into the `crud_service`, which reconfigures all interfaces to their default configurations.

**Answer: D**

Explanation:

Section: Automation and Orchestration Platforms

### NEW QUESTION # 101

Refer to the exhibit. An engineer is configuring a Cisco IOS XR device to function as a Cisco SR- PCE. The engineer must ensure that the Cisco SR-PCE has access to the routing information from the core devices on the network. Which code snippet must be added to the box in the code?

```

PCE Router
pce address
  ipv4 4.4.4.4

router bgp 1
  bgp router-id 4.4.4.4
  address-family ipv4 unicast
  table-policy fib

  neighbor-group epn
  remote-as 1
  update-source Loopback0
  address-family ipv4 labeled-unicast
  route-reflector-client

  neighbor 3.3.3.112
  use neighbor-group epn

1 CORE DEVICE
2 router bgp 1
3   bgp router-id 3.3.3.122
4
5   neighbor-group epn
6   remote-as 1
7   update-source Loopback0
8   address-family ipv4 labeled-unicast
9     route-reflector-client next-hop-self
10
11   neighbor 4.4.4.4
12     remote-as 1
13     update-source Loopback0
14     address-family ipv4 labeled-unicast
15
16
17 router isis 2
18   is-type level-2-only
19   net 49.0001.0000.0000.0122.00
20   distribute link-state instance-id 2
21   address-family ipv4 unicast
22     segment-routing mpls

```

- A. pce state-sync ipv4 3.3.3.112
- B. distribute link-state instance-id 1
- C. distribute bgp-ls instance-id 1
- D. address-family link-state link-state

**Answer: B**

Explanation:

The command distribute link-state instance-id 1 enables the advertisement of link-state information, allowing the SR-PCE to access the routing information from the core devices in the network.

### NEW QUESTION # 102

Drag and Drop Question

The YANG BGP service model is used to configure one or multiple loopback interfaces on routers in a network for iBGP and eBGP sessions. Drag and drop the code snippets from the bottom onto the blanks in the code to ensure that IP addresses are unique on individual routers and network-wide. Not all options are used. The exhibit does not show the entire code.

```

1 module bgp {
2   namespace "http://example.com/bgp";
3   prefix bgp;
4   import ietf-inet-types { prefix inet; }
5   import tailf-common { prefix tailf; }
6   import tailf-ncs { prefix ncs; }
7
8   list bgp {
9     key device;
10    uses ncs:service-data;
11    ncs:servicepoint bgp-servicepoint;
12    leaf device {
13      type leafref { path "/ncs:devices/ncs:device/ncs:name"; }
14    }
15    list loopback {
16      key "intf_number";
17
18      unique " [redacted] ";
19      leaf intf_number {
20        type uint32;
21      }
22      leaf ip {
23        type inet:ipv4-address;
24      }
25      must " [redacted] ";
26    }
27  }
28 }
29 }

```

|                                                                         |
|-------------------------------------------------------------------------|
| ip                                                                      |
| bgp/ip                                                                  |
| loopback/ip                                                             |
| count (/bgp:bgp[device=current()]/.././device]/loopback/ip=current())=3 |
| count (/bgp:bgp[device=current()]/.././device]/loopback/ip=current())=1 |

Answer:

Explanation:

```

1 module bgp {
2   namespace "http://example.com/bgp";
3   prefix bgp;
4   import ietf-inet-types { prefix inet; }
5   import tailf-common { prefix tailf; }
6   import tailf-ncs { prefix ncs; }
7
8   list bgp {
9     key device;
10    uses ncs:service-data;
11    ncs:servicepoint bgp-servicepoint;
12    leaf device {
13      type leafref { path "/ncs:devices/ncs:device/ncs:name"; }
14    }
15    list loopback {
16      key "intf_number";
17
18      unique ip
19      leaf intf_number {
20        type uint32;
21      }
22      leaf ip {
23        type inet:ip-address;
24
25        must -count (/bgp:bgp[device!=current()../../device]/loopback/ip=current())=0
26      }
27    }
28  }
29 }

```



The image shows the Cisco logo and three configuration snippets. The first snippet is 'bgp/ip', the second is 'loopback/ip', and the third is 'count (/bgp:bgp[device!=current()../../device]/loopback/ip=current())=1'. A large 'CISCO' watermark is overlaid on the snippets.

Explanation:

First box (unique): ip

Second box (must): count (/bgp:bgp[device!=current()../../device]/loopback/ip=current())=0 The unique ip; statement ensures each IP is unique within the loopback list for a device.

The must statement enforces that the loopback IP does not appear on any other device, making it unique network-wide.

### NEW QUESTION # 103

Which method maps MPLS EXP bit 5 to COS 5 on Cisco IOS XE?

```

configure terminal
ip access-list 101 permit ip any any mpls experimental 5
class-map match exp
match access-group 101
exit
policy-map EXP2Cos
class exp
set cos 5
exit
class class-default
random-detect
interface fastethernet 0/0
service-policy output EXP2Cos
exit

```

- A.

```
configure terminal
```

```
class-map match exp  
match mpls experimental topmost 5  
exit  
policy-map EXP2Cos  
class exp  
set cos 5  
exit  
class class-default  
random-detect
```

```
interface fastethernet 0/0
```

- B. service-policy input EXP2Cos

```
configure terminal  
class-map match exp  
match mpls cos 5  
exit  
policy-map EXP2Cos  
class exp  
set mpls experimental topmost 5  
exit  
class class-default  
random-detect  
  
interface fastethernet 0/0  
service-policy output EXP2Cos  
exit  
commit
```

- C.

```
configure terminal  
class-map match exp  
match mpls experimental topmost 5  
exit  
policy-map EXP2Cos  
class exp  
set cos 5  
exit  
class class-default  
random-detect  
  
interface fastethernet 0/0  
service-policy output EXP2Cos
```

- D.

```
configure terminal  
class-map match exp  
match mpls cos 5  
exit  
policy-map EXP2Cos  
class exp  
set mpls experimental topmost 5  
exit  
class class-default  
random-detect  
  
interface fastethernet 0/0  
service-policy output EXP2Cos  
exit  
commit
```

- E.

**Answer: D**

**NEW QUESTION # 104**

Refer to the exhibit.



