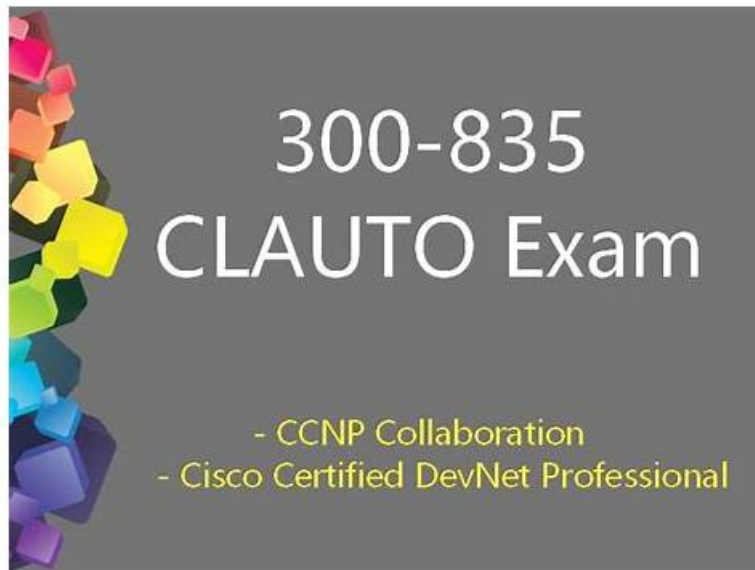


Pass Guaranteed 2026 Professional Cisco New 300-835 Test Format



2026 Latest GetValidTest 300-835 PDF Dumps and 300-835 Exam Engine Free Share: <https://drive.google.com/open?id=1qA5LHsYRBq2G2xFGADuar6Y5mrqEsla->

GetValidTest offers you the best practice tests for the preparation of 300-835 exams. The practice tests are designed to provide you the type of questions you are going to face in real 300-835 exams. The “simulated” real 300-835 exam scenario, created in the practice exam software, is meant to make you familiar with the actual 300-835 Exam. 300-835 announce several changes. Through one year, in their 300-835 exams according to the updated technologies. Make sure to purchase the most recent and updated version of 300-835 certification practice exam for best preparation of 300-835 exam.

Cisco 300-835 certification exam covers a variety of topics such as programming concepts, Cisco Webex APIs, Python programming, Cisco Unified Communications Manager (CUCM) APIs, and automation workflows. 300-835 exam is designed for network engineers, software developers, collaboration engineers, and other IT professionals who want to develop their skills in automating Cisco collaboration solutions. Passing 300-835 exam will entitle individuals to attain the Cisco Certified DevNet Specialist – Collaboration Automation and Programmability certification, which will demonstrate their proficiency in automating and programming Cisco collaboration solutions.

Exam Details

Cisco 300-835 is a 90-minute exam and the individuals have to answer all the questions during this allocated time. There will be about 60 of them, so you need to be ready. The items can be expected in different formats, including fill-in-the-blank, drag and drop, testlet, and multiple choice. This test is available in English only and the candidates must pay \$300.

The Cisco exams are registered and scheduled through Pearson VUE. This platform offers two delivery options, and the individuals can choose between online testing and a proctored exam. With the first option, you can have the test at any place, while a proctored exam should be taken at one of the Cisco testing centers.

>> New 300-835 Test Format <<

Exam 300-835 Testking - Online 300-835 Bootcamps

Do you still have doubts about the quality of the Cisco 300-835 product? No worries. Visit GetValidTest and download a free demo of Cisco Certification Exams for your pre-purchase mental satisfaction. Moreover, the Cisco 300-835 product of GetValidTest is available at an affordable price.

Cisco Automating and Programming Cisco Collaboration Solutions Sample

Questions (Q86-Q91):

NEW QUESTION # 86

A software engineer must configure a Cisco TelePresence device and custom reporting dashboard to retrieve and display the temperature of the device every minute. Drag and drop the code snippets from the bottom onto the boxes in the code to collect and send the data by using the xAPI. Not all options are used.

Select and Place:
Answer Area

```
import requests
import xml.etree.ElementTree as ET
import sched, time
s = sched.scheduler(time.time, time.sleep)
def main(sc):
    url = "https://192.168.159.12/[ ]"
    payload={}
    headers = { 'Authorization': 'Basic RtaWec25pdHJvZ2VvI0==' }
    response = requests.request("[ ]", url, headers=headers, data=payload,
    verify=False)
    tree = ET.fromstring(response.text)
    current_temp = tree.find('SystemUnit').find('Hardware').find('[ ]')
    threshold_temp = tree.find('SystemUnit').find('Hardware').find('Temperature
    Threshold')
    if float(current_temp.text) > float(threshold_temp.text):
        print("Temperature too high.")
        (60, 1, main, (sc,))
    [ ]
s.enter(60, 1, main, (s,))
s.run()
```

| | | | |
|------|--------|-------------|------------|
| GET | status | s.enter | status.xml |
| POST | Bearer | Temperature | |

Answer:

Explanation:

Answer Area

```
import requests
import xml.etree.ElementTree as ET
import sched, time
s = sched.scheduler(time.time, time.sleep)
def main(sc):
    url = "https://192.168.159.12/[ status.xml ]"
    payload={}
    headers = { 'Authorization': 'Basic RtaWec25pdHJvZ2VvI0==' }
    response = requests.request("[ GET ]", url, headers=headers, data=payload,
    verify=False)
    tree = ET.fromstring(response.text)
    current_temp = tree.find('SystemUnit').find('Hardware').find('[ Temperature ]')
    threshold_temp = tree.find('SystemUnit').find('Hardware').find('Temperature
    Threshold')
    if float(current_temp.text) > float(threshold_temp.text):
        print("Temperature too high.")
        (60, 1, main, (sc,))
    [ s.enter ]
s.enter(60, 1, main, (s,))
s.run()
```

| | | | |
|------|--------|-------------|------------|
| GET | status | s.enter | status.xml |
| POST | Bearer | Temperature | |

NEW QUESTION # 87

Refer to the exhibit. This Python script uses the websocket-based, xAPI library 'pyxows' to monitor and print event details to the console when users interact with UI Extension/In-Room Control 'Action Buttons' deployed to the Touch-10 interface of a Webex room device.

Which two code snippets successfully capture such events? (Choose two.)

```

import xows
import asyncio

async def start(ip, usr, pw):
    async with xows.XoWSClient(ip, username=usr, password=pw) as client:
        async def callback(data, id_):
            print(f'Feedback {id} (Id {id}): {data}')

        await client.wait_until_closed()

async def task():
    await start('10.10.10.1', 'admin', 'T357c45e')

asyncio.run(task())

```

- A. `await client.subscribe(['Event', 'UserInterface', 'Extensions', 'Panel', 'Clicked'], callback)`
- B. `await client.xEvent(['Event', '**'], callback)`
- C. `await client.xEvent(['Event', 'UserInterface', 'Extensions', 'Panel', 'Clicked'], callback)`
- D. `await client.subscribe(['Event', 'ActionButton', 'Clicked'], callback)`
- E. `await client.subscribe(['Event', '**'], callback)`

Answer: A,C

NEW QUESTION # 88

Drag and drop the code to add a Webex Teams Space Widget in an existing HTML page. Not all items are used.

Answer Area

```

<body>
  <div id="my-webexteams-widget" style="width: 500px; height: 500px;" />
  <script>
    <var widgetEl = document.getElementById(' ');
    // Init a new widget
    ciscopark.widget(widgetEl).  ( {
      accessToken: 'ACCESS_TOKEN',
      destinationId: 'SPACE_ID',
      destinationType: ' ',
    });
  </script>
</body>

```

my-webexteams-widget

spaceId

mySpaceWidget

webexteams-widget

space_ID

spaceWidget

RoomId

PersonEmail

Answer:

Explanation:

Answer Area

```
<body>
  <div id="my-webexteams-widget" style="width: 500px; height: 500px;" />
  <script>
    <var widgetEl = document.getElementById(' my-webexteams-widget ');
    // Init a new widget
    ciscopark.widget(widgetEl). spaceWidget      ((
      accessToken: 'ACCESS_TOKEN',
      destinationId: 'SPACE_ID',
      destinationType: ' spaceId '
    ));
  </script>
</body>
```

my-webexteams-widget spaceId mySpaceWidget

webexteams-widget space_ID spaceWidget

RoomId PersonEmail

NEW QUESTION # 89

Drag and Drop Question

Refer to the exhibit. Drag and drop the code snippets from the bottom onto the blanks in the code to construct a Python script for Cisco UCM that provisions directory numbers. Not all options are used.

element addLine

| | | | | | | |
|------------|---|------------------|----------|---------|-------|------------|
| namespace | http://www.cisco.com/AXL/API/12.5 | | | | | |
| type | axlapi:AddLineReq | | | | | |
| properties | content complex | | | | | |
| children | line | | | | | |
| attributes | Name | Type | Use | Default | Fixed | Annotation |
| | sequence | xsd:unsignedLong | optional | | | |
| source | <xsd:element name="addLine" type="axlapi:AddLineReq"> | | | | | |

Answer Area

```
from requests import   
from requests.auth import HTTPBasicAuth  
from zeep import Client, Settings  
from zeep.transports import Transport  
from zeep.cache import SqliteCache  
from zeep.exceptions import Fault  
session = Session()  
session.verify = False  
session.auth =  ('admin', 'password')  
transport = Transport(session=session, timeout=10, cache=SqliteCache())  
settings = Settings(strict=False, xml_huge_tree=True)  
client = Client('AXLAPI.wsdl', settings=settings, transport=transport)  
service = client.create_service(  
    "{http://www.cisco.com/AXLAPIService/}AXLAPIBinding",  
    "https://host.com:8443/axl/"  
)  
data = {  
    'line': {  
        'pattern': '1111',  
        'description': 'Test Line',  
        'usage': 'Device',  
        'routePartitionName': None  
    }  
}  
 (**data)
```

service.line

Session

HTTPBasicAuth

HTTPBearer

Requests

service.addLine

Answer:

Explanation:

Answer Area

```

from requests import Session
from requests.auth import HTTPBasicAuth
from zeep import Client, Settings
from zeep.transports import Transport
from zeep.cache import SqliteCache
from zeep.exceptions import Fault
session = Session()
session.verify = False
session.auth = HTTPBasicAuth('admin', 'password')
transport = Transport(session=session, timeout=10, cache=SqliteCache())
settings = Settings(strict=False, xml_huge_tree=True)
client = Client('AXLAPI.wsdl', settings=settings, transport=transport)
service = client.create_service(
    "(http://www.cisco.com/AXLAPIService/)AXLAPIBinding",
    "https://host.com:8443/axl/"
)
data = {
    'line': {
        'pattern': '1111',
        'description': 'Test Line',
        'usage': 'Device',
        'routePartitionName': None
    }
}
service.addLine(**data)

service.line
HTTPBearer Requests

```

Explanation:

Session is used for handling HTTP connections with authentication.

HTTPBasicAuth provides credentials to the UCM.

service.addLine calls the correct AXL method for provisioning a directory number (DN) in Cisco Unified Communications Manager.

NEW QUESTION # 90

Drag and drop the code to add a Webex Teams Space Widget in an existing HTML page. Not all items are used.

Answer Area

```

<body>
  <div id="my-webexteams-widget" style="width: 500px; height: 500px;"/>
  <script>
    <var widgetEl = document.getElementById(' ');
    // Init a new widget
    ciscopark.widget(widgetEl). (
      accessToken: 'ACCESS_TOKEN',
      destinationId: 'SPACE_ID',
      destinationType: ' ',
    );
  </script>
</body>

```

my-webexteams-widget spaceId mySpaceWidget

webexteams-widget space_ID spaceWidget

RoomId PersonEmail

Answer:

BONUS!!! Download part of GetValidTest 300-835 dumps for free: <https://drive.google.com/open?id=1qA5LHsYRBq2G2xFGADuar6Y5mrqEsla->