

Cheat Sheet for comprehensive Cisco DevNet Certifications - DevNet Specialist

****1. Introduction to Cisco DevNet****

- **DevNet Overview:**

- Cisco's platform for developers to build, test, and deploy network automation solutions.
- Focuses on APIs, SDKs, sandboxes, and community resources.

- **Key Components:**

- **DevNet Sandboxes:** Free, remote access to live Cisco environments.
 - **DevNet Learning Labs:** Hands-on tutorials and exercises.
 - **DevNet Code Exchange:** Open-source projects and code samples.
-

****2. Core Concepts****

- **Network Programmability:**

- **RESTful APIs:** Representational State Transfer APIs for CRUD operations.
- **NETCONF:** Network Configuration Protocol for managing network devices.
- **YANG:** Data modeling language for NETCONF.
- **gRPC:** High-performance RPC framework.

- **Automation Tools:**

- **Ansible:** Agentless automation tool for configuration management.
- **Puppet:** Infrastructure as code tool for managing configurations.
- **Chef:** Configuration management tool for infrastructure automation.
- **Terraform:** Infrastructure as code tool for provisioning resources.

- **Programming Languages:**

- **Python:** Primary language for network automation.
- **JavaScript:** Used in web-based automation tools.

- **Bash/Shell:** For scripting and CLI automation.
-

****3. Cisco APIs****

- Cisco DNA Center API:

- **Endpoints:** `/dna/intent/api/v1/`
- **Authentication:** OAuth 2.0 with client credentials.
- **Common Operations:**
 - `GET /network-device`: List network devices.
 - `POST /command/central/script/deploy`: Deploy scripts.

- Cisco IOS XE API:

- **RESTCONF:** RESTful API for IOS XE devices.
- **Endpoints:** `/restconf/data/`
- **Authentication:** Basic Auth or OAuth 2.0.
- **Common Operations:**
 - `GET /restconf/data/ietf-interfaces:interfaces`: List interfaces.
 - `PUT /restconf/data/ietf-interfaces:interfaces/interface=GigabitEthernet1`: Configure interface.

- Cisco ACI API:

- **Endpoints:** `/api/`
 - **Authentication:** Basic Auth or Token-based.
 - **Common Operations:**
 - `POST /api/aaaLogin.json`: Authenticate.
 - `GET /api/node/class/fvTenant.json`: List tenants.
-

****4. Python for Network Automation****

- Python Libraries:

- **requests:** HTTP library for API calls.

- `ncclient`: NETCONF client for Python.
- `pyATS`: Cisco's test automation framework.
- `netmiko`: Multi-vendor library for SSH connections.

- Example: Fetching Device Info via RESTCONF

```
import requests
from requests.auth import HTTPBasicAuth

url = "https://device/restconf/data/ietf-interfaces:interfaces"
headers = {
    "Accept": "application/yang-data+json",
    "Content-Type": "application/yang-data+json"
}
auth = HTTPBasicAuth('username', 'password')

response = requests.get(url, headers=headers, auth=auth,
verify=False)
print(response.json())
```

- Example: Configuring an Interface via NETCONF

```
from ncclient import manager

config = """
<config>
  <interfaces xmlns="urn:ietf:params:xml:ns:yang:ietf-interfaces">
    <interface>
      <name>GigabitEthernet1</name>
      <description>Configured via NETCONF</description>
      <enabled>true</enabled>
    </interface>
  </interfaces>
</config>
"""

with manager.connect(host='device', port=830, username='admin',
password='password', hostkey_verify=False) as m:
    response = m.edit_config(target='running', config=config)
    print(response)
```

****5. Ansible for Network Automation****

- Ansible Modules:

- ``ios_config``: Configure Cisco IOS devices.
- ``nxos_config``: Configure Cisco Nexus devices.
- ``aci_tenant``: Manage Cisco ACI tenants.
- ``dnac_command_runner``: Run commands on Cisco DNA Center.

- Example: Ansible Playbook for Configuring an Interface

```
- name: Configure Interface on Cisco IOS
  hosts: ios
  connection: network_cli
  gather_facts: no

  tasks:
    - name: Configure GigabitEthernet1
      ios_config:
        lines:
          - description Configured via Ansible
          - no shutdown
        save_when: modified
```

- Example: Ansible Playbook for Managing ACI Tenants

```
- name: Manage ACI Tenants
  hosts: aci
  connection: local
  gather_facts: no

  tasks:
    - name: Create a new tenant
      aci_tenant:
        host: "{{ aci_host }}"
        username: "{{ aci_username }}"
        password: "{{ aci_password }}"
        tenant: "{{ tenant_name }}"
        state: present
```

****6. DevNet Sandboxes****

- Accessing Sandboxes:

- **URL:** [DevNet Sandbox](https://devnetsandbox.cisco.com/)
- **Reservation:** Reserve a sandbox for a specific time period.
- **Types:**
 - **Always-On:** Available 24/7.
 - **Reservation-Based:** Requires booking.

- Common Sandboxes:

- **Cisco DNA Center:** For network automation and orchestration.
 - **Cisco IOS XE:** For testing RESTCONF and NETCONF.
 - **Cisco ACI:** For APIC-based automation.
 - **Cisco SD-WAN:** For vManage API testing.
-

****7. DevNet Learning Labs****

- Accessing Learning Labs:

- **URL:** [DevNet Learning Labs](https://developer.cisco.com/learning/)
- **Categories:**
 - **Network Programmability:** Basics of network automation.
 - **Cisco DNA Center:** Automation with DNA Center.
 - **Cisco ACI:** Automation with APIC.
 - **Cisco SD-WAN:** Automation with vManage.

- Example: Learning Lab Workflow

- **Step 1:** Select a lab from the catalog.
- **Step 2:** Follow the guided instructions.
- **Step 3:** Complete exercises and quizzes.
- **Step 4:** Earn badges and certificates.

****8. DevNet Code Exchange****

- **Accessing Code Exchange:**
 - **URL:** [DevNet Code Exchange](https://developer.cisco.com/codeexchange/)
 - **Types of Projects:**
 - **Sample Code:** Ready-to-use scripts.
 - **Open-Source Tools:** Community-driven projects.
 - **SDKs:** Software Development Kits for Cisco products.
- **Example: Using a Sample Code**
 - **Step 1:** Browse the repository.
 - **Step 2:** Clone the repository.
 - **Step 3:** Follow the README for setup instructions.
 - **Step 4:** Run the code in your environment.

****9. Tips and Tricks****

- **API Authentication:**
 - **OAuth 2.0:** Use client credentials flow for machine-to-machine communication.
 - **Basic Auth:** Use for simple API calls.
 - **Token Management:** Store tokens securely and refresh as needed.
- **Error Handling:**
 - **HTTP Status Codes:** Check for `200 OK`, `400 Bad Request`, `401 Unauthorized`, etc.
 - **Exception Handling:** Use `try-except` blocks in Python.
- **Version Control:**
 - **Git:** Use Git for version control of your automation scripts.
 - **GitHub/GitLab:** Host your repositories for collaboration.
- **Documentation:**

- **Swagger/OpenAPI:** Use for documenting REST APIs.
- **YANG Models:** Use for documenting NETCONF/RESTCONF APIs.

****10. Resources****

- **Cisco DevNet Documentation:**

- **URL:** [DevNet Documentation](<https://developer.cisco.com/docs/>)
- **Topics:** API references, SDKs, sample code, and tutorials.

- **Cisco Community Forums:**

- **URL:** [DevNet Community](<https://community.cisco.com/t5/devnet/ct-p/53-devnet>)
- **Topics:** Ask questions, share knowledge, and collaborate.

- **Cisco Learning Network:**

- **URL:** [Learning Network](<https://learningnetwork.cisco.com/>)
- **Topics:** Certification preparation, study groups, and webinars.

This cheat sheet provides a comprehensive overview of essential concepts, tools, and techniques for Cisco DevNet Specialist certification. Use it as a quick reference guide while preparing for the certification or during your daily automation tasks.

By Ahmed Baheeg Khorshid

ver 1.0