

Designing and Implementing Cisco Network Programmability - On Demand (NPDESI 1.0)

Modality: On Demand

Duration: 40 Hours

CLC: 8 Units

This course makes you ready for the Exam of 300-550 NPDESI leading to the Certification of 300-550 NPDESI. No Exam Voucher includes in this course, but you have the option to request the purchase of a Voucher for the Official Exam separately.

About this course:

You will figure out in this course, how to reduce the number of manual interactions with the network and enhance the utilization of contents and automation apparatuses to drive down operational inefficiencies. Within these objectives, you will examine programmability fundamentals of the network, for example, using Python and Linux and normal automation protocols, for example, REST and NETCONF. Understand how the protocols identify with YANG information models and also SDN controller southbound and northbound APIs. Also, you will figure out how to utilize gadget APIs on different platforms of Cisco. Begin using automation tools, for example, Puppet and Ansible.

Course Objective:

- Reduce the number of manual interactions with the system, and increase the utilization of contents and automation apparatuses to drive down inefficiencies of operation.
- Regular protocols of automation, for example, REST and NETCONF
- Examine network programmability basics including Python and Linux.
- YANG data models
- SDN controllers including APIC-EM, APIC, and OSC
- Agile and DevOps software development methodologies
- Device-level APIs, for example, Cisco IOS-XE, NX-OS, ASA OS, and IOS-XR.
- Automation instruments, for example, Chef, Ansible, and Puppet
- Working with JSON Objects in Python
- Using XML in Python
- NETCONF Capabilities Exchange
- Interacting with Cisco ASA REST API
- NX-API Discovery
- Consume NX-API with Python
- Consuming NETCONF on Nexus
- Using Python on the Nexus Switch
- Interacting with the Cisco IOS XE RESTCONF API
- Interacting with the Cisco IOS XE NETCONF API
- Interacting with the Cisco IOS XR NETCONF API
- Securing the Management Plane
- Validating YANG Models Using yangvalidator

- Writing a Custom YANG Model
- Viewing and Validating YANG Models with pyang
- BGP Configuration Change Using ydk-py (with OC BGP Models)
- Generate Python Bindings with ydk-gen
- Navigating YANG-Explorer
- Network Discovery Configuration
- APIC-EM Network Discovery and RBAC
- Consuming the APIC-EM API
- Compliance Checks with Ansible
- Tenant Provisioning with Ansible

Audience:

- Network engineer
- System engineer
- Network designer
- Field engineer
- Network operations
- Network programmer
- Network automation engineer
- Software engineer
- Network developer
- Application developer

Prerequisite:

- Baseline information about Python.
- Datacenter background of CCNA/CCNP-level network

Course Outline:

Course Introduction

- Network Programmability Fundamentals
- APIs and Automation Protocols
- Data Models
- SDN Controllers
- Network Operations

Labs:?

- Challenge 1: Using the Linux Command Line
- Challenge 2: Linux Networking
- Challenge 3: Python Foundations (Part 1)
- Challenge 4: Python Foundations (Part 2)
- Challenge 5: Writing and Troubleshooting Python Scripts
- Challenge 6: Custom Python Libraries

- Discovery 1: Working with JSON Objects in Python
- Discovery 2: Using XML in Python
- Discovery 3: NETCONF Capabilities Exchange
- Challenge 7: Interacting with Cisco ASA REST API
- Discovery 4: NX-API Discovery
- Challenge 8: Consume NX-API with Python
- Challenge 9: Consuming NETCONF on Nexus
- Challenge 10: Using Python on the Nexus Switch
- Challenge 11: Interacting with the Cisco IOS XE RESTCONF API
- Challenge 12: Interacting with the Cisco IOS XE NETCONF API
- Challenge 13: Interacting with the Cisco IOS XR NETCONF API
- Challenge 14: Securing the Management Plane
- Discovery 5: Validating YANG Models Using yangvalidator
- Challenge 15: Writing a Custom YANG Model
- Discovery 6: Viewing and Validating YANG Models with pyang
- Discovery 7: BGP Configuration Change Using ydk-py (with OC BGP Models)
- Discovery 8: Generate Python Bindings with ydk-gen
- Discovery 9: Navigating YANG-Explorer
- Discovery 10: Network Discovery Configuration
- Challenge 16: APIC-EM Network Discovery and RBAC
- Challenge 17: Consuming the APIC-EM API
- Discovery 11: ACI Fabric Discovery
- Discovery 12: Creating Objects with APIC GUI
- Discovery 13: Navigating the Object Model
- Discovery 14: Using API Inspector
- Discovery 15: Using ARYA
- Discovery 16: CLI Emulation
- Discovery 17: ACI Diagram Tool
- Discovery 18: ACI Toolkit—Lint
- Challenge 18: Using the APIC REST API
- Discovery 19: GitHub Pull Request
- Challenge 19: Working with Git
- Discovery 20: DevNet Sandbox
- Discovery 21: DevNet Learning Labs
- Discovery 22: DevNet GitHub
- Challenge 20: Integration Testing
- Challenge 21: Compliance Checks with Ansible
- Challenge 22: Tenant Provisioning with Ansible