



Introducing Automation for Cisco Solutions (CSAU) v1.1

What you'll learn in this course

The Introducing Automation for Cisco Solutions (CSAU) v1.1 course gives you a broad overview of network automation skills. Through a combination of lecture and hands-on labs, you will learn the fundamentals of automation such as working on model-driven programmability solutions with Representational State Transfer Configuration Protocol (RESTCONF) and Network Configuration Protocol (NETCONF) protocols. The course also covers data formats and types, including Extensible Markup Language (XML), JavaScript Object Notation (JSON), Yaml Ain't Markup Language (YAML), and Yet Another Next Generation (YANG), and their value in network automation, along with DevOps tools such as Ansible and Git.

This course is a requirement prior to enrolling in professional-level automation courses and exams because it provides crucial foundational knowledge essential to success:

Automation courses -
exams

Automation

- Implementing Automation for Cisco Enterprise Solutions (ENAU) -
300-435 Automating and Programming Cisco Enterprise Solutions (ENAUTO)
- Implementing Automation for Cisco Data Center Solutions (DCAUI) -
300-635 Automating and Programming Cisco Data Center Solutions (DCAUTO)
- Implementing Automation for Cisco Security Solutions (SAUI) -
300-735 Automating and programming Cisco Security Solutions (SAUTO)
- Implementing Automation for Cisco Service Provider Solutions (SPAUI) -
300-535 Automating and programming Cisco Service Provider Solutions (SPAUTO)
- Implementing Automation for Cisco Collaboration Solutions (CLAUI) -
300-835 Automating and Programming Cisco Collaboration Solutions (CLAUTO)



Who should enroll

This course is designed primarily for customer engineers and systems engineers in the following job roles:

- Automation architect
- Automation engineer
- Consulting systems engineer
- DevOps engineer
- Network administrator
- Network architect
- Network consulting engineer
- Network design engineer
- Network engineer
- Network operator
- Network reliability engineer
- Sales engineer
- Site reliability engineer
- Systems engineer
- Technical solutions architect

This course would also be useful for network manager job roles such as:

- Application developer
- Collaboration developer
- Collaboration solutions architect
- IT director
- Mobile developer
- Network Operations Center (NOC) manager
- Software architect
- Web developer

Technology areas

- Network automation

Course Duration

- Instructor-led training: 3 days
- Virtual instructor-led training: 3 days
- E-learning: Equivalent of 3 days in the classroom

How you'll benefit

This course will help you:

- Gain an overview of the skills you need to become a next-generation engineer
- Prepare to accelerate network automation in your organization
- Increase collaboration across internal and external teams using version control systems

Course details

Objectives

After taking this course, you should be able to:

- Articulate the role network automation and programmability play in the context of end-to-end network management and operations
- Define and differentiate between waterfall and agile software development methodologies
- Interpret and troubleshoot Python scripts with fundamental programming constructs built for network automation use cases
- Describe how DevOps principles, tools, and pipelines can be applied to network operations
- Understand the role of network automation development environments and associated technologies such as Python virtual environments, Vagrant, and Docker
- Understand and construct HTTP-based API calls to network devices
- Articulate the differences among and common use cases for XML, JSON, YAML, and protobuf
- Construct and interpret Python scripts using the Python requests module to automate devices that have HTTP-based APIs
- Understand the role YANG plays in network automation
- Understand that a number of tools exist to simplify working with YANG models
- Describe the functionality of RESTCONF and NETCONF and the differences between them
- Construct Ansible playbooks to configure network devices and retrieve operational state data from them
- Build Jinja2 templates and YAML data structures to generate desired state configurations

Recommended knowledge and training

Before taking this course, you should have the following knowledge and skills:

- Routing and switching including Open Shortest Path First (OSPF), Border Gateway Protocol (BGP), and basic configuration features such as interfaces, Simple Network Management Protocol (SNMP), and static routes
- Fundamentals of Python data structures and programming constructs such as loops, conditionals, and classes, or the equivalent of 3–6 months of experience writing Python scripts
- Basic Linux commands for navigating the file system and executing scripts
- Knowledge of working with text editors

How to enroll

To enroll in the CLFNDU course or explore our larger catalog of courses on Cisco Digital Learning, contact us at <training@fastlane-mea.com>

Outline

- Examining Network Management and Operations
- Exploring Software Development Methodologies
- Using Python for Network Automation
- Describing NetDevOps: DevOps for Networking
- Managing Automation Development Environments
- Introducing HTTP Network APIs
- Reviewing Data Formats and Data Encoding
- Using Python Requests to Automate HTTP-Based APIs
- Exploring YANG
- Using YANG Tools
- Automating Model-Driven APIs with Python
- Introducing Ansible for Network Automation
- Templating Configurations with Jinja2

Lab outline

- Use Network Automation Scripts
- Enforce Python Fundamentals on the Interactive Interpreter
- Automate Networks with Netmiko
- Use the Git Version-Control System and Collaborate on an Internal Project
- Build Reproducible Automation Environments
- Use HTTP-Based APIs with Postman
- Explore YAML and JSON Data
- Consume HTTP-Based APIs with Python Requests
- Explore YANG Tools
- Explore RESTCONF with Python
- Explore NETCONF with Python
- Configure Network Devices with Ansible
- Collect Network Data with Ansible
- Build and Deploy Configurations with Ansible

