

Day 1 — Orientation & Use Cases

- Course goals and outcomes
- Safety and sandbox policies
- Where Python fits vs Ansible
- What you'll build by Day 5

Goals (What You'll Be Able To Do)

- Automate multi-vendor changes safely (idempotent)
- Query and modify state via APIs (RESTCONF/Junos)
- Parse CLI/SNMP/syslog into JSON/SQLite
- Orchestrate fan-out with timeouts and retries

Safety & Scope

- Sandboxes only; no production changes
- Dry-run by default; explicit flag to apply
- Secrets via env vars (`NET_USERNAME` , `NET_PASSWORD`)
- Offline fixtures via `OFFLINE=1`

Tooling Overview

- Python 3.11, venv, pip
- Netmiko, NAPALM, HTTPX/Requests
- Jinja2 templates, sqlite3
- Ansible (check mode + diffs)

Modern NetOps Use Cases

- Configuration management & compliance (diffs)
- Inventory normalization (YAML/CSV → JSON/SQLite)
- Change windows & guardrails
- Event triage (syslog, SNMP)

Python vs Ansible

- Python: custom logic, data pipelines, orchestration
- Ansible: idempotent modules, inventory/vars, check mode/diff
- Hand-offs: generate data in Python → playbooks apply

Devices & Vendors

- Cisco IOS XE / NX-OS; Juniper Junos
- Firewalls (ASA) where driver permits (read-only demos)
- Lab backends: DevNet Sandbox, Juniper vLabs

Repo Layout (You Will Use Today)

- `day1_setup/` basics, inventory bootstrap
- `common/fixtures/` offline data
- `scripts/run_validations.py` quick checks

Today's Deliverables

- Working venv + package checks
- Inventory JSON built from sample CSV/YAML
- Parsed `show` outputs stored as JSON
- Daily quiz (concept checks)

Timeboxes

- Lecture $\leq 40\%$, Labs $\geq 60\%$
- Short breaks + single lunch
- Office hours buffer at end of day