TextFSM vs NAPALM & JSON

- Parsing CLI safely

- When to prefer NAPALM getters

- Normalizing to JSON

TextFSM (Netmiko)

- `use_textfsm=True` for known commands

- Depends on ntc-templates coverage

- Fast, but format-fragile

NAPALM Getters

- `get_facts`, `get_interfaces`, `get_route_to`
- Normalized dicts across vendors
- Prefer for structured read

JSON Handling

- Always pretty-print to files during labs
- Use `.get()` with defaults when parsing

Demo

- Run: `python day2_cli_to_data/01_netmiko_show.py`

- Run: `OFFLINE=1 python day2_cli_to_data/02_napalm_getters.py`

Takeaways

- Use getters where available; fall back to TextFSM carefully

- Persist outputs for validation