

Ansible AD-HOC

```
ansible all -i 81.28.97.16, -m ping
ansible groupeB -i hosts -m shell -a "df -Th" --one-line

# RAW: Lancer cmd sans utiliser python (équivalent ssh + cmd)
ansible all -i 81.28.97.16, -m raw -a "apt-get install tree -s"

# gather_facts avec filtre:
ansible all -i 81.28.97.16, -m setup -a "filter=ansible_dis"
```

Lancer un playbook

```
# Vérifier le code :
ansible-playbook -i hosts install.yml --syntax-check

# Lancer en simulant seulement (dry-run):
ansible-playbook -i hosts install.yml --check

# Voir les différentes tâches :
ansible-playbook -i hosts install.yml --list-tasks

# Lancer uniquement la tâche 'ftp'
ansible-playbook -i hosts install.yml --tags ftp

# Lancer le playbook (toutes les tâches)
ansible-playbook -i hosts install.yml
```

Inventory

INI inventory :

```
[nodes]
192.168.0.15
192.168.0.16
db[0-9].exemple.com
192.168.0.17 ansible_port=22000
[nodes:vars]
ansible_port=22
```

YAML inventory :

```
wordpress:
  hosts:
    81.28.97.16:
      user: node1
      domain: "81.28.97.16"
      user_url: "http://{{ domain }}/"
    81.28.97.17:
      user: node2
      domain: "81.28.97.17"
      user_url: "http://{{ domain }}/"
  vars:
    user_title: "'Mon BLOG'"
    ansible_port=22
```

Playbook basique

```
---
- name: installation serveur web
  hosts: serveurweb
  gather_facts: false
  remote_user: root
  vars:
    user: user1

  tasks:
    - name: Install apache php
      apt:
        pkg:
          - apache2
          - php

    - name: Ajout user
      user:
        name: "{{ user }}"
        password: !
        shell: /bin/false

    - name: Copy virtualhost
      template:
        src: templates/virtualhost.conf.j2
        dest: /etc/apache2/sit.../virtualhost.conf

  handlers:
    - name: relancer Apache
      service:
        name: apache2
        state: reloaded
```

Playbook avancé

Boucle sur items:

```
- name: Delete files
  file:
    path: "{{ item }}"
    state: absent
  with_items:
    - /etc/apache2/si.../virtualhost.conf
    - /var/www/{{ domain }}
```

Conditionals: (register + debug)

```
- name: Remove virtualhost
  command: a2dissite virtualhost.conf
  args:
    removes: /etc/apache2/s../virtualhost.conf

- name: check if wordpress directory exist
  stat:
    path: /var/www/wordpress
  register: wordpress_dir

- debug:
  var: wordpress_dir.stat.exists

- name: mkdir /var/www/wordpress
  file:
    path: /var/www/wordpress
    state: directory
  when: not wordpress_dir.stat.exists

- name: reboot Debian
  command: shutdown -r now
  when: ansible_os_family == "Debian"
```

Roles:

```
roles:
  - serveur
  - mysql
```

```
roles/
  serveur/
    files/
    templates/
    tasks/main.yml
    handlers/main.yml
    vars/main.yml
    meta/main.yml
  mysql/
  ...
```

