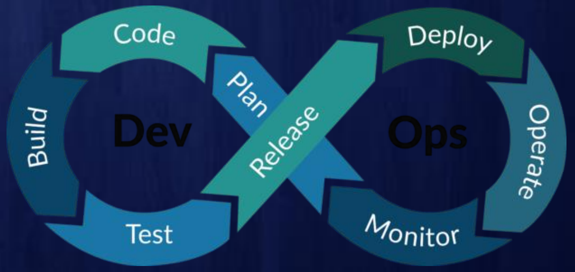
AWS SAA + SysOps + Developer + DevOps Course #Day-67

We will start at 8 AM,
Stay tuned



RAKESH TANINKI
LEARN TO UNLEARN

Recall:

- Ansible
  - Introduction
  - Architecture
  - Installation
  - Inventory
  - Variables
  - Playbooks - basics

Today’s topics:

- Ansible
  - Playbooks
    - Verifying Playbooks
    - Syntax checks
    - Loops
    - Conditions
  - Modules
  - Handlers
  - Roles
  - Collections
  - Demos

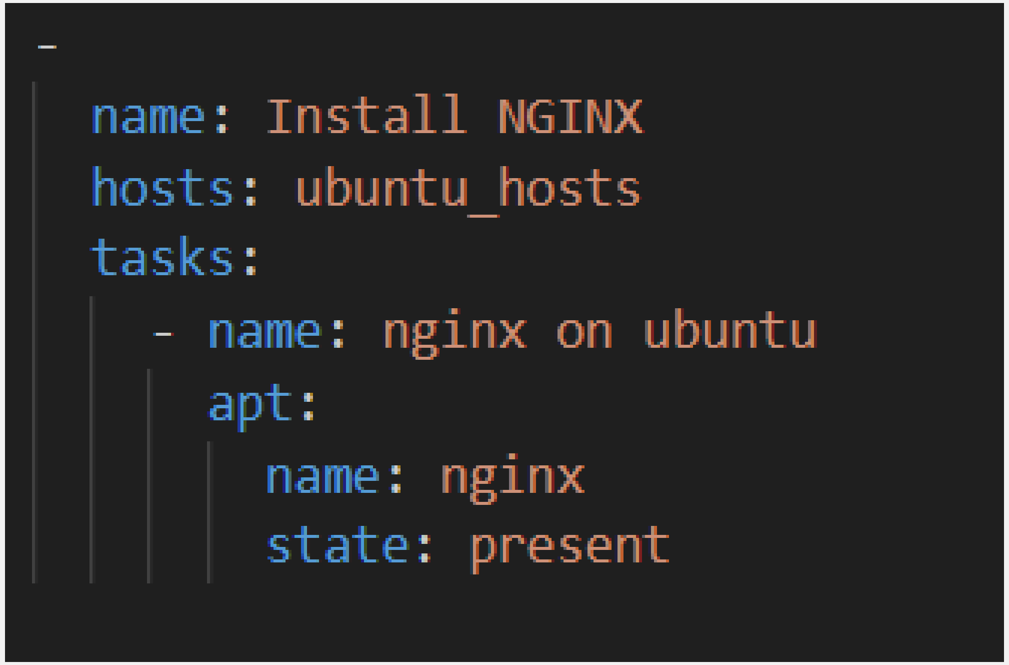
Ansible – Verifying Playbooks

- **check mode:**
  - **Dry run of the playbook where no actual changes made on hosts**
  - **Use** *--check* **option**
  - *ansible-playbook playbook.yml --check*
- **Diff mode:**
  - **Provides a before-and-after comparison of playbook changes**
  - **Use** *--diff* **option**
  - *ansible-playbook playbook.yml --check --diff*

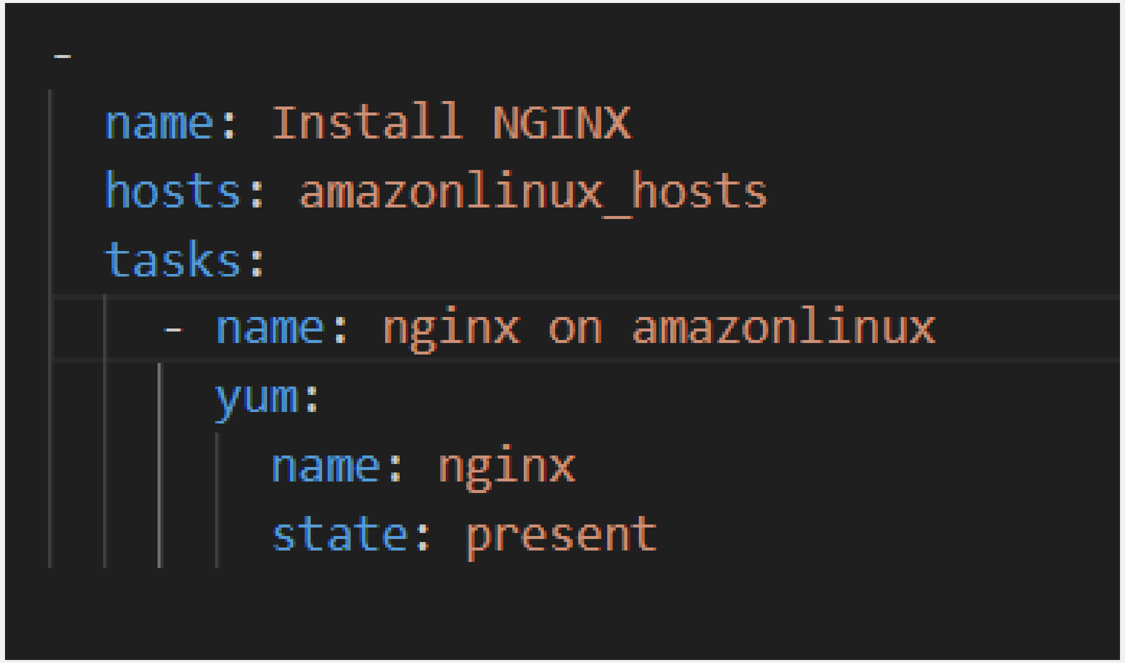
Ansible – Verifying Playbooks Syntax

- **Ensures playbook syntax is error-free**
- **Use the** *--syntax-check* **option**
- *ansible-playbook playbook.yml --syntax-check*

Ansible – Playbook Conditionals



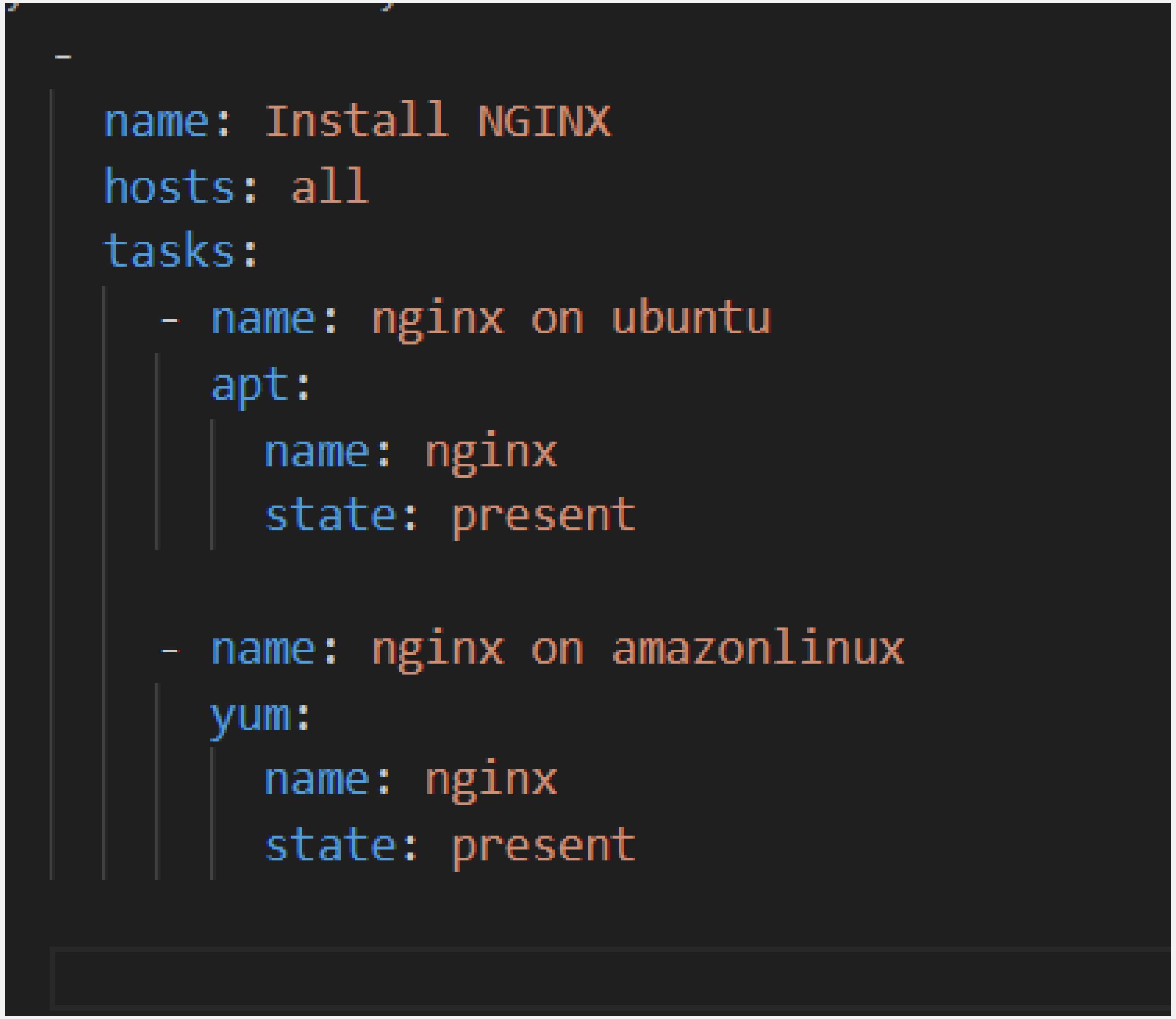
```yaml
-
  name: Install NGINX
  hosts: ubuntu_hosts
  tasks:
    - name: nginx on ubuntu
      apt:
        name: nginx
        state: present
```



```yaml
-
  name: Install NGINX
  hosts: amazonlinux_hosts
  tasks:
    - name: nginx on amazonlinux
      yum:
        name: nginx
        state: present
```

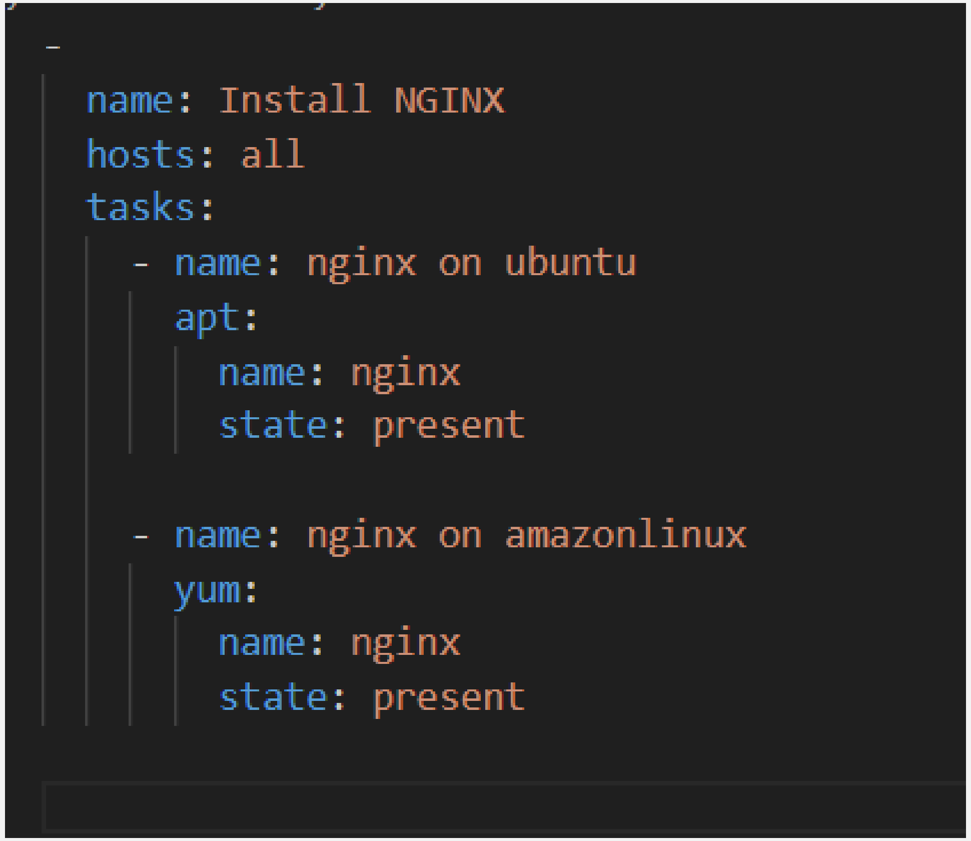
- **How to merge above two files to a single file ?**

Ansible – Playbook Conditionals

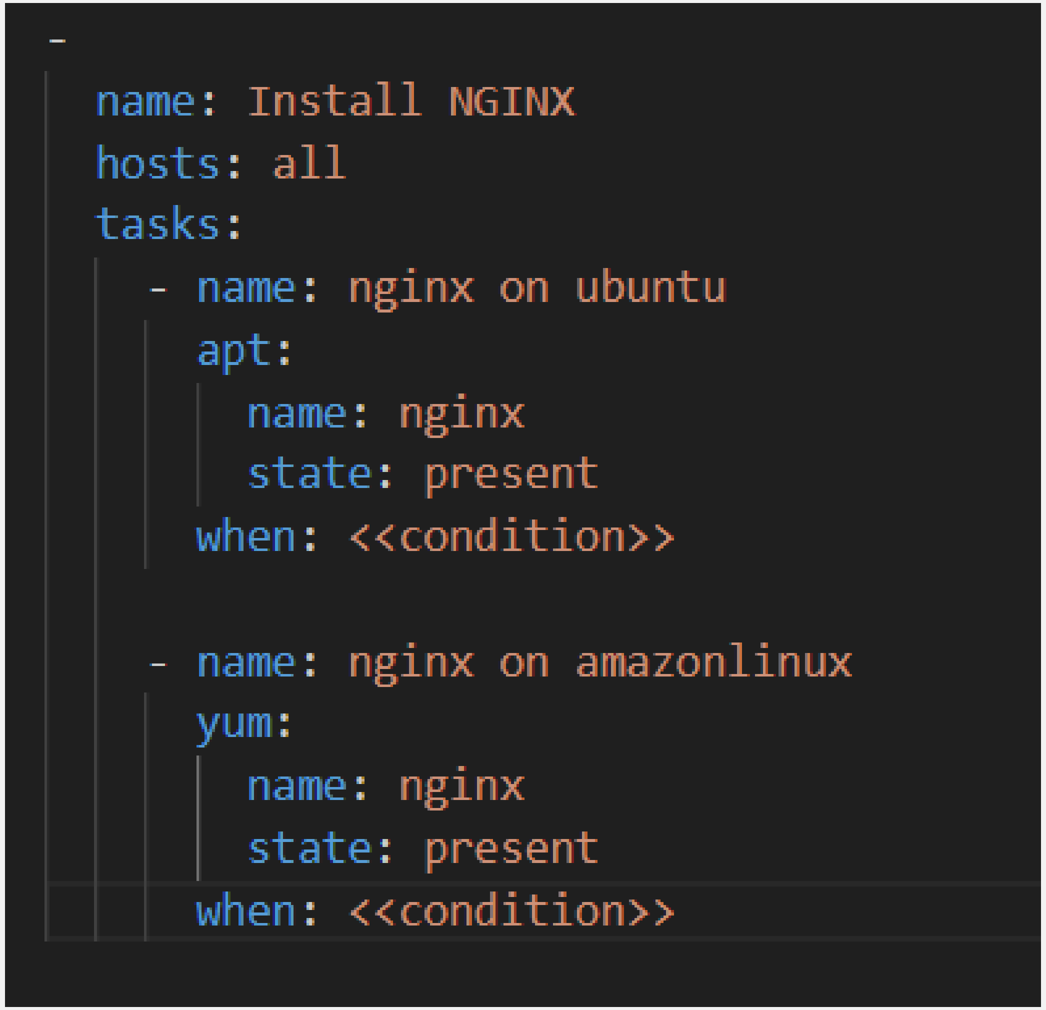


```yaml
-
  name: Install NGINX
  hosts: all
  tasks:
    - name: nginx on ubuntu
      apt:
        name: nginx
        state: present

    - name: nginx on amazonlinux
      yum:
        name: nginx
        state: present
```

Ansible – Playbook Conditionals

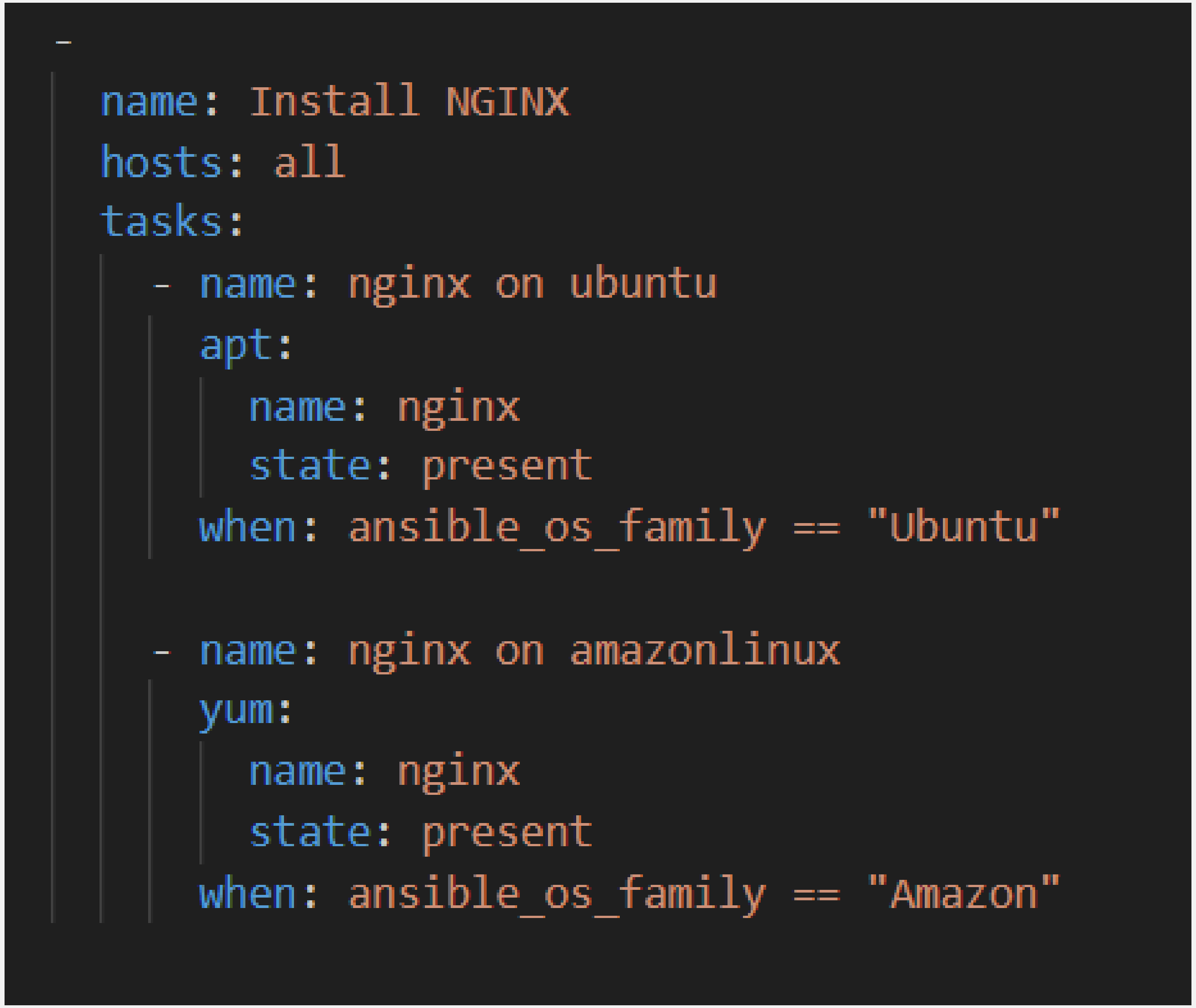


```
-
  name: Install NGINX
  hosts: all
  tasks:
    - name: nginx on ubuntu
      apt:
        name: nginx
        state: present

    - name: nginx on amazonlinux
      yum:
        name: nginx
        state: present
```



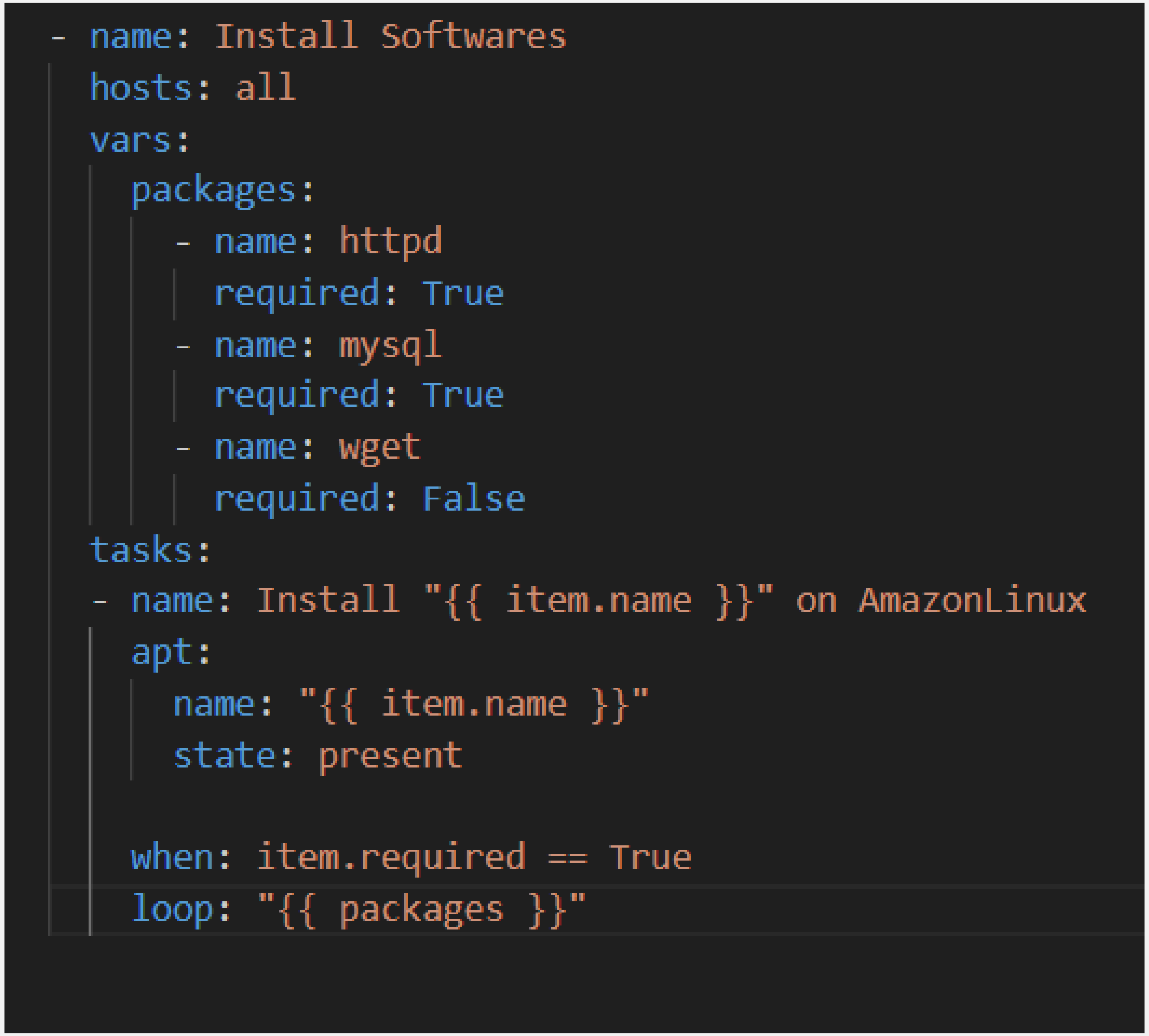
```
-
  name: Install NGINX
  hosts: all
  tasks:
    - name: nginx on ubuntu
      apt:
        name: nginx
        state: present
      when: <<condition>>

    - name: nginx on amazonlinux
      yum:
        name: nginx
        state: present
      when: <<condition>>
```

Ansible – Playbook Conditionals



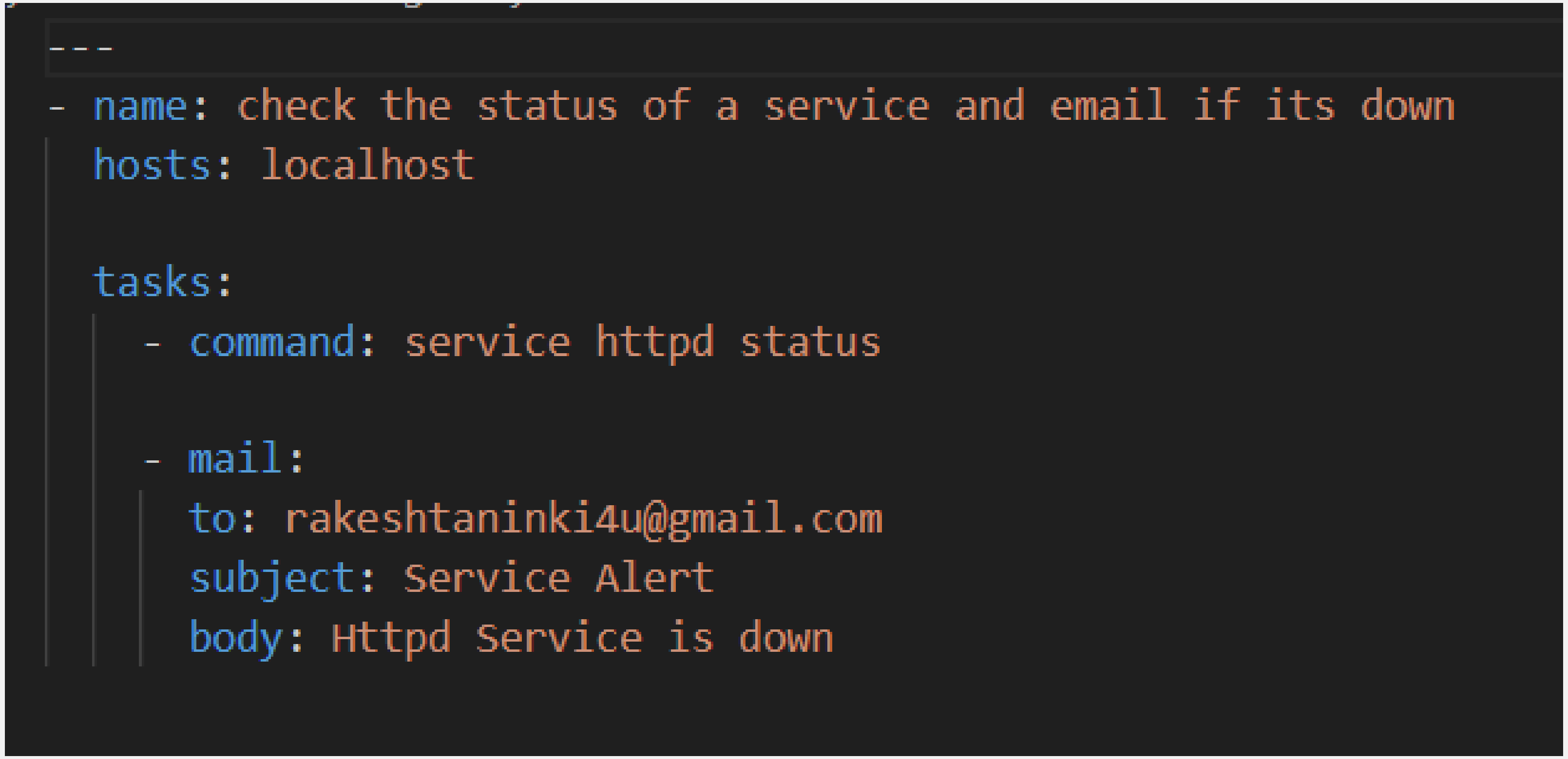
```
-
  name: Install NGINX
  hosts: all
  tasks:
    - name: nginx on ubuntu
      apt:
        name: nginx
        state: present
      when: ansible_os_family == "Ubuntu"

    - name: nginx on amazonlinux
      yum:
        name: nginx
        state: present
      when: ansible_os_family == "Amazon"
```

Ansible – Playbook Loops



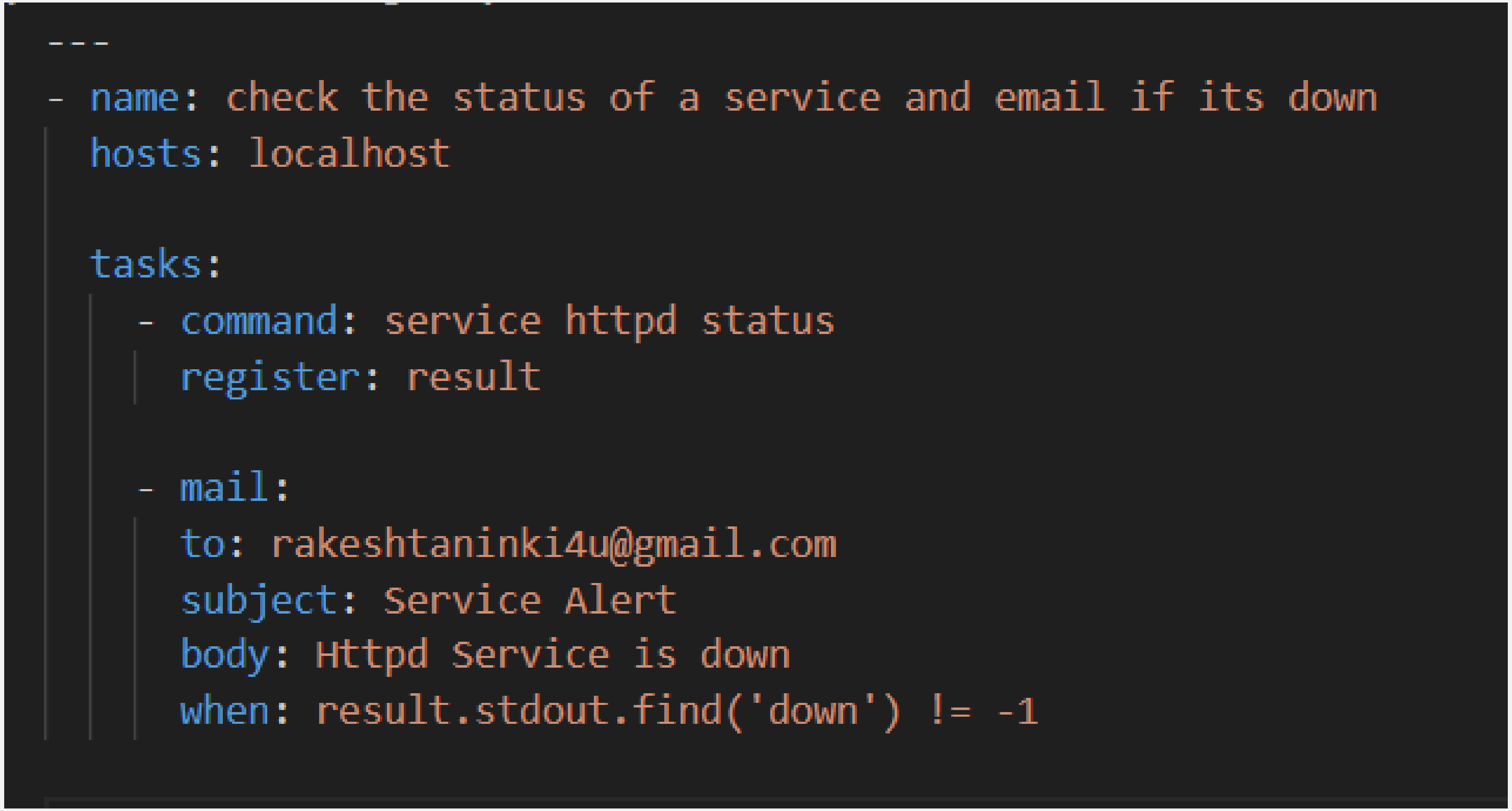
```yaml
- name: Install Softwares
  hosts: all
  vars:
    packages:
      - name: httpd
        required: True
      - name: mysql
        required: True
      - name: wget
        required: False
  tasks:
  - name: Install "{{ item.name }}" on AmazonLinux
    apt:
      name: "{{ item.name }}"
      state: present

    when: item.required == True
    loop: "{{ packages }}"
```

Ansible – Playbook Conditionals & Register



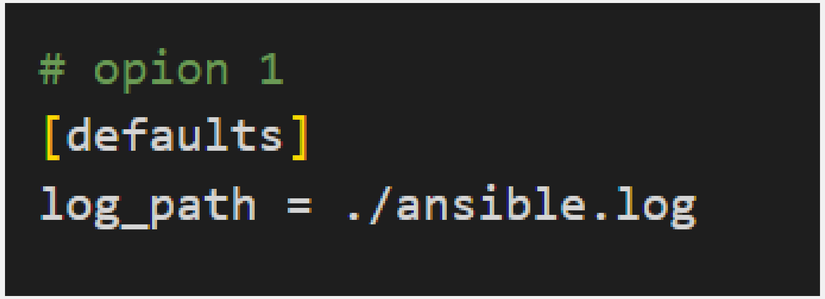
```
---
- name: check the status of a service and email if its down
  hosts: localhost

  tasks:
    - command: service httpd status

    - mail:
      to: rakeshtaninki4u@gmail.com
      subject: Service Alert
      body: Httpd Service is down
```

Ansible – Playbook Conditionals & Register

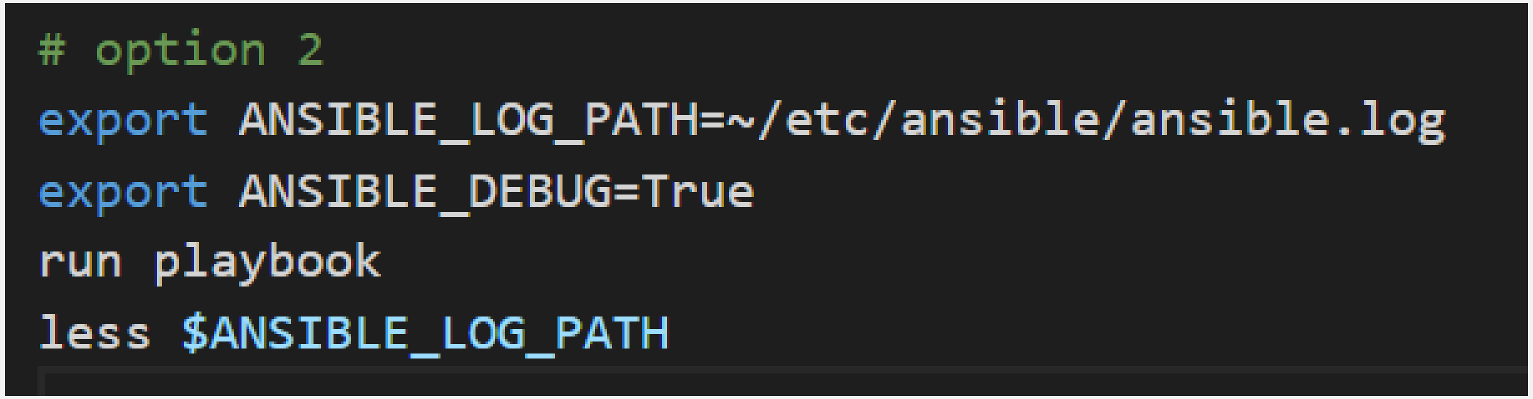


```yaml
---
- name: check the status of a service and email if its down
  hosts: localhost

  tasks:
    - command: service httpd status
      register: result

    - mail:
      to: rakeshtaninki4u@gmail.com
      subject: Service Alert
      body: Httpd Service is down
      when: result.stdout.find('down') != -1
```

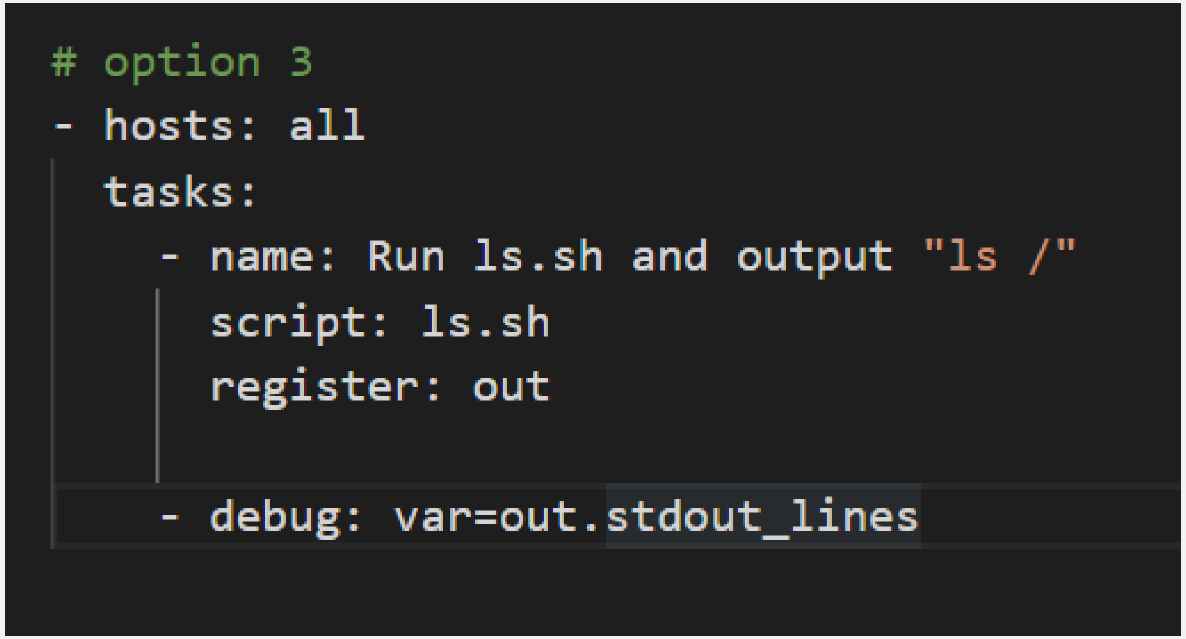
Ansible – Playbook Logging



```
# opion 1
[defaults]
log_path = ./ansible.log
```

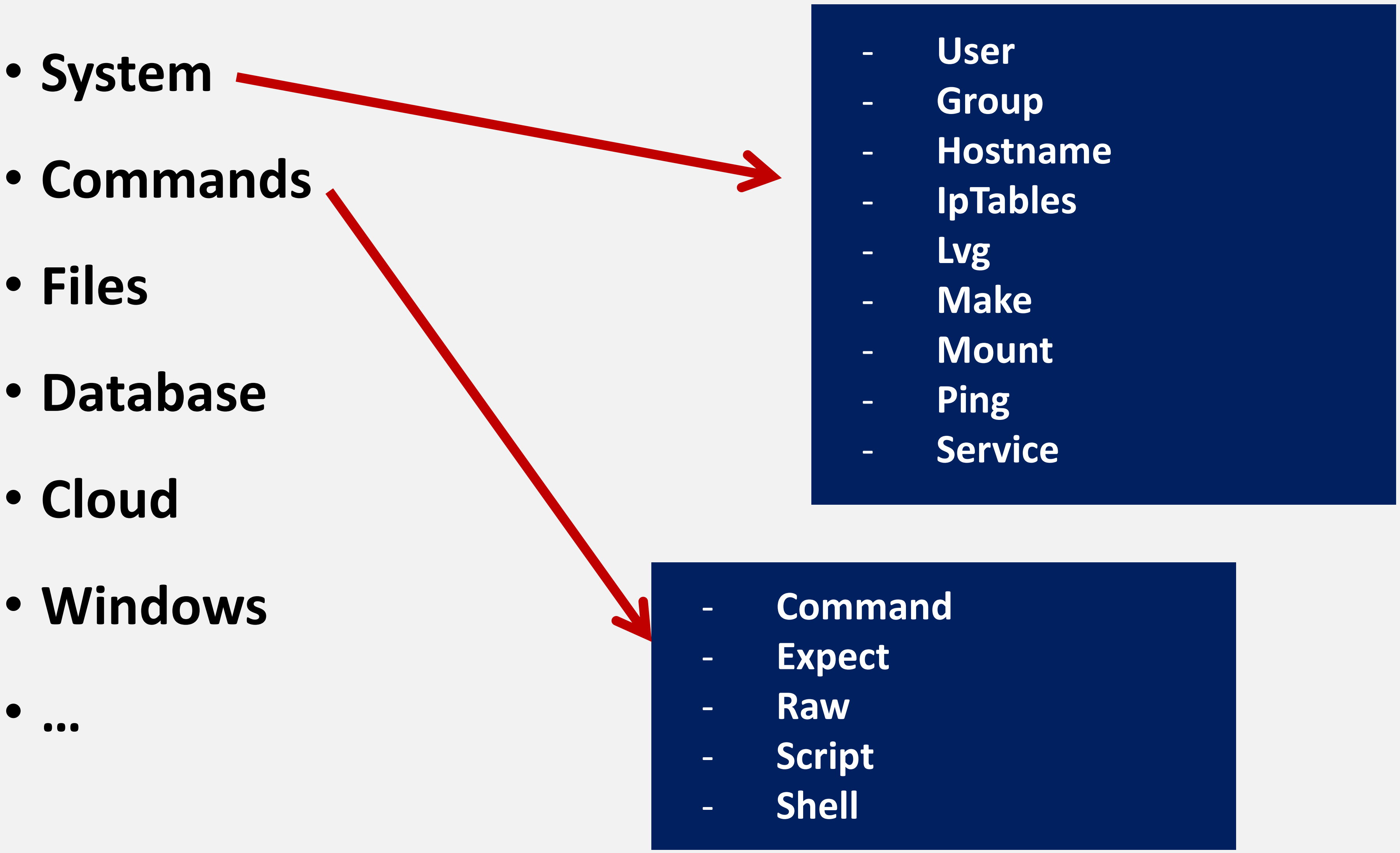


```
# option 2
export ANSIBLE_LOG_PATH=~/etc/ansible/ansible.log
export ANSIBLE_DEBUG=True
run playbook
less $ANSIBLE_LOG_PATH
```



```
# option 3
- hosts: all
  tasks:
    - name: Run ls.sh and output "ls /"
      script: ls.sh
      register: out

    - debug: var=out.stdout_lines
```

Ansible – Modules

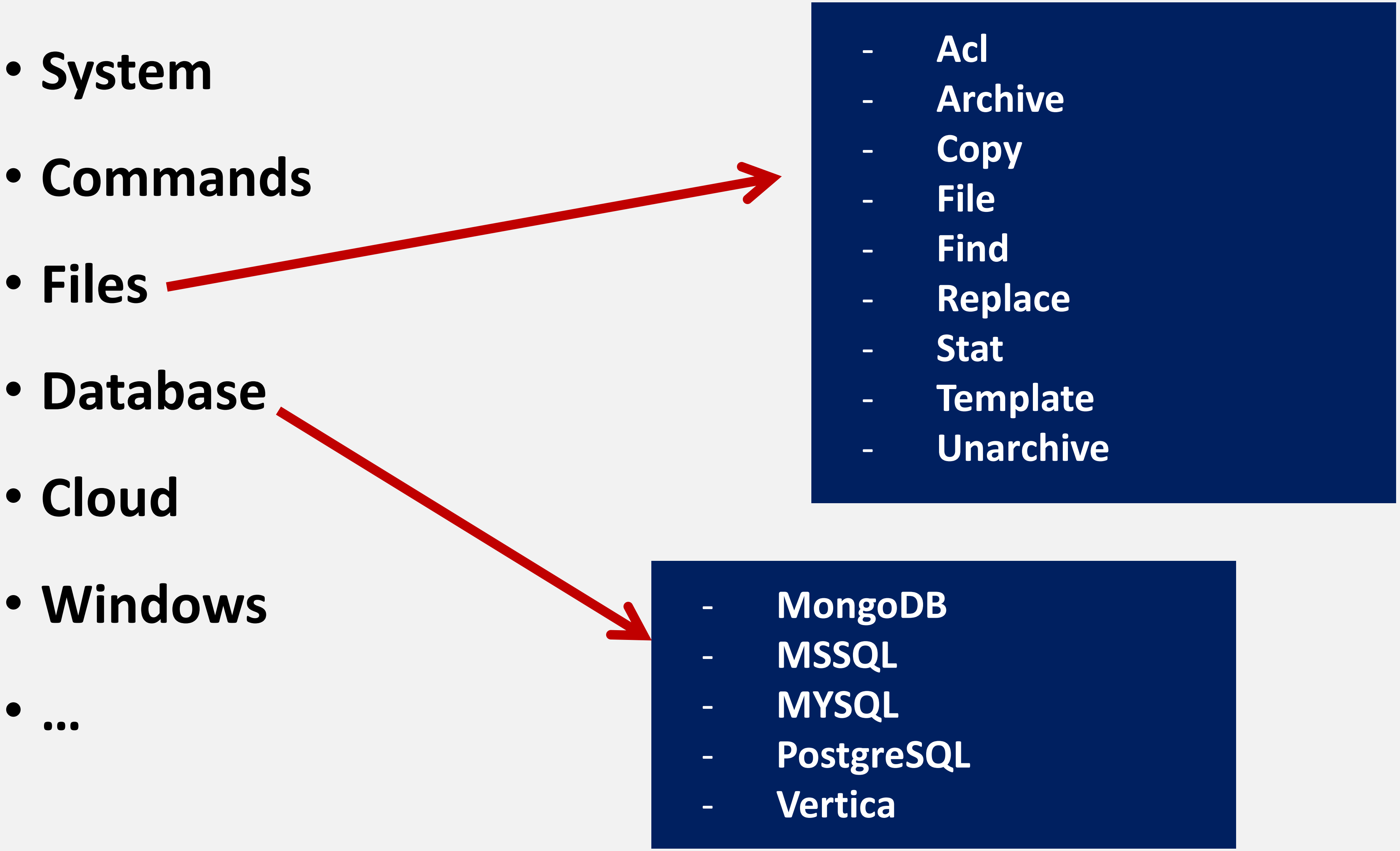


- System
- Commands
- Files
- Database
- Cloud
- Windows
- …

- User
- Group
- Hostname
- IpTables
- Lvg
- Make
- Mount
- Ping
- Service

- Command
- Expect
- Raw
- Script
- Shell

Ansible – Modules



- System
- Commands
- Files
- Database
- Cloud
- Windows
- …

- Acl
- Archive
- Copy
- File
- Find
- Replace
- Stat
- Template
- Unarchive

- MongoDB
- MSSQL
- MYSQL
- PostgreSQL
- Vertica

Ansible – Modules

- System
- Commands
- Files
- Database
- Cloud
- Windows
- …

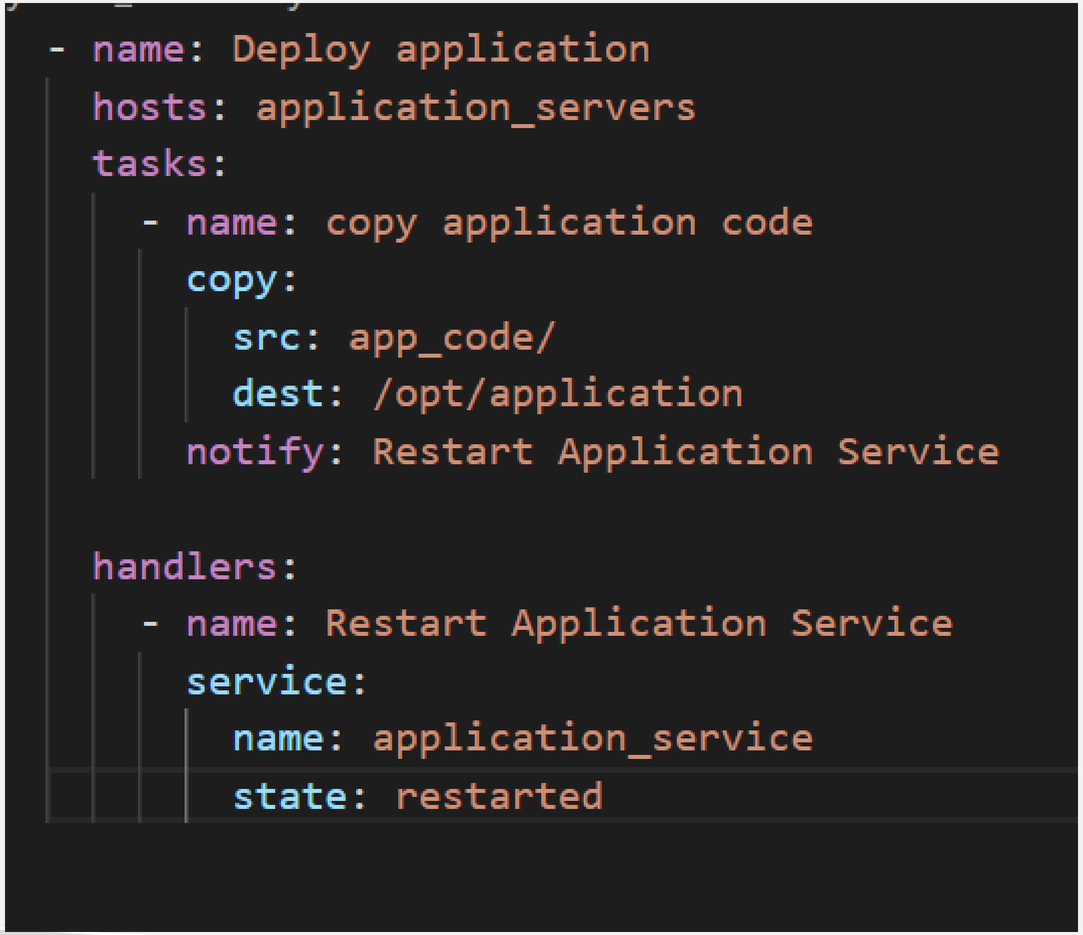
- AWS
- Azure
- GCP
- Digital Ocean
- Docker
- VMWare
- RackSpace

Ansible – Plugins

- AWS EC2 Autoscaling or Load balancing problem
- A piece of code that modifies the functionality of Ansible
- Enhance various aspects of Ansible like Inventory, modules, Callbacks
- Flexible and powerful way to customize
- Dynamic Inventory Plugin ➔ maintains the latest infra information
- Module Plugin ➔ AMI, Instance Types, Security Groups etc
- Action Plugin ➔ Load Balancer, Auto Scaling groups etc

Ansible – Handlers

- Tasks triggered by events / notifications
- Defined in Playbook, executed when notified by a task



```yaml
- name: Deploy application
  hosts: application_servers
  tasks:
    - name: copy application code
      copy:
        src: app_code/
        dest: /opt/application
      notify: Restart Application Service

  handlers:
    - name: Restart Application Service
      service:
        name: application_service
        state: restarted
```

Ansible – Roles

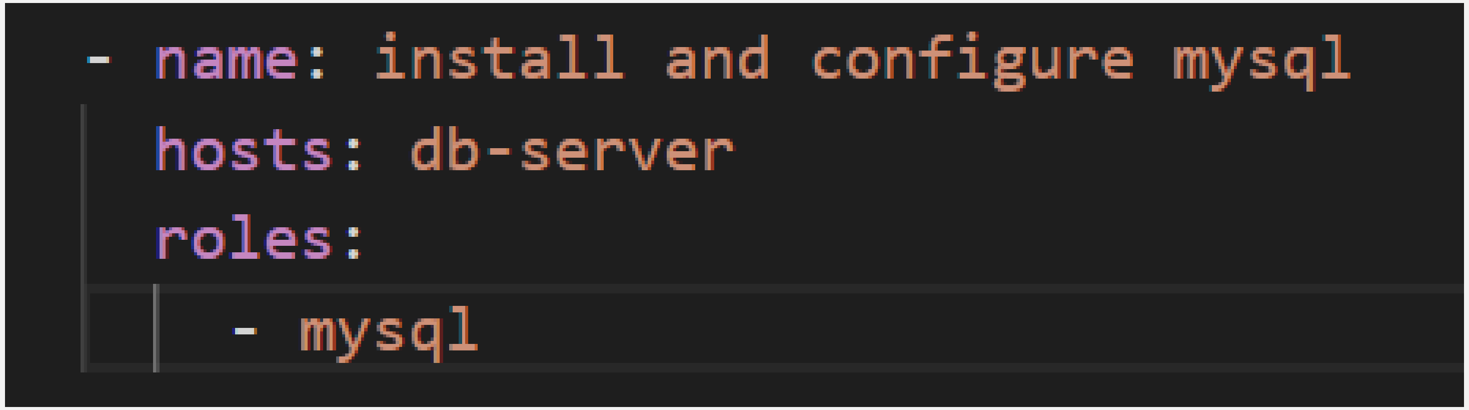
- **Roles are used to reuse the task definitions in other playbooks**

- **Ansible Galaxy – maintains the public roles / roles related functionality**

- **To create role directory structure –** *ansible-galaxy init mysql*

- *mysql*
  - *readme.md*
  - *templates/*
  - *tasks/*
  - *handlers/*
  - *vars/*
  - *defaults/*
  - *meta/*

Folder Structure



```
- name: install and configure mysql
  hosts: db-server
  roles:
    - mysql
```

/etc/ansible/roles

Ansible – Roles

- ***/etc/ansible/roles* – default path to maintain role folders**
- **Or**
- **Update the /etc/ansible/ansible.cfg**
- **roles_path = /etc/ansible/roles**
- **You can also upload your role to ansible galaxy through a github repo**
- **To search in galaxy,** *ansible-galaxy search mysql*
- **To install,** *ansible-galaxy install geerlingguy.mysql*

Ansible – Collections

- **Collections are also similar to Roles**
- **Collections can have plugins, roles, modules as extra compared to Roles**
- **These are also part of ansible galaxy only**
- **To install**
  - *ansible-galaxy collection install community.mysql*

Ansible – Vaults

- **Ansible Vault is a feature that allows you to keep all your secrets safe.**
- **It can encrypt entire files, entire YAML playbooks or even a few variables.**
- *ansible-vault encrypt_string --ask-vault-pass ‘password' --name 'ssh_password’*

Thank you, will meet in tomorrow’s session

ANY
QUESTIONS?