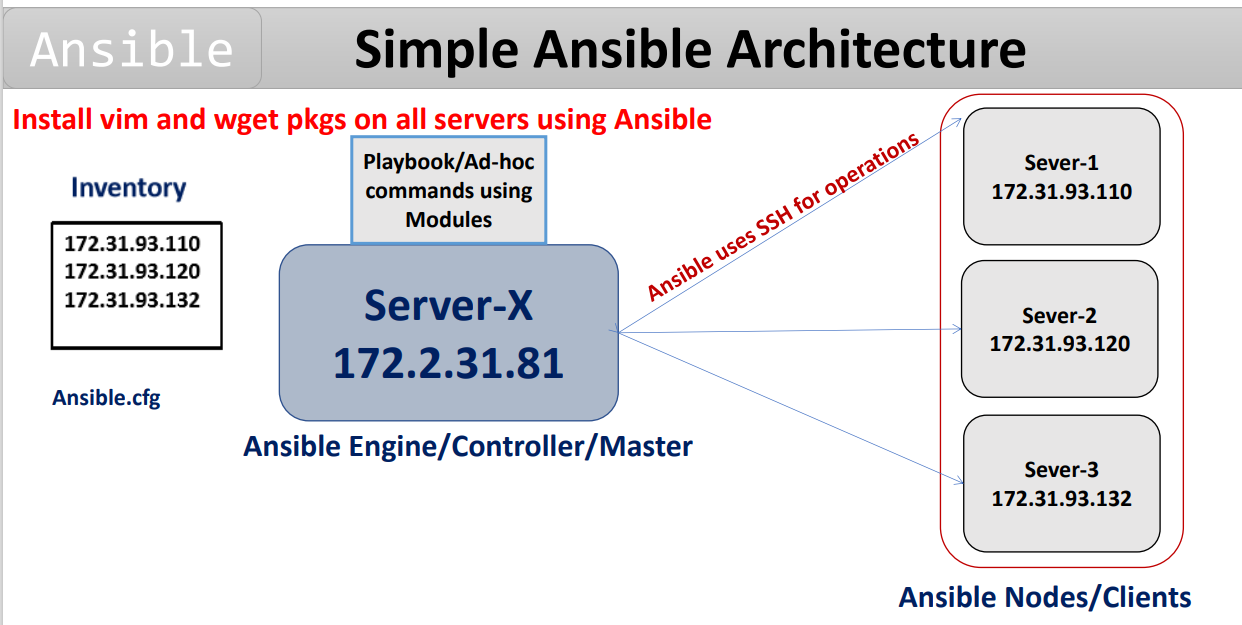
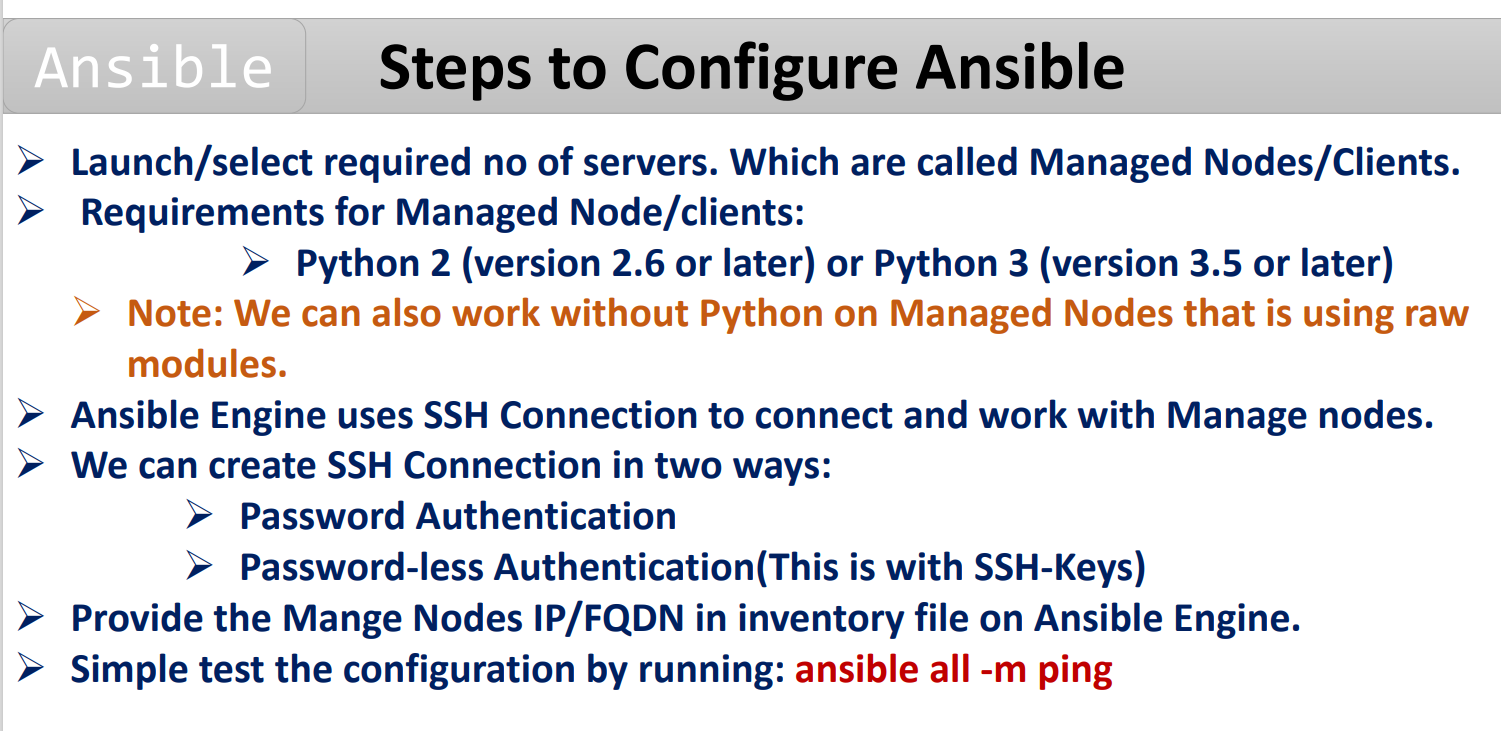
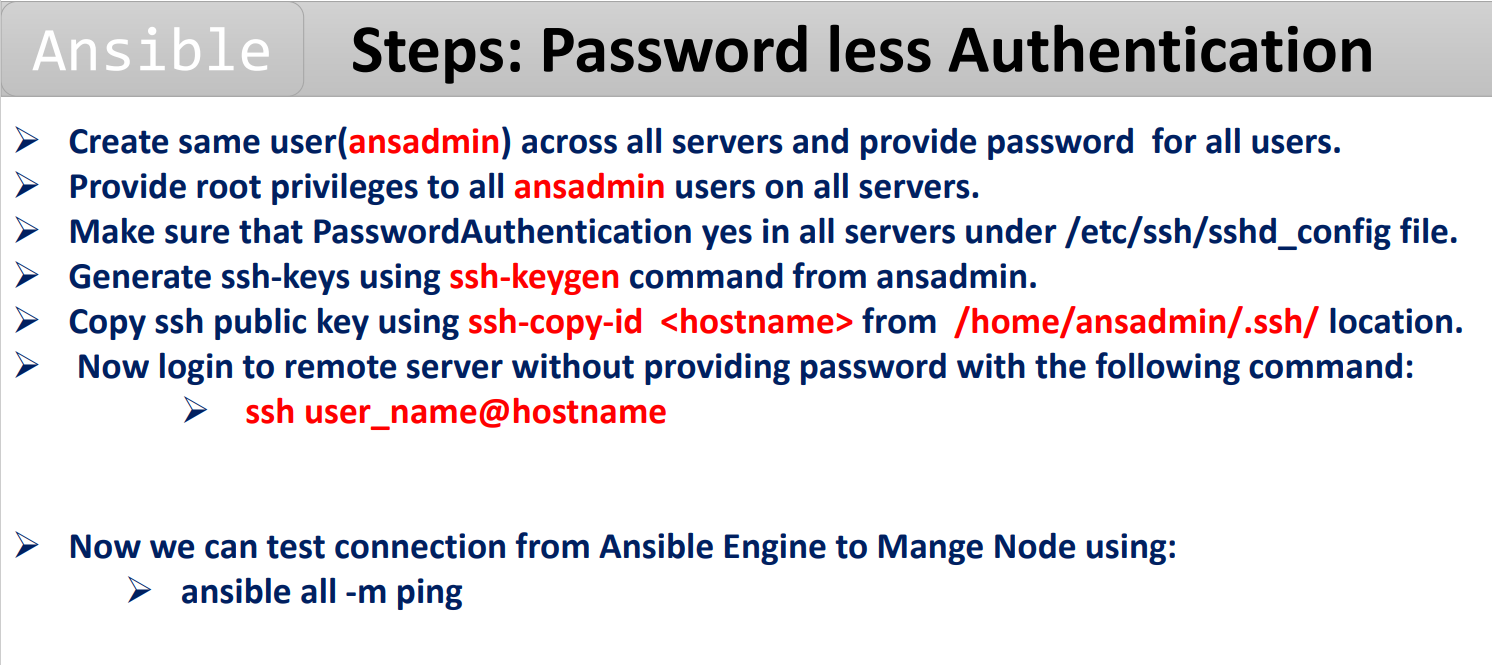
**ANSIBLE**



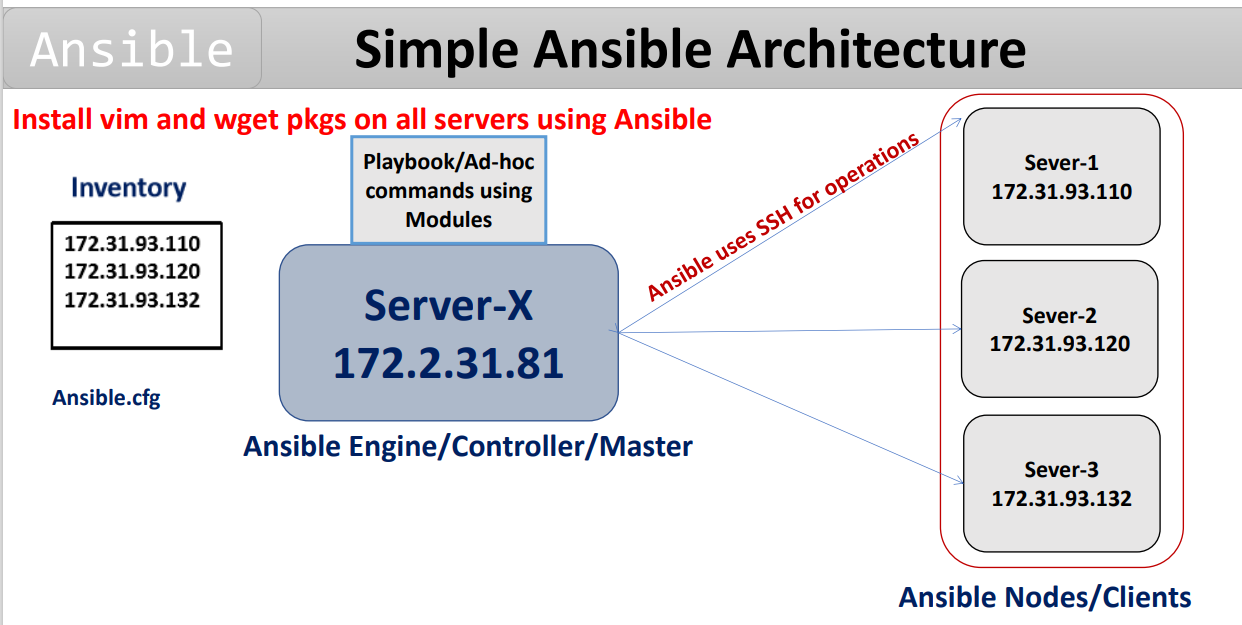


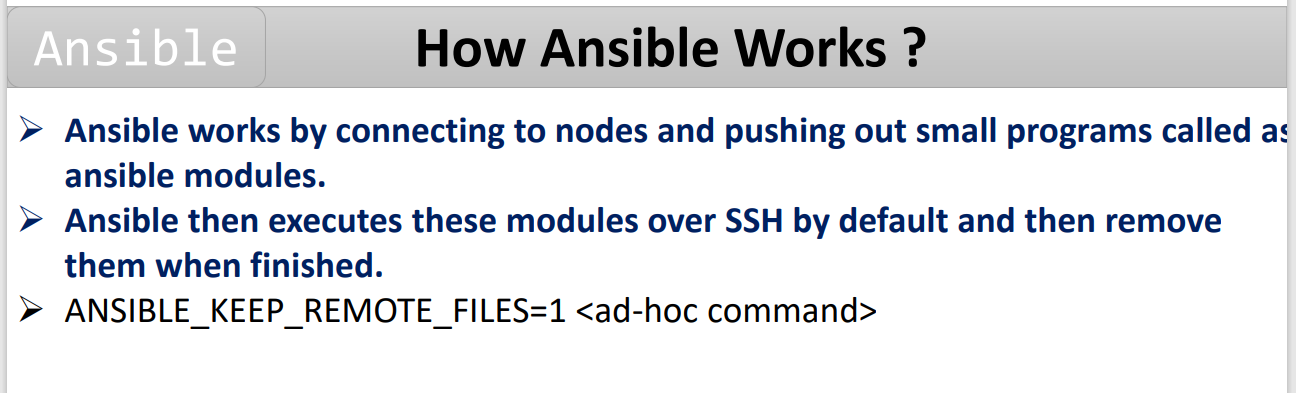


Password authentication:

Ansible IP -m ping -k

How Ansible Works ? (Executing Ad-hoc commands or Playbooks )





Ex: ANSIBLE\_KEEP\_REMOTE\_FILES=1 ansible all -m shell -a "uptime"

Ad-hoc Syntax:

Ansible group1:group2 -m module [-a argument value]

Ex: Ansible all -m shell -a uptime

Ex: Ansible all -m ping

Ansible-doc -l

Module: pre-defined program, this program can execute our tasks on managed nodes.

Copy:

Transfer a file from Ansible Engine to Nodes using copy module

* Syntax: ansible all -m copy -a “src=/source/file/path dest=/dest/location”
* ansible all -m copy -a "src=./demo.txt dest=/tmp/"
* ansible all -m copy -a "src=/home/ansadmin/kishor.txt dest=/tmp"

Fetch:

 Download a file from Ansible Managed Nodes to Ansible Engine

* syntax: ansible db\_servers -m fetch -a “src=/source/file/path dest=/dest/location”
* ansible all -m fetch -a "src=/tmp/kishor.txt dest=/home/ansadmin/"
* ansible group1 -m fetch -a "src=/tmp/demo.txt dest=/home/ansadmin/{{inventory\_hostname}}\_demo.txt flat=yes"

file creation:

Create or Delete a file or dierctory on Managed Nodes

* ansible all -m file -a “path=/tmp/hello.txt state=touch”
* ansible all -m file -a “path=/tmp/hello.txt state=touch mode=777”
* ansible all -m file -a “path=/tmp/hello.txt state=absent”

directory;

* ansible all -m file -a “path=/tmp/hello state=directory”
* ansible all -m file -a “path=/etc/hello.txt state=touch” -b
* -b or –become for root access

**Installation Module:**

 Install a package like git, httpd, mysql, git on Linux Systems using

Centos, RHEL, Amazon linux: module: yum

Ubuntu/Debian: module: apt

Ex: Ansible all -m yum -a “name=git state=present” -b

Present/Installed: for new installation of package

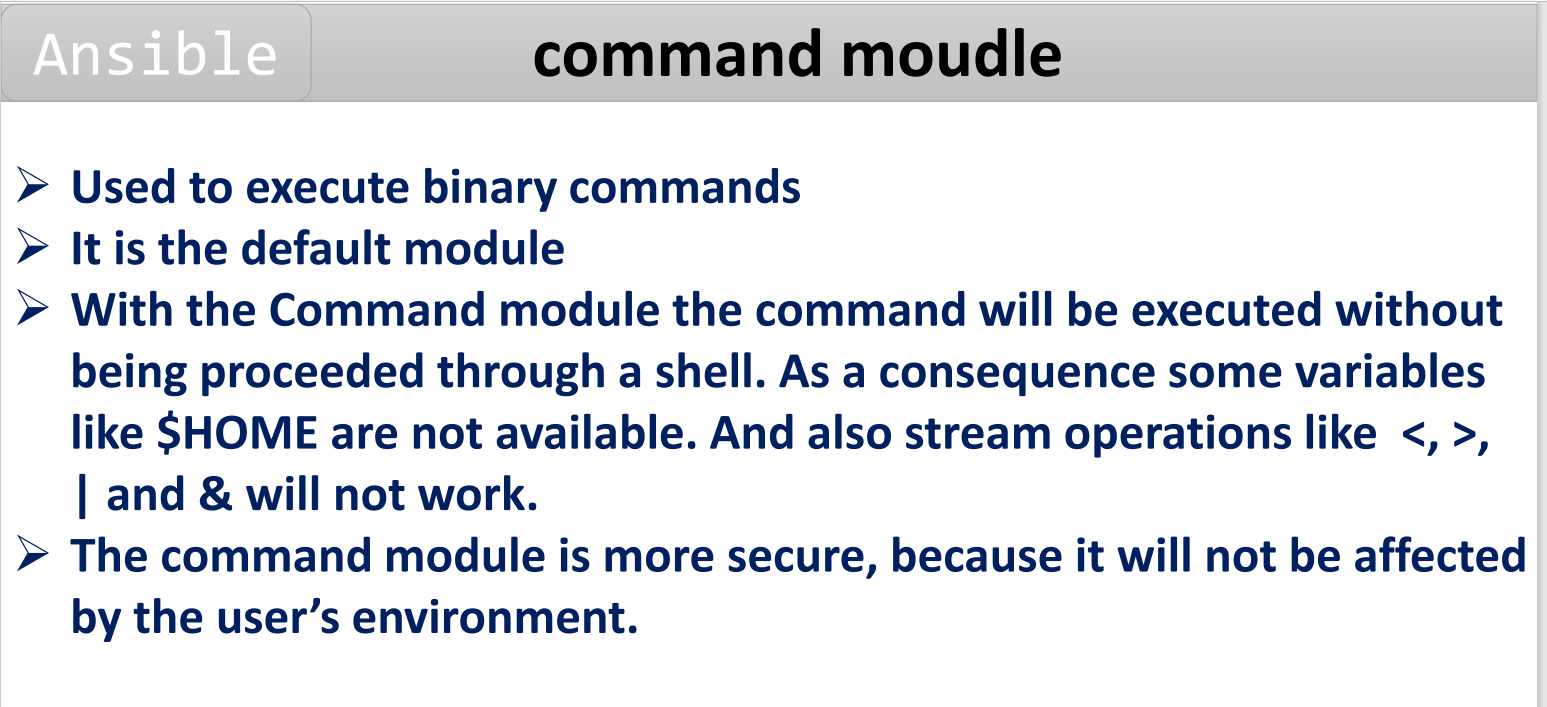
Absent/removed: For uninstalling of package

Latest: from old version package can be update with new version package.

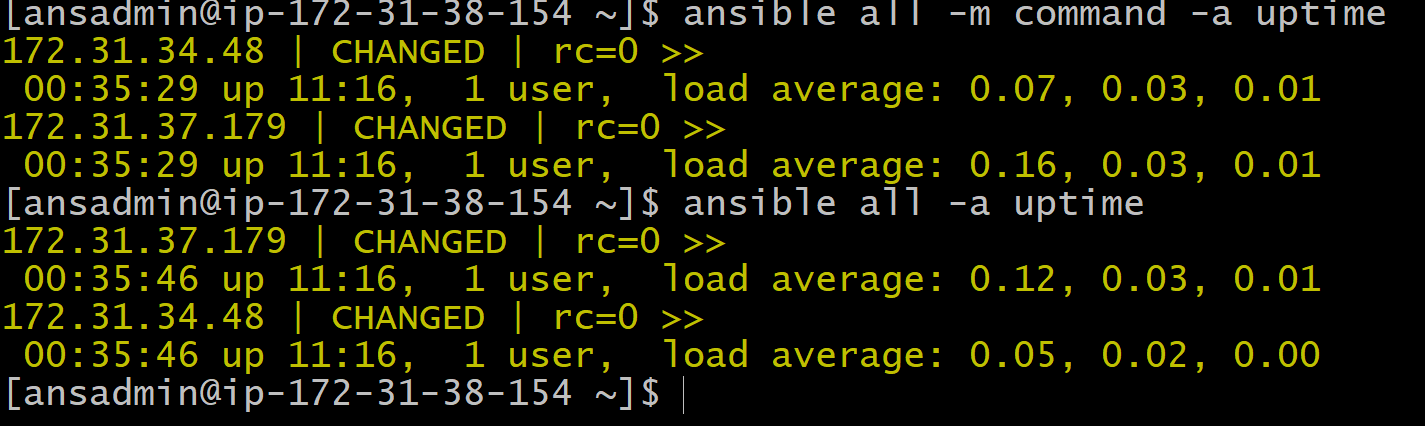
Ex: ansible webgroup -m yum -a “name=nginx state=present” -b

Ex: Git,httpd,mysql,nginx

**Command module:**

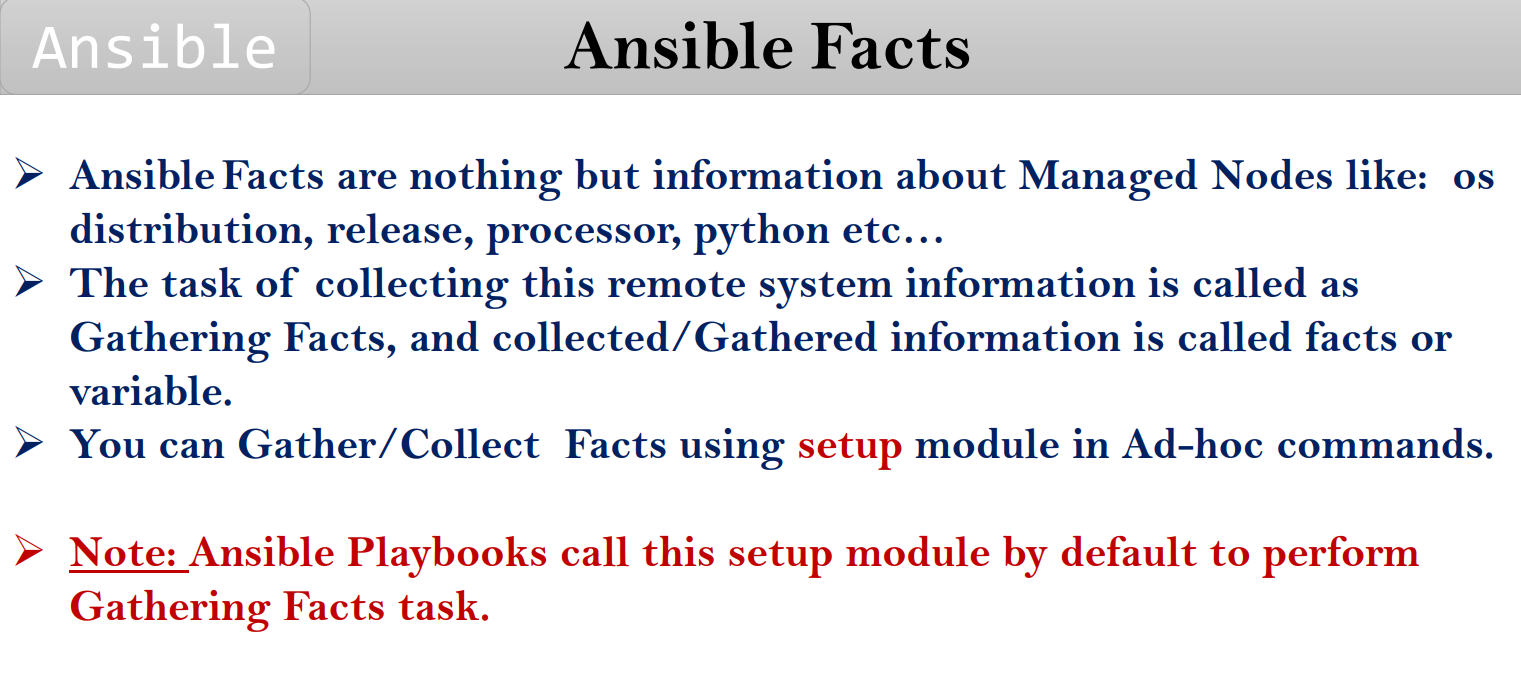


Ex: ansible all -m command -a “uptime”



Ansible all -m command -a “ls > results.txt”

**Ansible Facts:**



**Ex: dd**

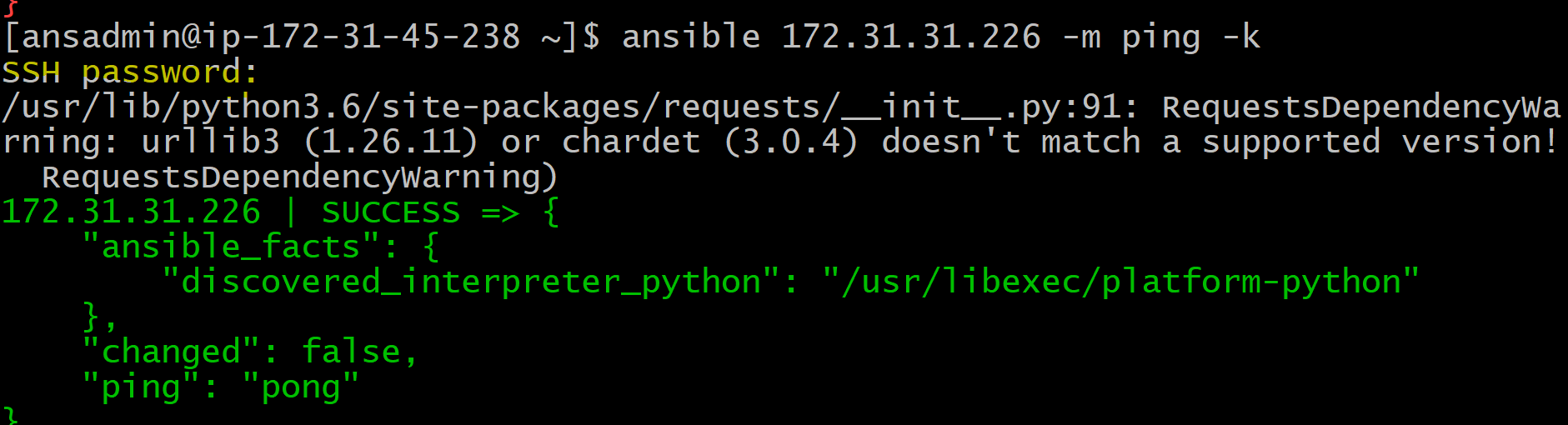
ansible 172.31.34.48 -m setup

**password Authentication:**

working with managed nodes with password:

two ways we can connect.

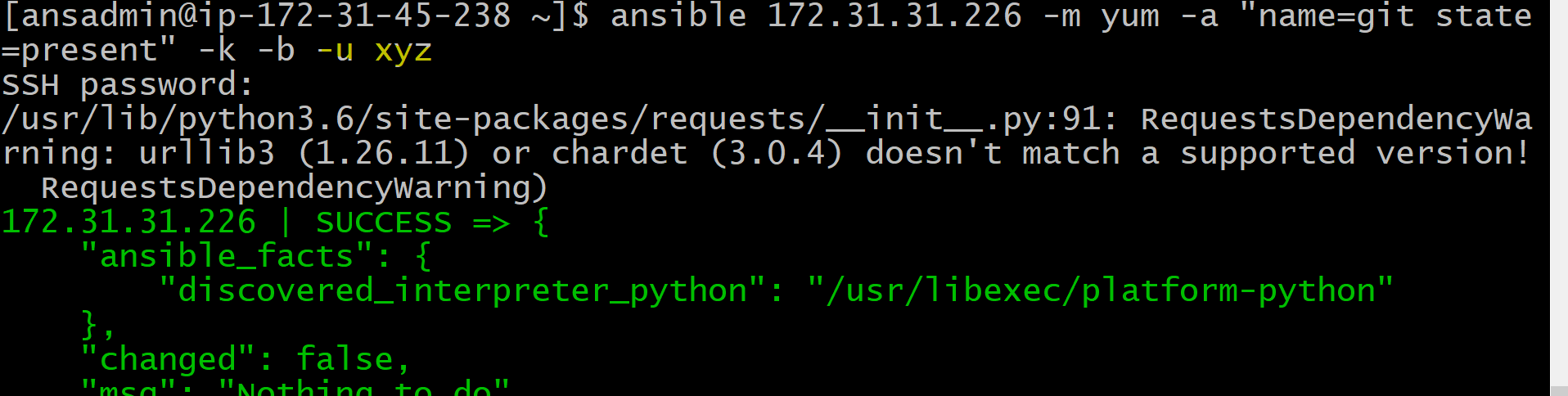
1. Using SSH keys
2. Using SSH password.



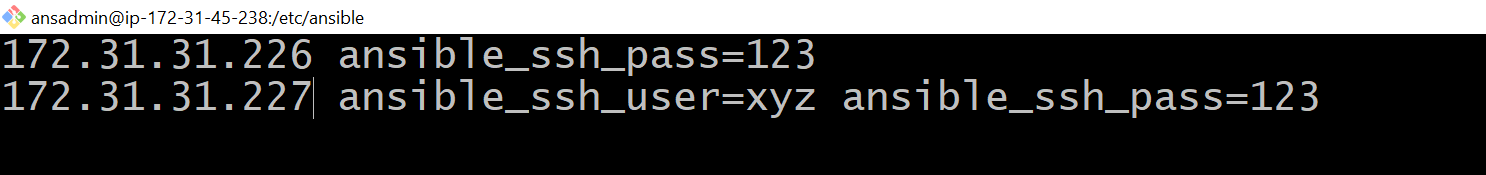
Ex:ansible 172.31.31.226 -m yum -a "name=git state=present" -k -b

Login with different users:

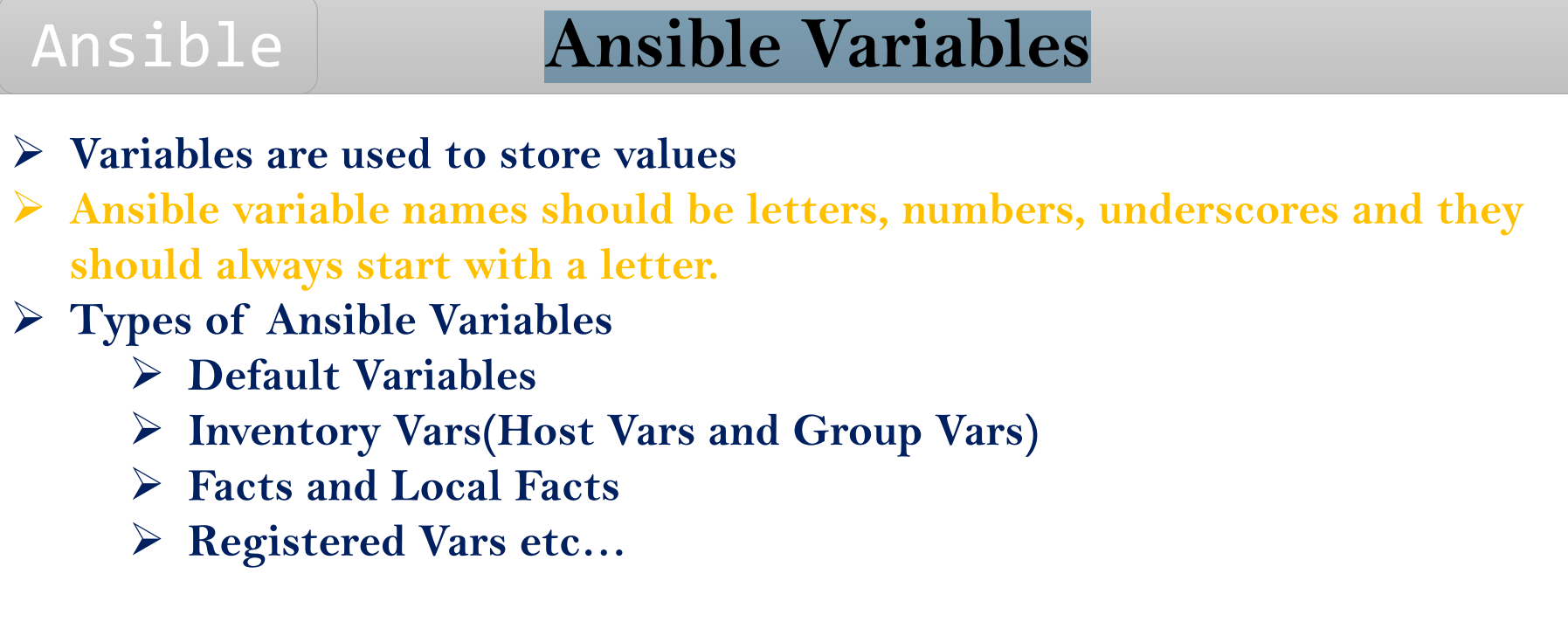
-u xyz



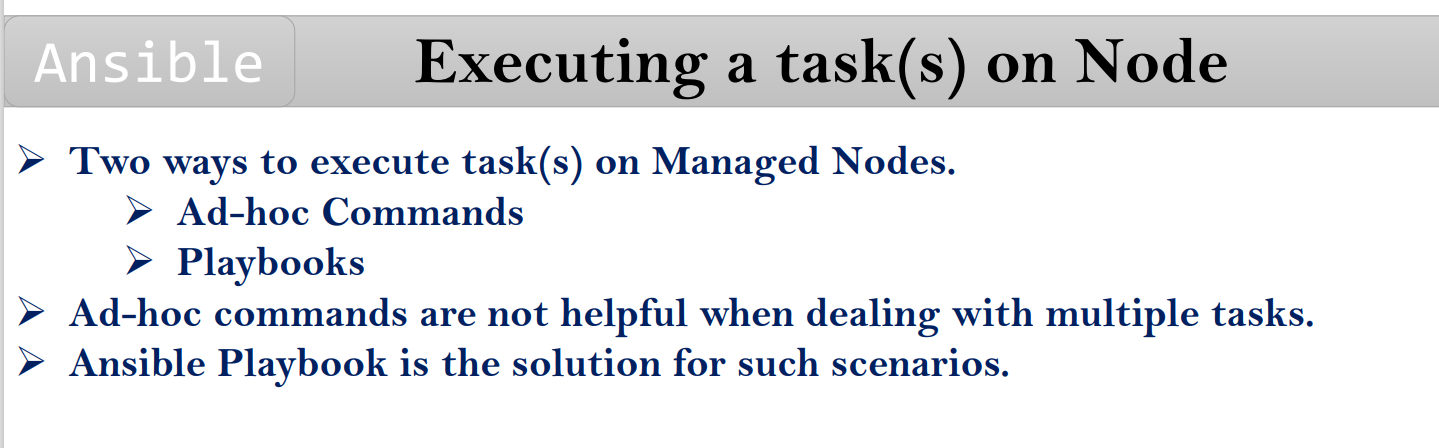
Store username and password:

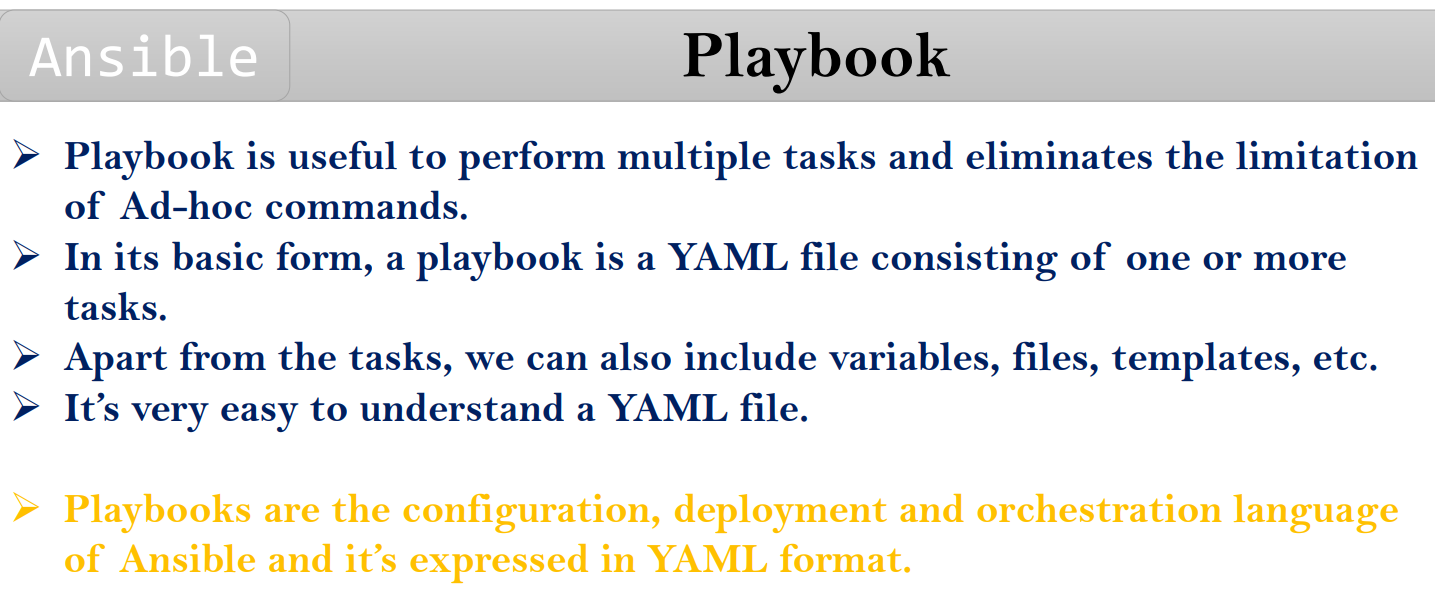


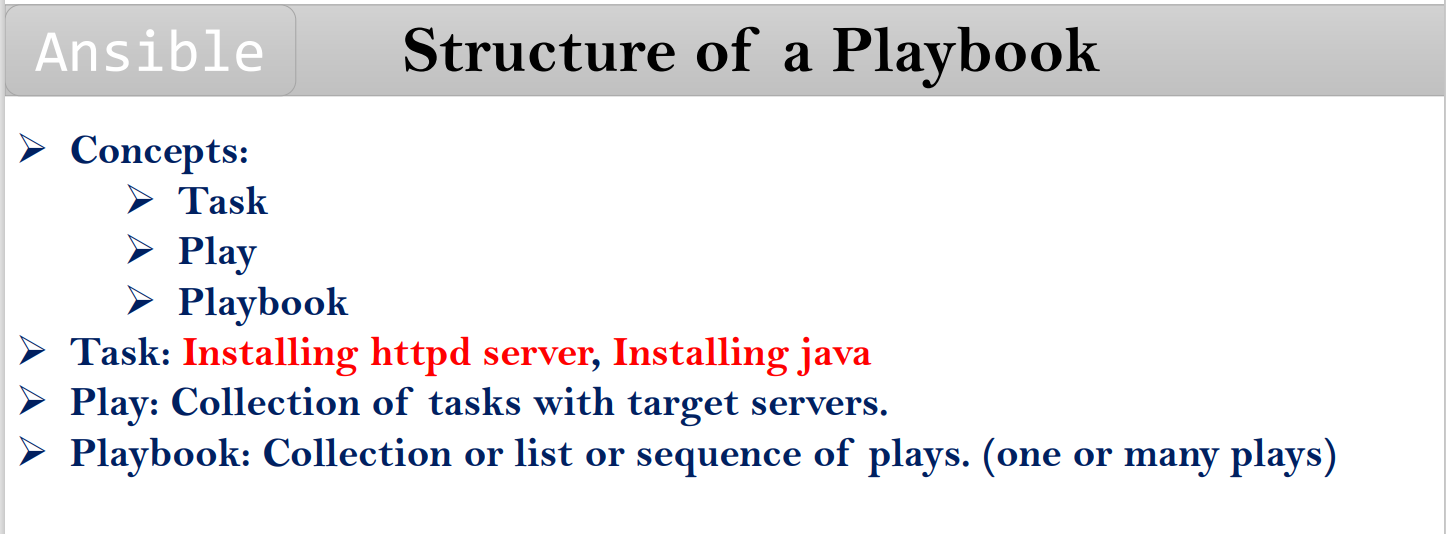
**Ansible Variables**

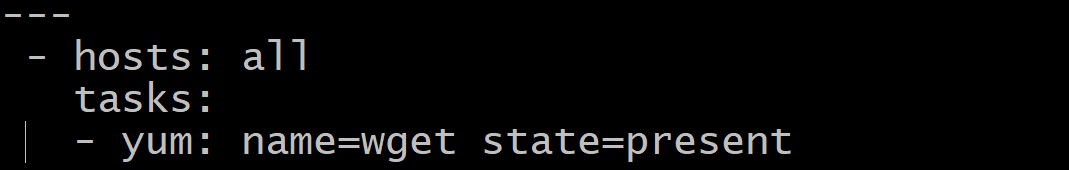


**Introduction to Playbooks:**

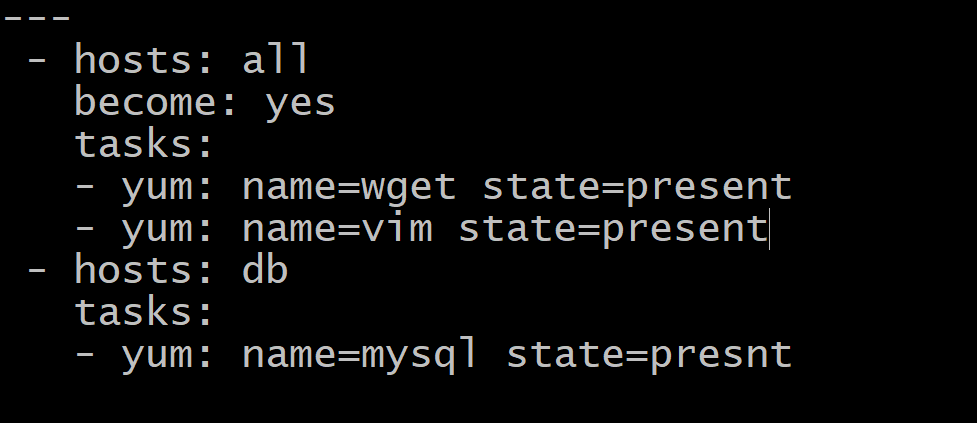




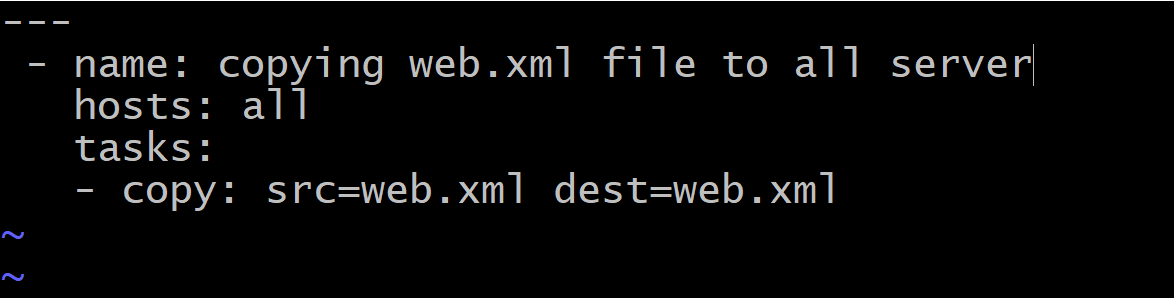


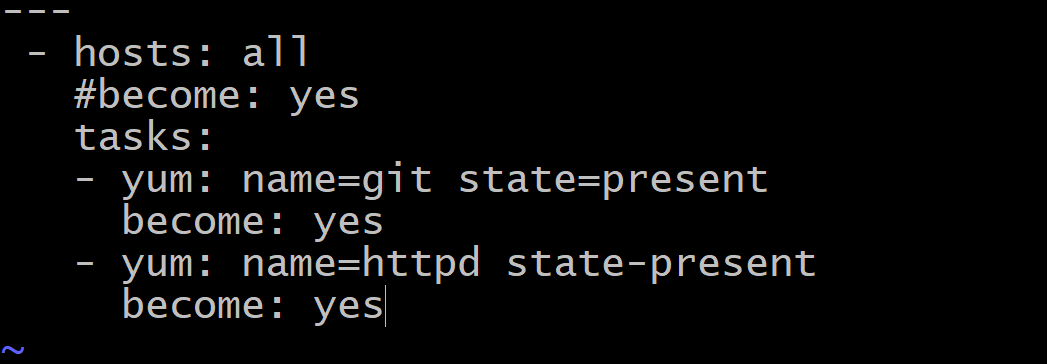
Simple play: 

Multiple play:



copy ex:



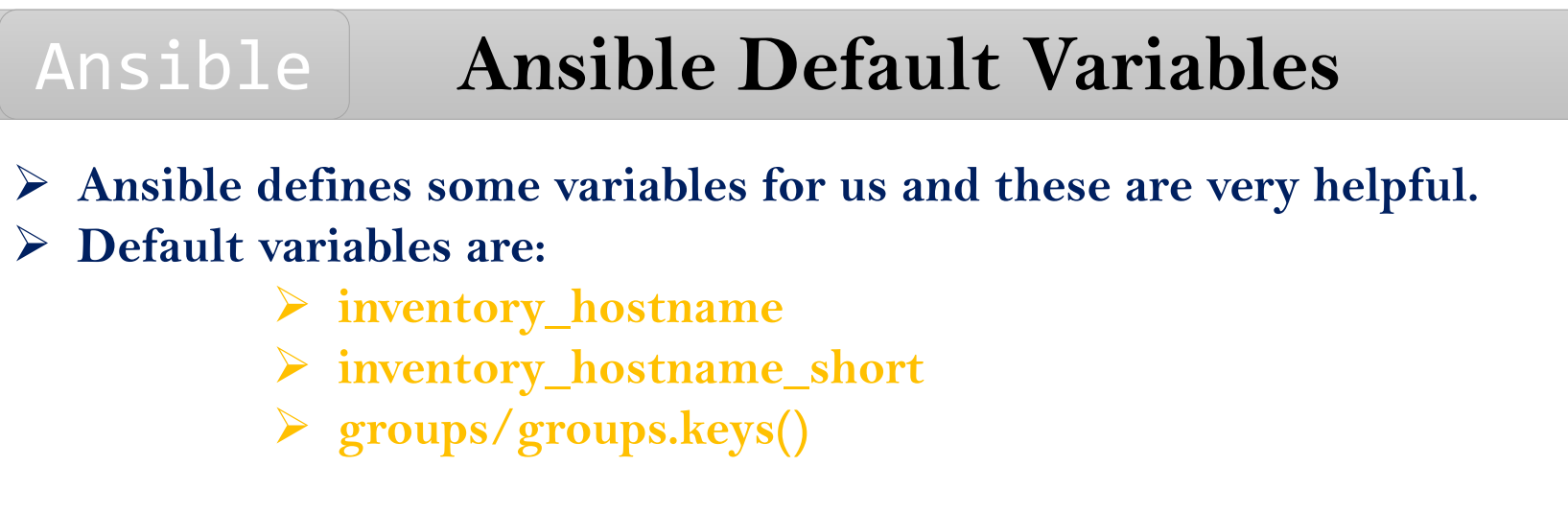


How to verify syntax validation:

ansible-playbook wget\_install.yml --syntax-check

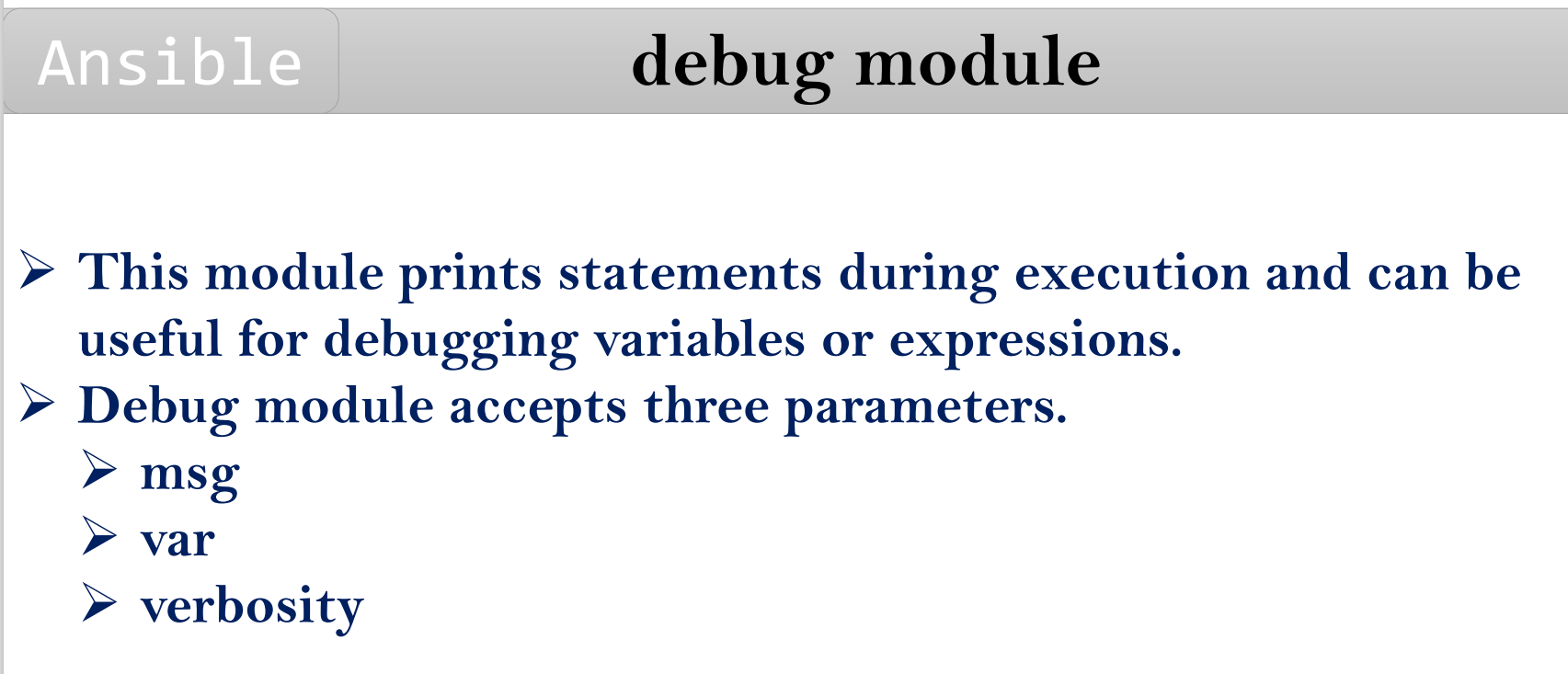
Dry run mode:

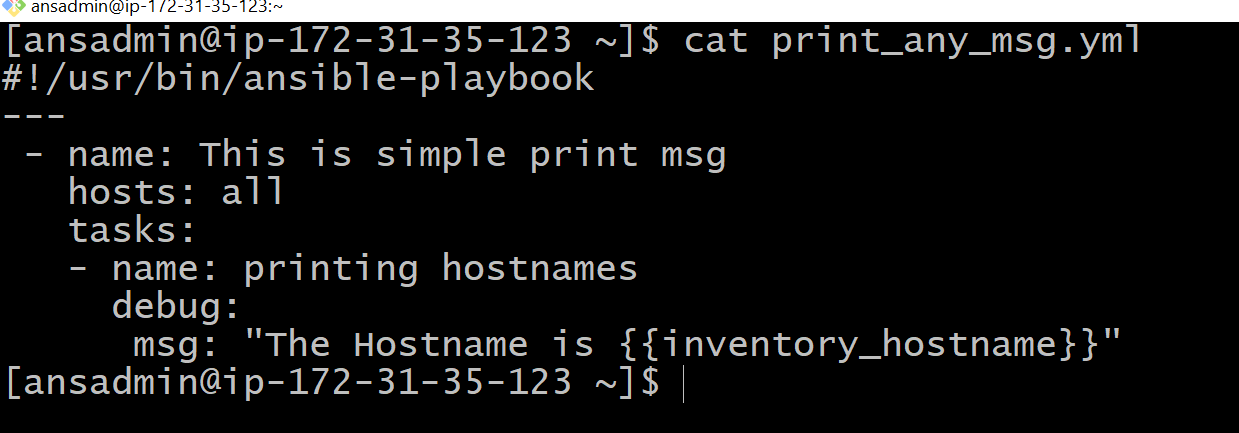
ansible-playbook wget\_install.yml --check



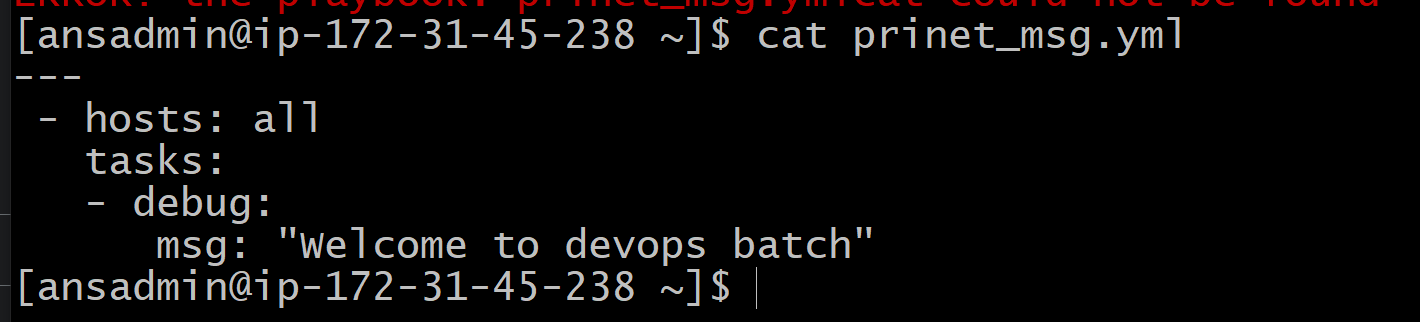
1. ansible localhost -m debug -a "var=groups.keys()"
2. ansible

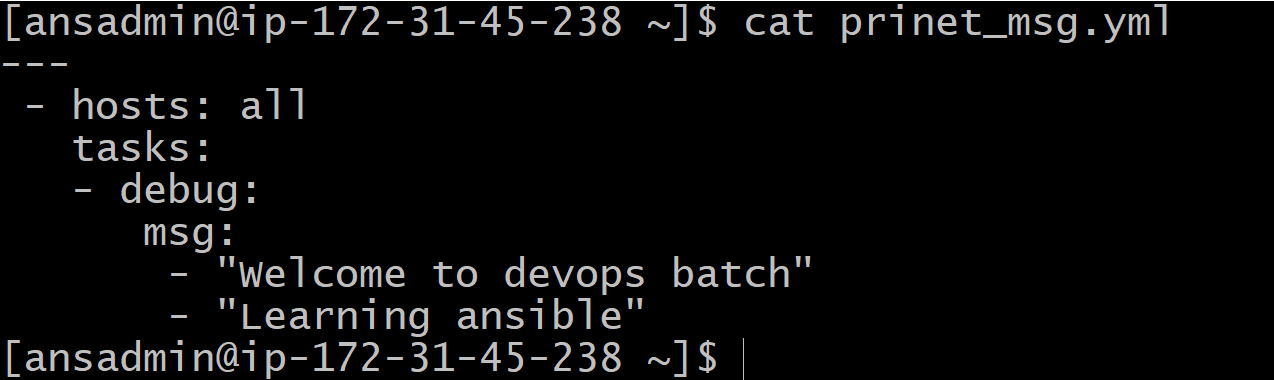
Debug module:





Print any message (ex: welcome to Playbooks) using playbook





**Scalar** representation: If have only one line value than we can call as scalar representation.

Key: value

X: 35

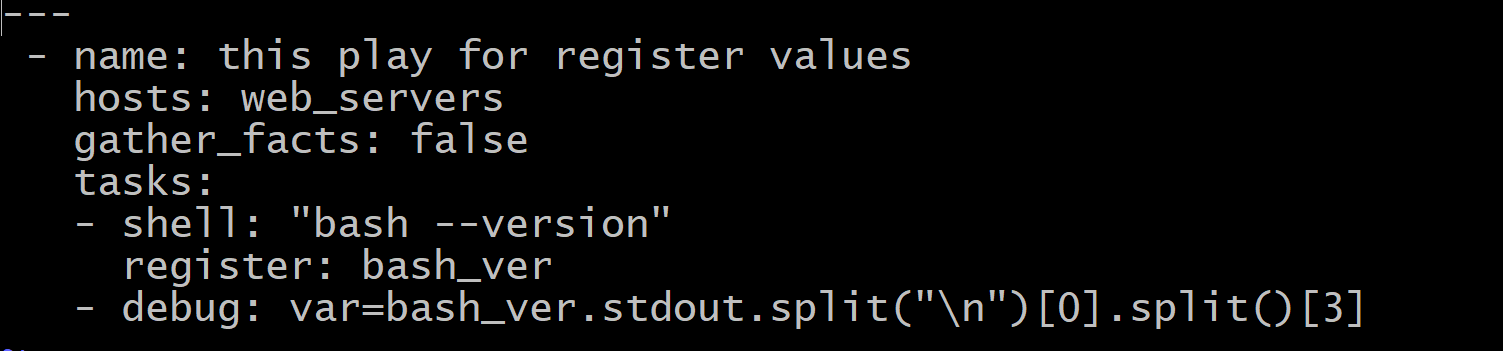
Y=99.9

Vars:

Declaring variables:

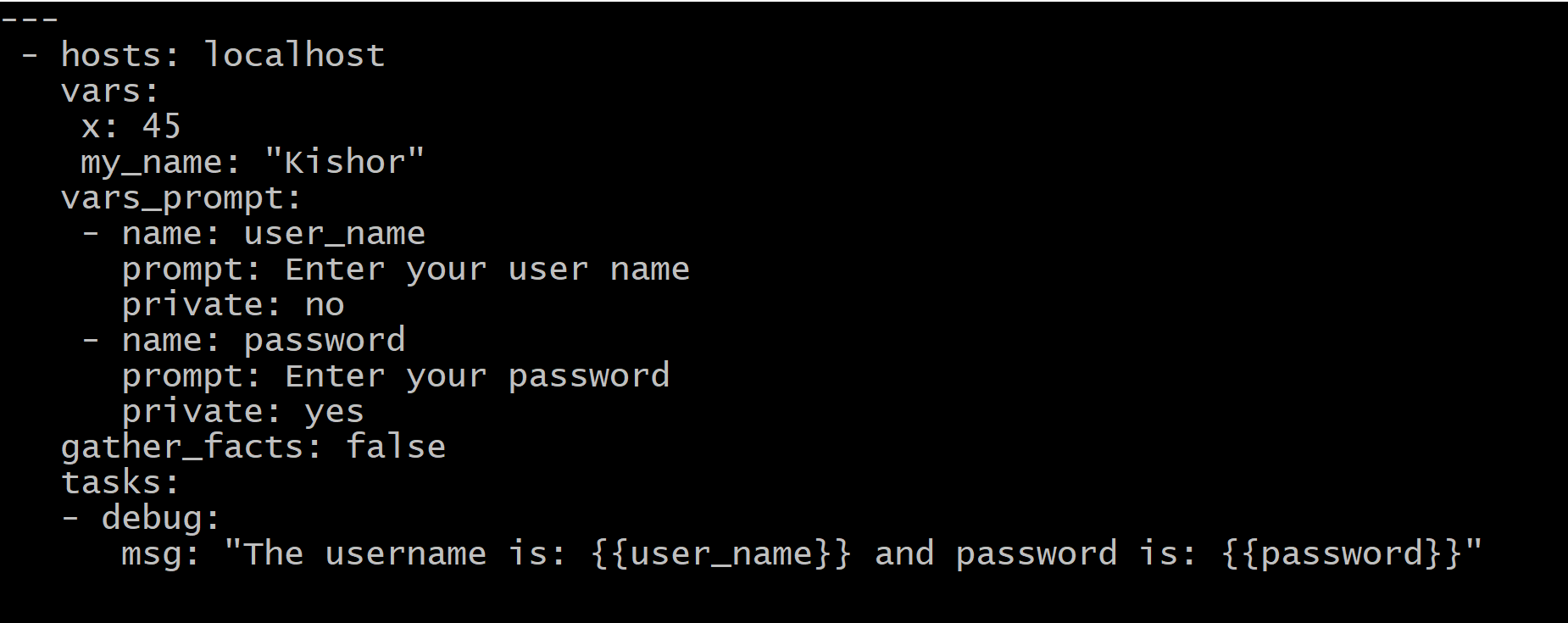
1. Variable and data types: notepad
2. Data Structures/ Data Collections: notepad
3. simple playbook to understand the usage of register:

bash\_ver.get(‘stdout’) or bash\_ver[‘stdout’] or bash\_ver.stdout,

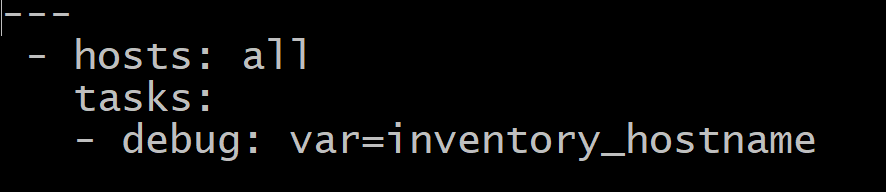


1. How to read a variable and print using ansible playbook ?

While running playbook it will take user details.



working with inventory\_hostname and hostvars variables:

Q

 Reusable tasks with import and include:

By using of 2 methods we can pull the other files to our yaml.

Import

Include

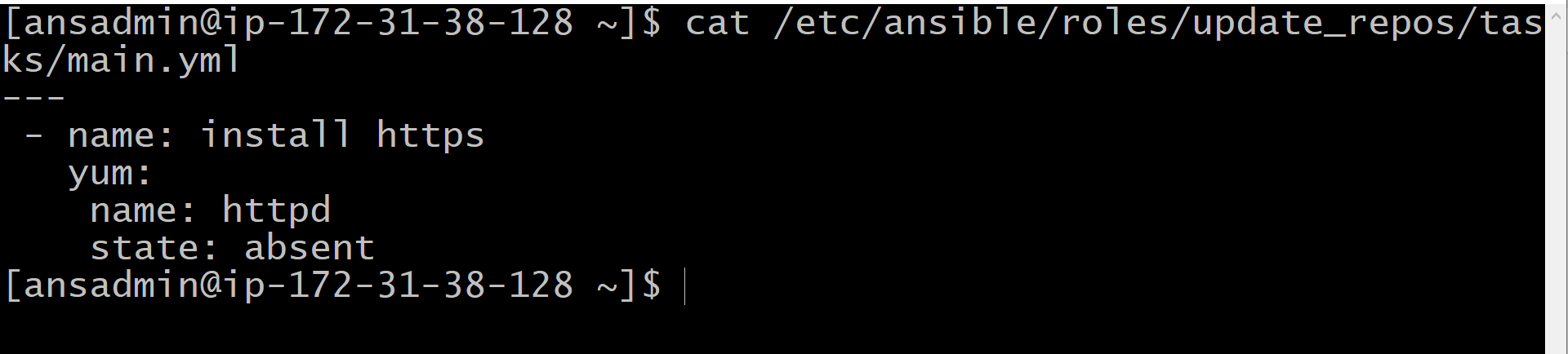
Import\_tasks: install\_java\_RedHat.yml

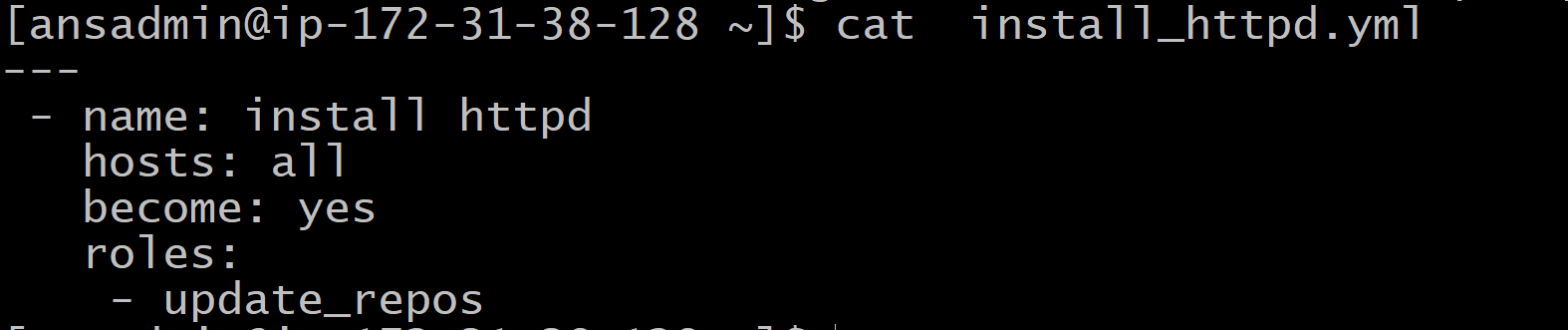
**Ansible\_Roles:**

Reuse tasks can be use.

Ansible-galaxy init update\_repos --offline







**Roles:**

