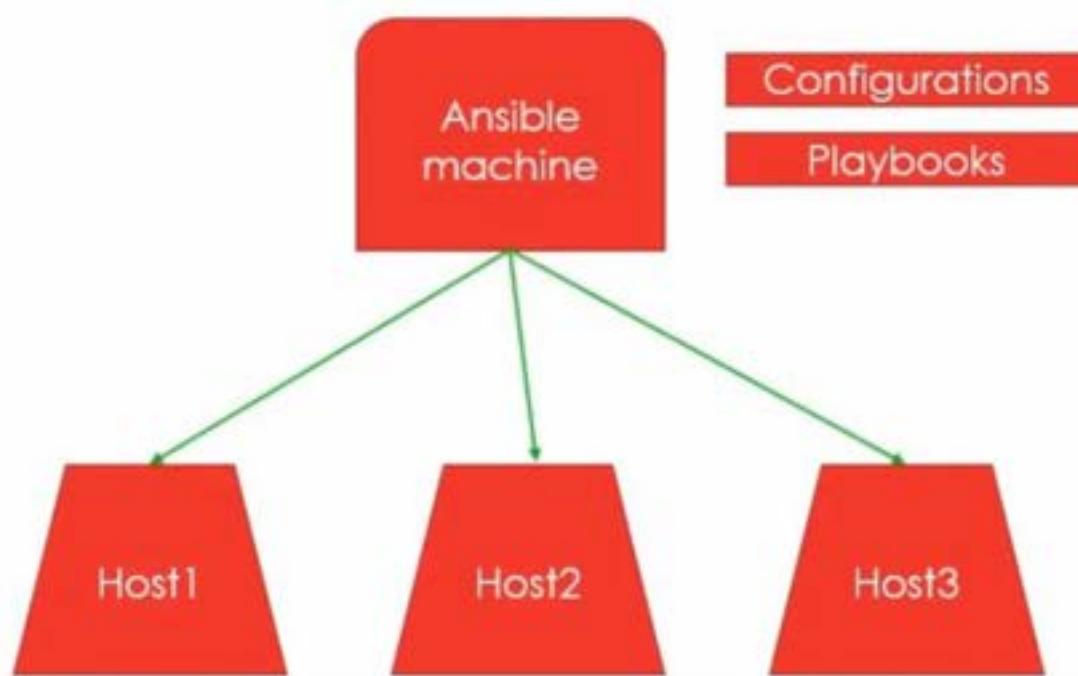


Hello Friends, Welcome back to my channel

- Today we are going to see about Ansible. From this tutorial we will start our learning on Ansible
- If you don't know what is Ansible!, Ansible is an open source configuration and application deployment tool.
- Ansible doesn't need any agents on the host machine to communicate, we will see more about these in upcoming tutorials
- Ansible can run on both windows and unix based machine.
- Today we are going to see how quickly we can install Ansible on CentOS and Ubuntu machine, we will also see some of the configuration files in Ansible.



Ansible Tutorial | Is It That Easy To Setup Ansible ? /etc/ansible/hosts & ansible.cfg

```
[root@localhost ~]# pip install ansible
-bash: pip: command not found
[root@localhost ~]# sudo yum install epel-release
```

I

◀ ▶ ⏪ 🔍 4:22 / 13:35 • Installation >

Scroll for details

CC ⚙ ⌂

Ansible Tutorial | Is It That Easy To Setup Ansible ? /etc/ansible/hosts & ansible.cfg

```
[root@localhost ~]# pip install ansible  
-bash: pip: command not found  
[root@localhost ~]# sudo yum install epel-release
```

Settings

◀ ▶ ⏪ 🔍 4:20 / 13:35 • Installation >

Scroll for details



Ansible Tutorial | Is It That Easy To Setup Ansible ? /etc/ansible/hosts & ansible.cfg

```
[root@localhost ~]# pip install ansible
-bash: pip: command not found
[root@localhost ~]# sudo yum install epel-release
Loaded plugins: fastestmirror
Determining fastest mirrors
epel/x86_64/metalink                                         | 8.3 KB  00:00:00
 * base: mirrors.piconets.webwerks.in
 * epel: download.nus.edu.sg
 * extras: mirrors.piconets.webwerks.in
 * updates: mirrors.piconets.webwerks.in
base                                                       | 3.6 KB  00:00:00
docker-ce-stable                                         | 3.5 KB  00:00:00
epel                                                       | 4.7 KB  00:00:00
extras                                                     | 2.9 KB  00:00:00
jenkins                                                   | 2.9 KB  00:00:00
updates                                                   | 2.9 KB  00:00:00
(1/7): extras/7/x86_64/primary_db                         | 205 kB  00:00:00
(2/7): docker-ce-stable/x86_64/primary_db                 | 45 kB   00:00:00
(3/7): jenkins/primary_db                                 | 33 kB   00:00:01
(4/7): epel/x86_64/group_gz                            | 95 kB   00:00:02
(5/7): updates/7/x86_64/primary_db                        | 3.0 MB  00:00:02
(6/7): epel/x86_64/updateinfo                           | 1.0 MB  00:00:07
(7/7): epel/x86_64/primary_db                           | 6.8 MB  00:00:49
Package epel-release-7-12.noarch already installed and latest version
Nothing to do
[root@localhost ~]# sudo yum install python-pip
```

Ansible Tutorial | Is It That Easy To Setup Ansible ? /etc/ansible/hosts & ansible.cfg

```
[root@localhost ~]# pip --version  
pip 8.1.2 from /usr/lib/python2.7/site-packages (python 2.7)  
[root@localhost ~]#
```

Play (k)



5:35 / 13:35 • Installation >

Scroll for details



Ansible Tutorial | Is It That Easy To Setup Ansible ? /etc/ansible/hosts & ansible.cfg

```
[root@localhost ~]# pip install ansible  
Collecting ansible
```

Play (k)



5:45 / 13:35 • Installation >

Scroll for details



Ansible Tutorial | Is It That Easy To Setup Ansible ? /etc/ansible/hosts & ansible.cfg

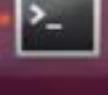
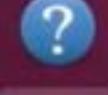
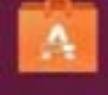
```
[root@localhost ~]# ansible --version
ansible 2.9.10
  config file = None
  configured module search path = [u'/root/.ansible/plugins/modules', u'/usr/share/ansible/plugins/modules']
  ansible python module location = /usr/lib/python2.7/site-packages/ansible
  executable location = /usr/bin/ansible
  python version = 2.7.5 (default, Oct 30 2018, 23:45:53) [GCC 4.8.5 20150623 (Red Hat 4.8.5-36)]
[root@localhost ~]#
```

Ansible Tutorial | Is It That Easy To Setup Ansible ? /etc/ansible/hosts & ansible.cfg



```
ubuntu@ubuntu:~  
File Edit View Search Terminal Help  
ubuntu@ubuntu:~$ sudo apt-get install ansible  
Reading package lists... Done  
Building dependency tree  
Reading state information... Done
```

Trash



Ansible Tutorial | Is It That Easy To Setup Ansible ? /etc/ansible/hosts & ansible.cfg



```
ubuntu@ubuntu:~  
File Edit View Search Terminal Help  
ubuntu@ubuntu:~$ ansible --version
```

◀ ▶ ⏪ ⏩ 🔍 8:31 / 13:35 • Installation >

Scroll for details

CC HD

Ansible Tutorial | Is It That Easy To Setup Ansible ? /etc/ansible/hosts & ansible.cfg



```
ubuntu@ubuntu:~$ File Edit View Search Terminal Help
ubuntu@ubuntu:~$ ansible --version
ansible 2.5.1
  config file = /etc/ansible/ansible.cfg
  configured module search path = [u'/home/ubuntu/.ansible/plugins/modules', u'/usr/share/ansible/plugins/modules']
  ansible python module location = /usr/lib/python2.7/dist-packages/ansible
  executable location = /usr/bin/ansible
  python version = 2.7.17 (default, Apr 15 2020, 17:20:14) [GCC 7.5.0]
ubuntu@ubuntu:~$
```

Trash

>-

•••

◀ ▶ ▶| 🔍 8:47 / 13:35 • Installation >

Scroll for details



Ansible Tutorial | Is It That Easy To Setup Ansible ? /etc/ansible/hosts & ansible.cfg



```
ubuntu@ubuntu: /etc/ansible
File Edit View Search Terminal Help
ubuntu@ubuntu:~$ ansible --version
ansible 2.5.1
  config file = /etc/ansible/ansible.cfg
  configured module search path = [u'/home/ubuntu/.ansible/plugins/modules', u'/usr/share/ansible/plugins/modules']
  ansible python module location = /usr/lib/python2.7/dist-packages/ansible
  executable location = /usr/bin/ansible
  python version = 2.7.17 (default, Apr 15 2020, 17:20:14) [GCC 7.5.0]
ubuntu@ubuntu:~$ cd /etc/ansible/
ubuntu@ubuntu:/etc/ansible$ ls
ansible.cfg  hosts
ubuntu@ubuntu:/etc/ansible$
```

A Ansible Tutorial | Is It That Easy To Setup Ansible ? /etc/ansible/hosts & ansible.cfg



```
ubuntu@ubuntu: /etc/ansible
File Edit View Search Terminal Help
ubuntu@ubuntu:~$ ansible --version
ansible 2.5.1
  config file = /etc/ansible/ansible.cfg
  configured module search path = [u'/home/ubuntu/.ansible/plugins/modules', u'/usr/share/ansible/plugins/modules']
  ansible python module location = /usr/lib/python2.7/dist-packages/ansible
  executable location = /usr/bin/ansible
  python version = 2.7.17 (default, Apr 15 2020, 17:20:14) [GCC 7.5.0]
ubuntu@ubuntu:~$ cd /etc/ansible/
ubuntu@ubuntu:/etc/ansible$ ls
ansible.cfg  hosts
ubuntu@ubuntu:/etc/ansible$ cat hosts
```

A Ansible Tutorial | Is It That Easy To Setup Ansible ? /etc/ansible/hosts & ansible.cfg



Trash

```
ubuntu@ubuntu: /etc/ansible
File Edit View Search Terminal Help
#beta.example.org
#192.168.1.100
#192.168.1.110

# If you have multiple hosts following a pattern you can specify
# them like this:

#www[001:006].example.com

# Ex 3: A collection of database servers in the 'dbservers' group

#[dbservers]
#
#db01.intranet.mydomain.net
#db02.intranet.mydomain.net
#10.25.1.56
#10.25.1.57

# Here's another example of host ranges, this time there are no
# leading 0s:

#db-[99:101]-node.example.com

ubuntu@ubuntu:/etc/ansible$
```

Ansible Tutorial | Is It That Easy To Setup Ansible ? /etc/ansible/hosts & ansible.cfg



Trash

```
ubuntu@ubuntu: /etc/ansible
File Edit View Search Terminal Help
# - Groups of hosts are delimited by [header] elements
# - You can enter hostnames or ip addresses
# - A hostname/ip can be a member of multiple groups

# Ex 1: Ungrouped hosts, specify before any group headers.

#green.example.com
#blue.example.com
#192.168.100.1
#192.168.100.10

# Ex 2: A collection of hosts belonging to the 'webservers' group

#[webservers]
#alpha.example.org
#beta.example.org
#192.168.1.100
#192.168.1.110

# If you have multiple hosts following a pattern you can specify
# them like this:

#www[001:006].example.com
```

Ansible Tutorial | Is It That Easy To Setup Ansible ? /etc/ansible/hosts & ansible.cfg



```
ubuntu@ubuntu: /etc/ansible
File Edit View Search Terminal Help
#
#   - Comments begin with the '#' character
#   - Blank lines are ignored
#   - Groups of hosts are delimited by [header] elements
#   - You can enter hostnames or ip addresses
#   - A hostname/ip can be a member of multiple groups

# Ex 1: Ungrouped hosts, specify before any group headers.

#green.example.com
#blue.example.com
#192.168.100.1
#192.168.100.10

# Ex 2: A collection of hosts belonging to the 'webservers' group

#[webservers]
#alpha.example.org
#beta.example.org
#192.168.1.100
#192.168.1.110

# If you have multiple hosts following a pattern you can specify
# them like this:
```

Ansible Tutorial | Is It That Easy To Setup Ansible ? /etc/ansible/hosts & ansible.cfg



```
ubuntu@ubuntu: /etc/ansible
File Edit View Search Terminal Help
ubuntu@ubuntu:/etc/ansible$ cat ansible.cfg
```

Trash



◀ ▶ ⏪ ⏩ 🔍 10:17 / 13:35 • Installation >

Scroll for details

CC HD

Ansible Tutorial | Is It That Easy To Setup Ansible ? /etc/ansible/hosts & ansible.cfg



```
ubuntu@ubuntu: /etc/ansible
File Edit View Search Terminal Help

# Set this to yes to allow libvirt_lxc connections to work without SELinux.
#libvirt_lxc_noseclabel = yes

[colors]
#highlight = white
#verbose = blue
#warn = bright purple
#error = red
#debug = dark gray
#deprecate = purple
#skip = cyan
#unreachable = red
#ok = green
#changed = yellow
#diff_add = green
#diff_remove = red
#diff_lines = cyan

[diff]
# Always print diff when running ( same as always running with -D/--diff )
# always = no
```

Ansible Tutorial | Is It That Easy To Setup Ansible ? /etc/ansible/hosts & ansible.cfg



```
ubuntu@ubuntu: /etc/ansible
File Edit View Search Terminal Help
# instead. These warnings can be silenced by adjusting the following
# setting or adding warn=yes or warn=no to the end of the command line
# parameter string. This will for example suggest using the git module
# instead of shelling out to the git command.
# command_warnings = False

# set plugin path directories here, separate with colons
#action_plugins      = /usr/share/ansible/plugins/action
#cache_plugins       = /usr/share/ansible/plugins/cache
#callback_plugins   = /usr/share/ansible/plugins/callback
#connection_plugins = /usr/share/ansible/plugins/connection
#lookup_plugins     = /usr/share/ansible/plugins/lookup
#inventory_plugins  = /usr/share/ansible/plugins/inventory
#vars_plugins        = /usr/share/ansible/plugins/vars
#filter_plugins      = /usr/share/ansible/plugins/filter
#test_plugins        = /usr/share/ansible/plugins/test
#terminal_plugins   = /usr/share/ansible/plugins/terminal
#strategy_plugins   = /usr/share/ansible/plugins/strategy

# by default, ansible will use the 'linear' strategy but you may want to try
# another one
#strategy = free
```

Ansible Tutorial | Is It That Easy To Setup Ansible ? /etc/ansible/hosts & ansible.cfg



```
ubuntu@ubuntu: /etc/ansible
File Edit View Search Terminal Help
# some basic default values...

#inventory      = /etc/ansible/hosts
#library        = /usr/share/my_modules/
#module_utils   = /usr/share/my_module_utils/
#remote_tmp     = ~/.ansible/tmp
#local_tmp      = ~/.ansible/tmp
#plugin_filters_cfg = /etc/ansible/plugin_filters.yml
#forks          = 5
#poll_interval  = 15
#sudo_user      = root
#ask_sudo_pass  = True
#ask_pass        = True
#transport      = smart
#remote_port    = 22
#module_lang    = C
#module_set_locale = False

# plays will gather facts by default, which contain information about
# the remote system.
#
# smart - gather by default, but don't regather if already gathered
# implicit - gather by default, turn off with gather_facts: False
# explicit - do not gather by default, must say gather_facts: True
```

An Ansible Tutorial | Is It That Easy?

youtube.com is now full screen

Exit Full Screen (Esc)

ansible.cfg

```
[root@localhost ~]# cd /etc/ansible/  
[root@localhost ansible]# ls  
ansible.cfg hosts roles  
[root@localhost ansible]#
```

Ansible Tutorial | Is It That Easy To Setup Ansible ? /etc/ansible/hosts & ansible.cfg

```
[root@localhost ~]# cd /etc/ansible/  
[root@localhost ansible]# ls  
ansible.cfg hosts roles  
[root@localhost ansible]# cat ansible.cfg
```

Play (k)



11:33 / 13:35 • Installation >

Scroll for details



Ansible Tutorial | Is It That Easy To Setup Ansible ? /etc/ansible/hosts & ansible.cfg

```
[selinux]
# file systems that require special treatment when dealing with security context
# the default behaviour that copies the existing context or uses the user default
# needs to be changed to use the file system dependent context.
#special_context_filesystems=nfs,vboxsf,fuse,ramfs,9p,vfat

# Set this to yes to allow libvirt_lxc connections to work without SELinux.
#libvirt_lxc_noseclabel = yes

[colors]
#highlight = white
#verbose = blue
#warn = bright purple
#error = red
#debug = dark gray
#deprecate = purple
#skip = cyan
#unreachable = red
#ok = green
#changed = yellow
#diff_add = green
#diff_remove = red
#diff_lines = cyan

[diff]
# Always print diff when running ( same as always running with -D/--diff )
# always = no

# Set how many context lines to show in diff
# context = 3
```

[root@localhost ansible]# 11:34 / 13:35 • Installation > Scroll for details

Ansible Tutorial | Is It That Easy To Setup Ansible ? /etc/ansible/hosts & ansible.cfg

```
[root@localhost ansible]# cd ..
[root@localhost etc]# cd ..
[root@localhost /]# clear
[root@localhost /]# cd /etc/ansible/
[root@localhost ansible]# ls
ansible.cfg  hosts  roles
[root@localhost ansible]# cat ansible.cfg
# config file for ansible -- https://ansible.com/
#
# nearly all parameters can be overridden in ansible-playbook
# or with command line flags. ansible will read ANSIBLE_CONFIG,
# ansible.cfg in the current working directory, .ansible.cfg in
# the home directory or /etc/ansible/ansible.cfg, whichever it
# finds first

[defaults]

# some basic default values...

#inventory      = /etc/ansible/hosts
#library        = /usr/share/my_modules/
#module_utils   = /usr/share/my_module_utils/
#remote_tmp     = ~/.ansible/tmp
#local_tmp      = ~/.ansible/tmp
#plugin_filters_cfg = /etc/ansible/plugin_filters.yml
#forks          = 5
#poll_interval  = 15
#sudo_user      = root
#ask_sudo_pass  = True
#ask_pass       = True
#transport      = smart
#remote_port    = 22
```



11:42 / 13:35 • Installation >

Scroll for details



Ansible Tutorial | Is It That Easy To Setup Ansible ? /etc/ansible/hosts & ansible.cfg



```
# Ansible facts are available inside the ansible_facts.* dictionary
# namespace. This setting maintains the behaviour which was the default prior
# to 2.5, duplicating these variables into the main namespace, each with a
# prefix of 'ansible_'.
# This variable is set to True by default for backwards compatibility. It
# will be changed to a default of 'False' in a future release.
# ansible_facts.
# inject_facts_as_vars = True

# additional paths to search for roles in, colon separated
#roles_path      = /etc/ansible/roles

# uncomment this to disable SSH key host checking
#host_key_checking = False

# change the default callback, you can only have one 'stdout' type enabled at a time.
#stdout_callback = skippy

## Ansible ships with some plugins that require whitelisting,
## this is done to avoid running all of a type by default.
## These setting lists those that you want enabled for your system.
## Custom plugins should not need this unless plugin author specifies it.

# enable callback plugins, they can output to stdout but cannot be 'stdout' type.
#callback_whitelist = timer, mail

# Determine whether includes in tasks and handlers are "static" by
# default. As of 2.0, includes are dynamic by default. Setting these
# values to True will make includes behave more like they did in the
# 1.x versions.
#task_includes_static = False
#handler_includes_static = False
```

◀ ▶ ▶ 🔍 12:03 / 13:35 • Installation >

Scroll for details



Ansible Tutorial | Is It That Easy To Setup Ansible ? /etc/ansible/hosts & ansible.cfg

```
# handler_includes_static = False

# Controls if a missing handler for a notification event is an error or a warning
#error_on_missing_handler = True

# change this for alternative sudo implementations
#sudo_exe = sudo

# What flags to pass to sudo
# WARNING: leaving out the defaults might create unexpected behaviours
#sudo_flags = -H -S -n

# SSH timeout
#timeout = 10

# default user to use for playbooks if user is not specified
# (/usr/bin/ansible will use current user as default)
#remote_user = root

# logging is off by default unless this path is defined
# if so defined, consider logrotate
#log_path = /var/log/ansible.log

# default module name for /usr/bin/ansible
#module_name = command

# use this shell for commands executed under sudo
# you may need to change this to bin/bash in rare instances
# if sudo is constrained
#executable = /bin/sh

# if inventory variables overlap, which one wins?
# or are hash values merged together? The default is 'Scroll for details'
```

◀ ▶ ⏪ ⏩ ⏴ 12:08 / 13:35 • Installation >



Ansible Tutorial | Is It That Easy To Setup Ansible ? /etc/ansible/hosts & ansible.cfg

```
[selinux]
# file systems that require special treatment when dealing with security context
# the default behaviour that copies the existing context or uses the user default
# needs to be changed to use the file system dependent context.
#special_context_filesystems=nfs,vboxsf,fuse,ramfs,9p,vfat

# Set this to yes to allow libvirt_lxc connections to work without SELinux.
#libvirt_lxc_noseclabel = yes

[colors]
#highlight = white
#verbose = blue
#warn = bright purple
#error = red
#debug = dark gray
#deprecate = purple
#skip = cyan
#unreachable = red
#ok = green
#changed = yellow
#diff_add = green
#diff_remove = red
#diff_lines = cyan

[diff]
# Always print diff when running ( same as always running with -D/--diff )
# always = no

# Set how many context lines to show in diff
# context = ?
```

Subtitles/closed captions (c)

[root@localhost ansible]

Scroll for details



Ansible Tutorial | Is It That Easy To Setup Ansible ? /etc/ansible/hosts & ansible.cfg

[root@localhost ansible]# cat hosts

```
Ansible Tutorial | Is It That Easy To Setup Ansible ? /etc/ansible/hosts & ansible.cfg
```

```
## green.example.com
## blue.example.com
## 192.168.100.1
## 192.168.100.10

# Ex 2: A collection of hosts belonging to the 'webservers' group

## [webservers]
## alpha.example.org
## beta.example.org
## 192.168.1.100
## 192.168.1.110

# If you have multiple hosts following a pattern you can specify
# them like this:

## www[001:006].example.com

# Ex 3: A collection of database servers in the 'dbservers' group

## [dbservers]
##
## db01.intranet.mydomain.net
## db02.intranet.mydomain.net
## 10.25.1.56
## 10.25.1.57

# Here's another example of host ranges, this time there are no
# leading 0s:

## db-[99:101]-node.example.com
```

[root@localhost ansible] 12:36 / 13:35 • Installation >

Scroll for details

CC HD

Ansible Tutorial | Is It That Easy To Setup Ansible ? /etc/ansible/hosts & ansible.cfg

```
[root@localhost ansible]# cat hosts
# This is the default ansible 'hosts' file.
#
# It should live in /etc/ansible/hosts
#
# - Comments begin with the '#' character
# - Blank lines are ignored
# - Groups of hosts are delimited by [header] elements
# - You can enter hostnames or ip addresses
#   - A hostname/ip can be a member of multiple groups
#
# Ex 1: Ungrouped hosts, specify before any group headers.

## green.example.com
## blue.example.com
## 192.168.100.1
## 192.168.100.10

# Ex 2: A collection of hosts belonging to the 'webservers' group

## [webservers]
## alpha.example.org
## beta.example.org
## 192.168.1.100
## 192.168.1.110

# If you have multiple hosts following a pattern you can specify
# them like this:

## www[001:006].example.com

# Ex 3: A collection of database servers in the 'dbservers' group
```

Hello Friends, Welcome back to my channel

- In Todays tutorial we are going to see more about Ansible. In our last video we have seen how to install Ansible on Centos and Ubuntu machine.
- Let us see how we can make use of Ad-hoc ansible commands in this lecture.
- Ad-hoc commands are great for tasks which are not repeated and done very rarely. We will see about how to run the commands in Ansible and make use of different modules in this video.

Ansible Ad Hoc Commands | Ansible Modules : File, Ping, Copy, Yum, User, Service, Shell |Th...



```
[root@ansible ~]# sudo yum install epel-release
Loaded plugins: fastestmirror
Determining fastest mirrors
```

◀ ▶ ▶| 🔍 2:48 / 30:03

Scroll for details



Ansible Ad Hoc Commands | Ansible Modules : File, Ping, Copy, Yum, User, Service, Shell |Th...



```
[root@ansible ~]# ansible --version
ansible 2.9.10
  config file = /etc/ansible/ansible.cfg
  configured module search path = [u'/root/.ansible/plugins/modules', u'/usr/share/ansible/plugins/modules']
  ansible python module location = /usr/lib/python2.7/site-packages/ansible
  executable location = /usr/bin/ansible
  python version = 2.7.5 (default, Oct 30 2018, 23:45:53) [GCC 4.8.5 20150623 (Red Hat 4.8.5-36)]
[root@ansible ~]#
```

Ansible Ad Hoc Commands | Ansible Modules : File, Ping, Copy, Yum, User, Service, Shell |Th...



```
[root@ansible ~]# ansible --version
ansible 2.9.10
  config file = /etc/ansible/ansible.cfg
  configured module search path = [u'/root/.ansible/plugins/modules', u'/usr/share/ansible/plugins/modules']
  ansible python module location = /usr/lib/python2.7/site-packages/ansible
  executable location = /usr/bin/ansible
  python version = 2.7.5 (default, Oct 30 2018, 23:45:53) [GCC 4.8.5 20150623 (Red Hat 4.8.5-36)]
[root@ansible ~]# cd /etc/ansible/
[root@ansible ansible]#
```

Ansible Ad Hoc Commands | Ansible Modules : File, Ping, Copy, Yum, User, Service, Shell |Th...



```
[root@ansible ~]# ansible --version
ansible 2.9.10
  config file = /etc/ansible/ansible.cfg
  configured module search path = [u'/root/.ansible/plugins/modules', u'/usr/share/ansible/plugins/modules']
  ansible python module location = /usr/lib/python2.7/site-packages/ansible
  executable location = /usr/bin/ansible
  python version = 2.7.5 (default, Oct 30 2018, 23:45:53) [GCC 4.8.5 20150623 (Red Hat 4.8.5-36)]
[root@ansible ~]# cd /etc/ansible/
[root@ansible ansible]# ls
ansible.cfg  hosts  roles
[root@ansible ansible]#
```

Ansible Ad Hoc Commands | Ansible Modules : File, Ping, Copy, Yum, User, Service, Shell |Th...



```
[root@ansible ~]# ansible --version
ansible 2.9.10
  config file = /etc/ansible/ansible.cfg
  configured module search path = [u'/root/.ansible/plugins/modules', u'/usr/share/ansible/plugins/modules']
  ansible python module location = /usr/lib/python2.7/site-packages/ansible
  executable location = /usr/bin/ansible
  python version = 2.7.5 (default, Oct 30 2018, 23:45:53) [GCC 4.8.5 20150623 (Red Hat 4.8.5-36)]
[root@ansible ~]# cd /etc/ansible/
[root@ansible ansible]# ls
ansible.cfg  hosts  roles
[root@ansible ansible]# vi hosts
```

Ansible Ad Hoc Commands | Ansible Modules : File, Ping, Copy, Yum, User, Service, Shell | Th...



```
# This is the default ansible 'hosts' file.

# It should live in /etc/ansible/hosts

# - Comments begin with the '#' character
# - Blank lines are ignored
# - Groups of hosts are delimited by [header] elements
# - You can enter hostnames or ip addresses
# - A hostname/ip can be a member of multiple groups

# Ex 1: Ungrouped hosts, specify before any group headers.

## green.example.com
## blue.example.com
## 192.168.100.1
## 192.168.100.10

# Ex 2: A collection of hosts belonging to the 'webservers' group

## [webservers]
## alpha.example.org
## beta.example.org
## 192.168.1.100
## 192.168.1.110

# If you have multiple hosts following a pattern you can specify
# them like this:

## www[001:006].example.com
```

```
# Ex 3: A collection of database servers in the 'dbservers' group
```

"hosts" 4M, 1016C 4:58 / 30:03

Scroll for details



Ansible Ad Hoc Commands | Ansible Modules : File, Ping, Copy, Yum, User, Service, Shell | Th...



```
# This is the default ansible 'hosts' file.  
#  
# It should live in /etc/ansible/hosts  
  
# - Comments begin with the '#' character  
# - Blank lines are ignored  
# - Groups of hosts are delimited by [header] elements  
# - You can enter hostnames or ip addresses  
# - A hostname/ip can be a member of multiple groups  
  
# Ex 1: Ungrouped hosts, specify before any group headers.
```

```
## green.example.com  
## blue.example.com  
## 192.168.100.1  
## 192.168.100.10
```

```
# Ex 2: A collection of hosts belonging to the 'webservers' group
```

```
## [webservers]  
## alpha.example.org  
## beta.example.org  
## 192.168.1.100  
## 192.168.1.110  
  
# If you have multiple hosts following a pattern you can specify  
# them like this:
```

```
## www[001:006].example.com
```

```
# Ex 3: A collection of database servers in the 'dbservers' group
```

Ansible Ad Hoc Commands | Ansible Modules : File, Ping, Copy, Yum, User, Service, Shell | Th...



```
# This is the default ansible 'hosts' file.  
#  
# It should live in /etc/ansible/hosts  
  
# - Comments begin with the '#' character  
# - Blank lines are ignored  
# - Groups of hosts are delimited by [header] elements  
# - You can enter hostnames or ip addresses  
# - A hostname/ip can be a member of multiple groups
```

```
# Ex 1: Ungrouped hosts, specify before any group headers.
```

```
## green.example.com  
## blue.example.com  
## servers]  
## alpha.example.org
```

```
# Ex 2: A collection of hosts belonging to the 'webservers' group
```

```
[testservers]  
192.168.145.145  
  
## [webservers]  
## alpha.example.org  
## beta.example.org  
## 192.168.1.100  
## 192.168.1.110
```

```
# If you have multiple hosts following a pattern you can specify  
# them like this:
```

Ansible Ad Hoc Commands | Ansible Modules : File, Ping, Copy, Yum, User, Service, Shell | Th...



```
# This is the default ansible 'hosts' file.  
#  
# It should live in /etc/ansible/hosts  
#  
# - Comments begin with the '#' character  
# - Blank lines are ignored  
# - Groups of hosts are delimited by [header] elements  
# - You can enter hostnames or ip addresses  
# - A hostname/ip can be a member of multiple groups  
  
# Ex 1: Ungrouped hosts, specify before any group headers.  
  
## green.example.com  
## blue.example.com  
## servers]  
## alpha.example.org  
  
# Ex 2: A collection of hosts belonging to the 'webservers' group  
  
[testservers]  
192.168.145.145  
  
## [webservers]  
## alpha.example.org  
## beta.example.org  
## 192.168.1.100  
## 192.168.1.110  
  
# If you have multiple hosts following a pattern you can specify  
# them like this:
```

Ansible Ad Hoc Commands | Ansible Modules : File, Ping, Copy, Yum, User, Service, Shell | Th... ⓘ ⓘ ⓘ

```
[root@ansible1 ansible]# ansible testservers -m ping
```

I

7:13 / 30:03

Scroll for details

CC ⚙️ 🔍

Ansible Ad Hoc Commands | Ansible Modules : File, Ping, Copy, Yum, User, Service, Shell | Th...



```
[root@ansible1 ansible]# ansible testservers -m ping
The authenticity of host '192.168.145.145 (192.168.145.145)' can't be established.
ECDSA key fingerprint is SHA256:aDT3dCzT1HpEfgPbF+Ju20bMMkF1dXS67Q36h1Nf3mo.
ECDSA key fingerprint is MD5:4f:eb:02:0b:8f:e5:e1:1c:66:37:e3:ac:53:52:be:1b.
Are you sure you want to continue connecting (yes/no)? yes
192.168.145.145 | UNREACHABLE! => |
  "changed": false,
  "msg": "Failed to connect to the host via ssh: Warning: Permanently added '192.168.145.145' (ECDSA) to the list of known hosts.\r\nPermission denied (publickey,gssapi-keyex,gssapi-with-mic,password).",
  "unreachable": true
}
[root@ansible1 ansible]#
```

Ansible Ad Hoc Commands | Ansible Modules : File, Ping, Copy, Yum, User, Service, Shell |Th...



```
[root@ansible ~]# ssh-keygen
Generating public/private rsa key pair.
Enter file in which to save the key (/root/.ssh/id_rsa):
```

Ansible Ad Hoc Commands | Ansible Modules : File, Ping, Copy, Yum, User, Service, Shell | Th...



```
[root@ansible ~]# ssh-keygen
Generating public/private rsa key pair.
Enter file in which to save the key (/root/.ssh/id_rsa):
Enter passphrase (empty for no passphrase):
Enter same passphrase again:
Your identification has been saved in /root/.ssh/id_rsa.
Your public key has been saved in /root/.ssh/id_rsa.pub.
The key fingerprint is:
SHA256:L+iL2zkcEflDqTVSkZH2uf467CbkDfsx6R1CGHctmTc root@ansible1
The key's randomart image is:
+---[RSA 2048]---+
|   o+*   |
| + O     + |
| O.o...= E |
| o o+o. o .|
| .S...    |
| ..oo..   |
| ..+.*. ,  |
| +o.+o*= . |
| o.=o ==+o |
+---[SHA256]---+
[root@ansible ~]#
```

Ansible Ad Hoc Commands | Ansible Modules : File, Ping, Copy, Yum, User, Service, Shell | Th...



```
[root@ansible ~]# ssh-keygen
Generating public/private rsa key pair.
Enter file in which to save the key (/root/.ssh/id_rsa):
Enter passphrase (empty for no passphrase):
Enter same passphrase again:
Your identification has been saved in /root/.ssh/id_rsa.
Your public key has been saved in /root/.ssh/id_rsa.pub.
The key fingerprint is:
SHA256:L+iL2zkcEf1DqTVSkZH2uf467CbkDfsx6R1CGHctmTc root@ansible1
The key's randomart image is:
+---[RSA 2048]---+
|   o+*   |
| + O     + |
| O.o..= E |
| o o+o. o .|
| .S...    |
| ..oo..   |
| ..+.*.  |
| +o.+o*= .|
| o.=o ==+o |
+---[SHA256]---+
[root@ansible ~]# /root/.ssh
-bash: /root/.ssh: No such file or directory
[root@ansible ~]# /root/.ssh
-bash: /root/.ssh: Is a directory
[root@ansible ~]# cd /root/.ssh
[root@ansible .ssh]#
```

Ansible Ad Hoc Commands | Ansible Modules : File, Ping, Copy, Yum, User, Service, Shell | Th...



```
[root@ansible ~]# ssh-keygen
Generating public/private rsa key pair.
Enter file in which to save the key (/root/.ssh/id_rsa):
Enter passphrase (empty for no passphrase):
Enter same passphrase again:
Your identification has been saved in /root/.ssh/id_rsa.
Your public key has been saved in /root/.ssh/id_rsa.pub.
The key fingerprint is:
SHA256:L+iL2zkcEf1DqTVSkZH2uf467CbkDfsx6R1CGHctmTc root@ansible1
The key's randomart image is:
+---[RSA 2048]---+
|   o+*   |
| + O     + |
| O.o..= E |
| o o+o. o .|
| .S...    |
| ..oo..   |
| ..+.*.  |
| +o.+o*= .|
| o.=o ==+o |
+---[SHA256]---+
[root@ansible ~]# /root/.ssh
-bash: /root/.ssh: No such file or directory
[root@ansible ~]# /root/.ssh
-bash: /root/.ssh: Is a directory
[root@ansible ~]# cd /root/.ssh
[root@ansible .ssh]# ls
id_rsa  id_rsa.pub  known_hosts
[root@ansible .ssh]#
```

Ansible Ad Hoc Commands | Ansible Modules : File, Ping, Copy, Yum, User, Service, Shell | Th...



```
[root@ansible2 ~]# ipaddr  
-bash: ipaddr: command not found  
[root@ansible2 ~]# ip addr  
1: lo: <LOOPBACK,UP,LOWER_UP> mtu 65536 qdisc noqueue state UNKNOWN group default qlen 1000  
    link/loopback 00:00:00:00:00:00 brd 00:00:00:00:00:00  
    inet 127.0.0.1/8 scope host lo  
        valid_lft forever preferred_lft forever  
    inet6 ::1/128 scope host  
        valid_lft forever preferred_lft forever  
2: ens33: <BROADCAST,MULTICAST,UP,LOWER_UP> mtu 1500 qdisc pfifo_fast state UP group default qlen 1000  
    link/ether 00:0c:29:23:a2:10 brd ff:ff:ff:ff:ff:ff  
    inet 192.168.145.145/24 brd 192.168.145.255 scope global noprefixroute dynamic ens33  
        valid_lft 1511sec preferred_lft 1511sec  
    inet6 fe80::5c6e:f999:e04:80a5/64 scope link noprefixroute  
        valid_lft forever preferred_lft forever  
[root@ansible2 ~]# cd /root/.ssh  
-bash: cd: /root/.ssh: No such file or directory  
[root@ansible2 ~]#
```

Ansible Ad Hoc Commands | Ansible Modules : File, Ping, Copy, Yum, User, Service, Shell | Th...



```
[root@ansible ~]# ssh-keygen
Generating public/private rsa key pair.
Enter file in which to save the key (/root/.ssh/id_rsa):
Enter passphrase (empty for no passphrase):
Enter same passphrase again:
Your identification has been saved in /root/.ssh/id_rsa.
Your public key has been saved in /root/.ssh/id_rsa.pub.
The key fingerprint is:
SHA256:L+il2zkcEf1DqTVSk2H2uf467CbkDfsx6R1CGHctmTc root@ansible1
The key's randomart image is:
+---[RSA 2048]---+
|   o+*   |
| + O     + |
| O.o..= E |
| o o+o. o .|
| .S...    |
| ..oo..   |
| ..+.** . |
| +o.+o*= . |
|   o.=o ==+o |
+---[SHA256]---+
[root@ansible ~]# /root/.ssh
-bash: /root/.ssh: No such file or directory
[root@ansible ~]# /root/.ssh
-bash: /root/.ssh: Is a directory
[root@ansible ~]# cd /root/.ssh
[root@ansible .ssh]# ls
id_rsa  id_rsa.pub  known_hosts
[root@ansible .ssh]# ssh-copy-id root@192.168.145.145
```

Ansible Ad Hoc Commands | Ansible Modules : File, Ping, Copy, Yum, User, Service, Shell | Th...



```
[root@ansible2 ~]# ipaddr  
-bash: ipaddr: command not found  
[root@ansible2 ~]# ip addr  
1: lo: <LOOPBACK,UP,LOWER_UP> mtu 65536 qdisc noqueue state UNKNOWN group default qlen 1000  
    link/loopback 00:00:00:00:00:00 brd 00:00:00:00:00:00  
    inet 127.0.0.1/8 scope host lo  
        valid_lft forever preferred_lft forever  
    inet6 ::1/128 scope host  
        valid_lft forever preferred_lft forever  
2: ens33: <BROADCAST,MULTICAST,UP,LOWER_UP> mtu 1500 qdisc pfifo_fast state UP group default qlen 1000  
    link/ether 00:0c:29:23:a2:10 brd ff:ff:ff:ff:ff:ff  
    inet 192.168.145.145/24 brd 192.168.145.255 scope global noprefixroute dynamic ens33  
        valid_lft 1511sec preferred_lft 1511sec  
    inet6 fe80::5c6e:f999:e04:80a5/64 scope link noprefixroute  
        valid_lft forever preferred_lft forever  
[root@ansible2 ~]# cd /root/.ssh  
-bash: cd: /root/.ssh: No such file or directory  
[root@ansible2 ~]# cd /root/.ssh  
[root@ansible2 .ssh]# ls  
authorized_keys  
[root@ansible2 .ssh]#
```

Ansible Ad Hoc Commands | Ansible Modules : File, Ping, Copy, Yum, User, Service, Shell | Th...



```
[root@ansible1 .ssh]# ansible testservers -m ping  
192.168.145.145 | SUCCESS => |  
  "ansible_facts": {  
    "discovered_interpreter_python": "/usr/bin/python"  
  },  
  "changed": false,  
  "ping": "pong"  
}  
[root@ansible1 .ssh]#
```

Ansible Ad Hoc Commands | Ansible Modules : File, Ping, Copy, Yum, User, Service, Shell | Th...



```
[root@ansible1 .ssh]# ansible testservers -m ping
192.168.145.145 | SUCCESS => 
  "ansible_facts": {
    "discovered_interpreter_python": "/usr/bin/python"
  },
  "changed": false,
  "ping": "pong"
}
[root@ansible1 .ssh]# ls
id_rsa  id_rsa.pub  known_hosts
[root@ansible1 .ssh]# cd ~
[root@ansible ~]# ls
```

Ansible Ad Hoc Commands | Ansible Modules : File, Ping, Copy, Yum, User, Service, Shell |Th...



```
[root@ansible1 .ssh]# ansible testservers -m ping
192.168.145.145 | SUCCESS => |
  "ansible_facts": {
    "discovered_interpreter_python": "/usr/bin/python"
  },
  "changed": false,
  "ping": "pong"
}
[root@ansible1 .ssh]# ls
id_rsa  id_rsa.pub  known_hosts
[root@ansible1 .ssh]# cd ~
[root@ansible1 ~]# ls
anaconda-ks.cfg
[root@ansible1 ~]#
```

Ansible Ad Hoc Commands | Ansible Modules : File, Ping, Copy, Yum, User, Service, Shell | Th...



```
[root@ansible1 .ssh]# ansible testservers -m ping
192.168.145.145 | SUCCESS => [
  "ansible_facts": {
    "discovered_interpreter_python": "/usr/bin/python"
  },
  "changed": false,
  "ping": "pong"
}
[root@ansible1 .ssh]# ls
id_rsa  id_rsa.pub  known_hosts
[root@ansible1 .ssh]# cd ~
[root@ansible1 ~]# ls
anaconda-ks.cfg
[root@ansible1 ~]# ansible testservers -m file -a "dest=/root/file state=directory"
192.168.145.145 | CHANGED => [
  "ansible_facts": {
    "discovered_interpreter_python": "/usr/bin/python"
  },
  "changed": true,
  "gid": 0,
  "group": "root",
  "mode": "0755",
  "owner": "root",
  "path": "/root/file",
  "secontext": "unconfined_u:object_r:admin_home_t:s0",
  "size": 6,
  "state": "directory",
  "uid": 0
}
[root@ansible1 ~]#
```

Ansible Ad Hoc Commands | Ansible Modules : File, Ping, Copy, Yum, User, Service, Shell |Th...



```
[root@ansible2 ~]# ls  
anaconda-ks.cfg  
[root@ansible2 ~]# ls  
anaconda-ks.cfg  file  
[root@ansible2 ~]#
```

Ansible Ad Hoc Commands | Ansible Modules : File, Ping, Copy, Yum, User, Service, Shell | Th...



```
"ping": "pong"
}
[root@ansible1 .ssh]# ls
id_rsa  id_rsa.pub  known_hosts
[root@ansible1 .ssh]# cd ~
[root@ansible1 ~]# ls
anaconda-ks.cfg
[root@ansible1 ~]# ansible testservers -m file -a "dest=/root/file state=directory"
192.168.145.145 | CHANGED => {
  "ansible_facts": {
    "discovered_interpreter_python": "/usr/bin/python"
  },
  "changed": true,
  "gid": 0,
  "group": "root",
  "mode": "0755",
  "owner": "root",
  "path": "/root/file",
  "secontext": "unconfined_u:object_r:admin_home_t:s0",
  "size": 6,
  "state": "directory",
  "uid": 0
}
[root@ansible1 ~]# ansible testservers -m file -a "dest=/root/file state=absent"
192.168.145.145 | CHANGED => {
  "ansible_facts": {
    "discovered_interpreter_python": "/usr/bin/python"
  },
  "changed": true,
  "path": "/root/file",
  "state": "absent"
}
```

[root@ansible1 ~]# 14:28 / 30:03

Scroll for details



Ansible Ad Hoc Commands | Ansible Modules : File, Ping, Copy, Yum, User, Service, Shell | Th... i b ↗

```
[root@ansible ~]# ansible testservers -m shell -a "mkdir test"
[WARNING]: Consider using the file module with state=directory rather than running 'mkdir'. If you need to use command because file is insufficient you can add 'warn: false' to this command task or set 'command_warnings=False' in ansible.cfg to get rid of this message.
192.168.145.145 | CHANGED | rc=0 >>

[root@ansible ~]#
```

Ansible Ad Hoc Commands | Ansible Modules : File, Ping, Copy, Yum, User, Service, Shell | Th... i l ↗

```
[root@ansible1 ~]# ls  
anaconda-ks.cfg  
[root@ansible1 ~]# vi [REDACTED]
```

I

◀ ▶ ⏪ ⏩ 🔍 16:50 / 30:03

Scroll for details

CC HD 🔍

[Ansible Ad Hoc Commands | Ansible Modules : File, Ping, Copy, Yum, User, Service, Shell | The Document Foundation](#)

```
[root@ansible ~]# ls
```

anaconda-ks.cfg

```
[root@ansible1 ~]# vi index.txt
```



Scroll for details





Ansible Ad Hoc Commands | Ansible Modules : File, Ping, Copy, Yum, User, Service, Shell | TheCompleteTutorials.com

```
[root@ansible1 ~]# ls  
anaconda-ks.cfg  
[root@ansible1 ~]# vi index.txt  
[root@ansible1 ~]# ls  
anaconda-ks.cfg  index.txt  
[root@ansible1 ~]#
```



Scroll for details



Ansible Ad Hoc Commands | Ansible Modules : File, Ping, Copy, Yum, User, Service, Shell |Th...



```
[root@ansible2 ~]# ls  
anaconda-ks.cfg  
[root@ansible2 ~]# ls  
anaconda-ks.cfg file  
[root@ansible2 ~]# ls  
anaconda-ks.cfg file  
[root@ansible2 ~]# ls  
anaconda-ks.cfg  
[root@ansible2 ~]# ls  
anaconda-ks.cfg test  
[root@ansible2 ~]# ls  
anaconda-ks.cfg test  
[root@ansible2 ~]# cd
```

I

Ansible Ad Hoc Commands | Ansible Modules : File, Ping, Copy, Yum, User, Service, Shell |Th...



```
[root@ansible2 ~]# ls  
anaconda-ks.cfg  
[root@ansible2 ~]# ls  
anaconda-ks.cfg file  
[root@ansible2 ~]# ls  
anaconda-ks.cfg file  
[root@ansible2 ~]# ls  
anaconda-ks.cfg  
[root@ansible2 ~]# ls  
anaconda-ks.cfg test  
[root@ansible2 ~]# ls  
anaconda-ks.cfg test  
[root@ansible2 ~]# cd test  
[root@ansible2 test]# ls  
[root@ansible2 test]#
```

Ansible Ad Hoc Commands | Ansible Modules : File, Ping, Copy, Yum, User, Service, Shell | Th...



```
[root@ansible ~]# ls
anaconda-ks.cfg
[root@ansible ~]# vi index.txt
[root@ansible ~]# ls
anaconda-ks.cfg  index.txt
[root@ansible ~]# ansible testservers -m copy -a "src=/root/index.txt dest=/root/test"
192.168.145.145 | CHANGED => [
  "ansible_facts": {
    "discovered_interpreter_python": "/usr/bin/python"
  },
  "changed": true,
  "checksum": "fc3dae7143a2971f1b9e00873bf4aac8b07603db",
  "dest": "/root/test/index.txt",
  "gid": 0,
  "group": "root",
  "md5sum": "e153075a10cddc084888999884c76003",
  "mode": "0644",
  "owner": "root",
  "secontext": "system_u:object_r:admin_home_t:s0",
  "size": 20,
  "src": "/root/.ansible/tmp/ansible-tmp-1595353230.46-7685-117462555040781/source",
  "state": "file",
  "uid": 0
}
[root@ansible ~]#
```

Ansible Ad Hoc Commands | Ansible Modules : File, Ping, Copy, Yum, User, Service, Shell |Th...



```
[root@ansible2 ~]# ls  
anaconda-ks.cfg  
[root@ansible2 ~]# ls  
anaconda-ks.cfg  file  
[root@ansible2 ~]# ls  
anaconda-ks.cfg  file  
[root@ansible2 ~]# ls  
anaconda-ks.cfg  
[root@ansible2 ~]# ls  
anaconda-ks.cfg  test  
[root@ansible2 ~]# ls  
anaconda-ks.cfg  test  
[root@ansible2 ~]# cd test  
[root@ansible2 test]# ls  
[root@ansible2 test]# pwd  
/root/test  
[root@ansible2 test]# ls  
index.txt  
[root@ansible2 test]# ca
```

Ansible Ad Hoc Commands | Ansible Modules : File, Ping, Copy, Yum, User, Service, Shell |Th...



```
[root@ansible2 ~]# ls  
anaconda-ks.cfg  
[root@ansible2 ~]# ls  
anaconda-ks.cfg  file  
[root@ansible2 ~]# ls  
anaconda-ks.cfg  file  
[root@ansible2 ~]# ls  
anaconda-ks.cfg  
[root@ansible2 ~]# ls  
anaconda-ks.cfg  test  
[root@ansible2 ~]# ls  
anaconda-ks.cfg  test  
[root@ansible2 ~]# cd test  
[root@ansible2 test]# ls  
[root@ansible2 test]# pwd  
/root/test  
[root@ansible2 test]# ls  
index.txt  
[root@ansible2 test]# cat index.txt  
Hello From Tips4you  
[root@ansible2 test]#
```

Ansible Ad Hoc Commands | Ansible Modules : File, Ping, Copy, Yum, User, Service, Shell |Th...



```
[root@ansible2 test]# wget  
-bash: wget: command not found  
[root@ansible2 test]#
```

◀ ▶ ▶| 🔍 20:01 / 30:03

Scroll for details

CC HD

Ansible Ad Hoc Commands | Ansible Modules : File, Ping, Copy, Yum, User, Service, Shell | Th... i b ↗

```
[root@ansible1 ~]# ansible testservers -m yum -a "name=wget state=present"
192.168.145.145 | CHANGED => {
    "ansible_facts": {
        "discovered_interpreter_python": "/usr/bin/python"
    },
    "changed": true,
    "changes": {
        "installed": [
            "wget"
        ],
        "msg": "",
        "rc": 0,
        "results": [
            "Loaded plugins: fastestmirror\nLoading mirror speeds from cached hostfile\n * base: centos.excellmedia.net\n * extras: centos.excellmedia.net\n * updates: centos.excellmedia.net\nResolving Dependencies\n--> Running transaction check\n--> Package wget.x86_64 0:1.14-18.el7_6.1 will be installed\n--> Finished Dependency Resolution\nDependencies Resolved\n\n=====\n           Repository          Size\non      Installing: wget      x86_64     1.14-18.el7_6.1      base      547 k\n\nTransaction Summary\n=====\nInstall 1 Package\n\nTotal download size: 547 k\nInstalled size: 2.0 M\nDownloading packages:\nRunning transaction check\nRunning transaction test\nTransaction test succeeded\nRunning transaction\n  Installing : wget-1.14-18.el7_6.1.x86_64                               1/1 \n  Verifying  : wget-1.14-18.el7_6.1.x86_64                               1/1 \n\nInstalled:\n  wget.x86_64 0:1.14-18.el7_6.1\n\nComplete!\n"
    }
}
[root@ansible1 ~]#
```

Ansible Ad Hoc Commands | Ansible Modules : File, Ping, Copy, Yum, User, Service, Shell |Th...



```
[root@ansible2 test]# wget  
-bash: wget: command not found  
[root@ansible2 test]# wget  
wget: missing URL  
Usage: wget [OPTION]... [URL]...
```

```
Try 'wget --help' for more options.  
[root@ansible2 test]#
```

Ansible Ad Hoc Commands | Ansible Modules : File, Ping, Copy, Yum, User, Service, Shell |Th...



```
[root@ansible2 test]# wget  
-bash: wget: command not found  
[root@ansible2 test]# wget  
wget: missing URL  
Usage: wget [OPTION]... [URL]...  
  
Try 'wget --help' for more options.  
[root@ansible2 test]# systemctl status httpd  
Unit httpd.service could not be found.  
[root@ansible2 test]#
```

I

Ansible Ad Hoc Commands | Ansible Modules : File, Ping, Copy, Yum, User, Service, Shell | Th...



```
[root@ansible1 ~]# ansible testservers -m yum -a "name=wget state=present"
192.168.145.145 | CHANGED => [
  "ansible_facts": {
    "discovered_interpreter_python": "/usr/bin/python"
  },
  "changed": true,
  "changes": {
    "installed": [
      "wget"
    ],
    "msg": "",
    "rc": 0,
    "results": [
      "Loaded plugins: fastestmirror\nLoading mirror speeds from cached hostfile\n * base: centos.excellmedia.net\n * extras: centos.excellmedia.net\n * updates: centos.excellmedia.net\nResolving Dependencies\n--> Running transaction check\n--> Package wget.x86_64 0:1.14-18.el7_6.1 will be installed\n--> Finished Dependency Resolution\n\nDependencies Resolved\n\n=====\n           Package          Arch      Versi
on\non          Repository      Size\n---\nInstalling: wget        x86_64      1.14-18.el7_6.1      base      547 k\n\nTransaction Summary\n=====\nInstall 1 Package\n\nTotal download size:\n 547 k\nInstalled size: 2.0 M\nDownloading packages:\nRunning transaction check\nRunning transaction test\nTransaction test succeeded\nRunning transaction\n  Installing : wget-1.14-18.el7_6.1.x86_64\n                                         1/1\nVerifying : wget-1.14-18.el7_6.1.x86_64\n                                         1/1\n\nInstalled:\n  wget.x86_64 0:1.14-18.el7_6.1\n\nComplete!\n"
  ]
}
[root@ansible1 ~]# ansible testservers -m yum -a "name=httpd state=present"
```

```
Ansible Ad Hoc Commands | Ansible Modules : File, Ping, Copy, Yum, User, Service, Shell | Th... i b ↗
fyling : wget-1.14-18.el7_6.1.x86_64          1/1 \n\nInstalled:\n  wget.x86_64 0:1.14-18.el7_6.1
          \n\nComplete!\n"
]
}
[root@ansible ~]# ansible testservers -m yum -a "name=httpd state=present"
192.168.145.145 | CHANGED => {
    "ansible_facts": {
        "discovered_interpreter_python": "/usr/bin/python"
    },
    "changed": true,
    "changes": {
        "installed": [
            "httpd"
        ]
    },
    "msg": "",
    "rc": 0,
    "results": [
        "Loaded plugins: fastestmirror\nLoading mirror speeds from cached hostfile\n * base: centos.excellmedia.net\n * extras: centos.excellmedia.net\n * updates: centos.excellmedia.net\nResolving Dependencies\n--> Running transaction check\n--> Package httpd.x86_64 0:2.4.6-93.el7.centos will be installed\n--> Processing Dependency: httpd-tools = 2.4.6-93.el7.centos for package: httpd-2.4.6-93.el7.centos.x86_64\n--> Processing Dependency: /etc/mime.types for package: httpd-2.4.6-93.el7.centos.x86_64\n--> Processing Dependency: libaprutil-1.so.0()(64bit) for package: httpd-2.4.6-93.el7.centos.x86_64\n--> Processing Dependency: libapr-1.so.0()(64bit) for package: httpd-2.4.6-93.el7.centos.x86_64\n--> Running transaction check\n--> Package apr.x86_64 0:1.4.8-5.el7 will be installed\n--> Package apr-util.x86_64 0:1.5.2-6.el7 will be installed\n--> Package httpd-tools.x86_64 0:2.4.6-93.el7.centos will be installed\n--> Package mailcap.noarch 0:2.1.41-2.el7 will be installed\n--> Finished Dependency Resolution\nDependencies Resolved\n\n-----\n Package           Arch      Version       Repository  Size\n-----\n\nInstalling:\n  httpd           x86_64   2.4.6-93.el7.centos\n                           base     2.7 M\nInstalling for dependencies:\n  apr              x86_64   1.4.8-5.el7\n                           base    103 k\n  apr-util         x86_64   0:1.5.2-6.el7\n                           base    92 k\n  mailcap          noarch  2.1.41-2.el7\n                           noarch  31 k\n\nTransaction Summary\n  3 packages installed\n  0 packages updated\n  0 packages removed\n  0 packages downgraded\n  0 packages rebuilt\n  0 packages cleaned\n  0 packages left as conflicts\n  0 packages not upgraded due to dependency problems\n\nTotal download size: 3.0 M\nTotal available disk space: 1.4 G\n\nReal time: 22:41 / 30:03\n
```

Ansible Ad Hoc Commands | Ansible Modules : File, Ping, Copy, Yum, User, Service, Shell

Suggest Docker

```
[root@ansible2 test]# wget  
-bash: wget: command not found  
[root@ansible2 test]# wget  
wget: missing URL  
Usage: wget [OPTION]... [URL]...  
  
Try 'wget --help' for more options.  
[root@ansible2 test]# systemctl status httpd  
Unit httpd.service could not be found.  
[root@ansible2 test]# systemctl status httpd  
● httpd.service - The Apache HTTP Server  
    Loaded: loaded (/usr/lib/systemd/system/httpd.service; disabled; vendor preset: disabled)  
    Active: inactive (dead)  
      Docs: man:httpd(8)  
           man:apachectl(8)  
[root@ansible2 test]#
```

Ansible Ad Hoc Commands | Ansible Modules : File, Ping, Copy, Yum, User, Service, Shell | Th... i l ↗

```
[root@ansible1 ~]# ansible testservers -m service -a "name=httpd state=started"
```

I

◀ ▶ ⏪ 🔍 23:42 / 30:03

Scroll for details

CC HD

Ansible Ad Hoc Commands | Ansible Modules : File, Ping, Copy, Yum, User, Service, Shell | Th...

```
/5 \n Verifying : mailcap-2.1.41-2.el7.noarch                                         3/5 \n Verifying : httpd-2.4.6-93.el7.centos
entos.x86_64          4/5 \n Verifying : apr-util-1.5.2-6.el7.x86_64
5/5 \n\nInstalled:\n  httpd.x86_64 0:2.4.6-93.el7.centos
                         apr.x86_64 0:1.4.8-5.el7
                         apr-util.x86_64 0:1.5.2-6.el7
                         httpd-tools.x86_64 0:2.4.6-93.el7.centos
  mailcap.noarch 0:2.1.41-2.el7      \n\nComplete!\n"
}
]

[root@ansible1 ~]# clear
[root@ansible1 ~]# ansible testservers -m service -a "name=httpd state=started"
192.168.145.145 | CHANGED => [
  "ansible_facts": {
    "discovered_interpreter_python": "/usr/bin/python"
  },
  "changed": true,
  "name": "httpd",
  "state": "started",
  "status": {
    "ActiveEnterTimestampMonotonic": "0",
    "ActiveExitTimestampMonotonic": "0",
    "ActiveState": "inactive",
    "After": "remote-fs.target network.target systemd-journald.socket nss-lookup.target -.mount tmp.mount system.slice basic.target",
    "AllowIsolate": "no",
    "AmbientCapabilities": "0",
    "AssertResult": "no",
    "AssertTimestampMonotonic": "0",
    "Before": "shutdown.target",
    "BlockIOAccounting": "no",
    "BlockIOWeight": "18446744073709551615",
    "CPUAccounting": "no",
    "CPUQuotaPerSecUsec": "infinity",
    "CPUSchedulingPolicy": "0",
    "CPUSchedulingPriority": "0"
  }
]
```

Ansible Ad Hoc Commands | Ansible Modules : File, Ping, Copy, Yum, User, Service, Shell | Th... i l ↗

```
wget: missing URL
Usage: wget [OPTION]... [URL]...

Try 'wget --help' for more options.
[root@ansible2 test]# systemctl status httpd
Unit httpd.service could not be found.
[root@ansible2 test]# systemctl status httpd
● httpd.service - The Apache HTTP Server
   Loaded: loaded (/usr/lib/systemd/system/httpd.service; disabled; vendor preset: disabled)
   Active: inactive (dead)
     Docs: man:httpd(8)
           man:apachectl(8)
[root@ansible2 test]# systemctl status httpd
● httpd.service - The Apache HTTP Server
   Loaded: loaded (/usr/lib/systemd/system/httpd.service; disabled; vendor preset: disabled)
   Active: active (running) since Tue 2020-07-21 13:52:15 EDT; 17s ago
     Docs: man:httpd(8)
           man:apachectl(8)
   Main PID: 8057 (httpd)
      Status: "Total requests: 0; Current requests/sec: 0; Current traffic: 0 B/sec"
   CGroup: /system.slice/httpd.service
           ├─8057 /usr/sbin/httpd -DFOREGROUND
           ├─8058 /usr/sbin/httpd -DFOREGROUND
           ├─8059 /usr/sbin/httpd -DFOREGROUND
           ├─8060 /usr/sbin/httpd -DFOREGROUND
           ├─8061 /usr/sbin/httpd -DFOREGROUND
           └─8062 /usr/sbin/httpd -DFOREGROUND

Jul 21 13:51:50 ansible2 systemd[1]: Starting The Apache HTTP Server...
Jul 21 13:52:05 ansible2 httpd[8057]: AH00558: httpd: Could not reliably determine the server's fully qualified d...essage
Jul 21 13:52:15 ansible2 systemd[1]: Started The Apache HTTP Server.
Hint: Some lines were ellipsized, use -l to show in full.
[root@ansible2 test]# 24:17 / 30:03
```

Scroll for details

CC HD

Ansible Ad Hoc Commands | Ansible Modules : File, Ping, Copy, Yum, User, Service, Shell | Th...



```
"TasksAccounting": "no",
"TasksCurrent": "18446744073709551615",
"TasksMax": "18446744073709551615",
"TimeoutStartUsec": "1min 30s",
"TimeoutStopUsec": "1min 30s",
"TimerSlackNsec": "50000",
"Transient": "no",
"Type": "notify",
"Umask": "0022",
"UnitFilePreset": "disabled",
"UnitFileState": "disabled",
"Wants": "system.slice",
"WatchdogTimestampMonotonic": "0",
"WatchdogUsec": "0"
}
}
[root@ansible1 ~]# ansible testservers -m service -a "name=httpd state=stopped"
192.168.145.145 | CHANGED => {
  "ansible_facts": [
    "discovered_interpreter_python": "/usr/bin/python"
  ],
  "changed": true,
  "name": "httpd",
  "state": "stopped",
  "status": {
    "ActiveEnterTimestamp": "Tue 2020-07-21 13:52:15 EDT",
    "ActiveEnterTimestampMonotonic": "1958804061",
    "ActiveExitTimestampMonotonic": "0",
    "ActiveState": "active",
    "After": "basic.target system.slice -.mount tmp.mount nss-lookup.target remote-fs.target systemd-journald.socket network.target",
    "AllowIsolate": "no",
    "AmbientCapabilities": "none"
  }
}

```

Scroll for details



Ansible Ad Hoc Commands | Ansible Modules : File, Ping, Copy, Yum, User, Service, Shell | Th...



```
● httpd.service - The Apache HTTP Server
   Loaded: loaded (/usr/lib/systemd/system/httpd.service; disabled; vendor preset: disabled)
   Active: active (running) since Tue 2020-07-21 13:52:15 EDT; 17s ago
     Docs: man:httpd(8)
           man:apachectl(8)
Main PID: 8057 (httpd)
   Status: "Total requests: 0; Current requests/sec: 0; Current traffic: 0 B/sec"
   CGrou...  
└─8057 /usr/sbin/httpd -DFOREGROUND
   ├─8058 /usr/sbin/httpd -DFOREGROUND
   ├─8059 /usr/sbin/httpd -DFOREGROUND
   ├─8060 /usr/sbin/httpd -DFOREGROUND
   ├─8061 /usr/sbin/httpd -DFOREGROUND
   └─8062 /usr/sbin/httpd -DFOREGROUND

Jul 21 13:51:50 ansible2 systemd[1]: Starting The Apache HTTP Server...
Jul 21 13:52:05 ansible2 httpd[8057]: AH00558: httpd: Could not reliably determine the server's fully qualified d...essage
Jul 21 13:52:15 ansible2 systemd[1]: Started The Apache HTTP Server.
Hint: Some lines were ellipsized, use -l to show in full.
[root@ansible2 test]# systemctl status httpd
● httpd.service - The Apache HTTP Server
   Loaded: loaded (/usr/lib/systemd/system/httpd.service; disabled; vendor preset: disabled)
   Active: inactive (dead)
     Docs: man:httpd(8)
           man:apachectl(8)

Jul 21 13:51:50 ansible2 systemd[1]: Starting The Apache HTTP Server...
Jul 21 13:52:05 ansible2 httpd[8057]: AH00558: httpd: Could not reliably determine the server's fully qualified d...essage
Jul 21 13:52:15 ansible2 systemd[1]: Started The Apache HTTP Server.
Jul 21 13:53:11 ansible2 systemd[1]: Stopping The Apache HTTP Server...
Jul 21 13:53:12 ansible2 systemd[1]: Stopped The Apache HTTP Server.
Hint: Some lines were ellipsized, use -l to show in full.
[root@ansible2 test]#
```

Scroll for details



Ansible Ad Hoc Commands | Ansible Modules : File, Ping, Copy, Yum, User, Service, Shell |Th...



```
[root@ansible ~]# more /etc/passwd
root:x:0:0:root:/root:/bin/bash
bin:x:1:1:bin:/bin:/sbin/nologin
daemon:x:2:2:daemon:/sbin:/sbin/nologin
adm:x:3:4:adm:/var/adm:/sbin/nologin
lp:x:4:7:lp:/var/spool/lpd:/sbin/nologin
sync:x:5:0:sync:/sbin:/bin/sync
shutdown:x:6:0:shutdown:/sbin:/sbin/shutdown
halt:x:7:0:halt:/sbin:/sbin/halt
mail:x:8:12:mail:/var/spool/mail:/sbin/nologin
operator:x:11:0:operator:/root:/sbin/nologin
games:x:12:100:games:/usr/games:/sbin/nologin
ftp:x:14:50:FTP User:/var/ftp:/sbin/nologin
nobody:x:99:99:Nobody:/:/sbin/nologin
systemd-network:x:192:192:systemd Network Management:/:/sbin/nologin
dbus:x:81:81:System message bus:/:/sbin/nologin
polkitd:x:999:998:User for polkitd:/:/sbin/nologin
sshd:x:74:74:Privilege-separated SSH:/var/empty/sshd:/sbin/nologin
postfix:x:89:89::/var/spool/postfix:/sbin/nologin
chrony:x:998:996::/var/lib/chrony:/sbin/nologin
[root@ansible ~]#
```

Ansible Ad Hoc Commands | Ansible Modules : File, Ping, Copy, Yum, User, Service, Shell | Th...



```
[root@ansible2 test]# more /etc/passwd
root:x:0:0:root:/root:/bin/bash
bin:x:1:1:bin:/bin:/sbin/nologin
daemon:x:2:2:daemon:/sbin:/sbin/nologin
adm:x:3:4:adm:/var/adm:/sbin/nologin
lp:x:4:7:lp:/var/spool/lpd:/sbin/nologin
sync:x:5:0:sync:/sbin:/bin/sync
shutdown:x:6:0:shutdown:/sbin:/sbin/shutdown
halt:x:7:0:halt:/sbin:/sbin/halt
mail:x:8:12:mail:/var/spool/mail:/sbin/nologin
operator:x:11:0:operator:/root:/sbin/nologin
games:x:12:100:games:/usr/games:/sbin/nologin
ftp:x:14:50:FTP User:/var/ftp:/sbin/nologin
nobody:x:99:99:Nobody:/:/sbin/nologin
systemd-network:x:192:192:systemd Network Management:/:/sbin/nologin
dbus:x:81:81:System message bus:/:/sbin/nologin
polkitd:x:999:998:User for polkitd:/:/sbin/nologin
sshd:x:74:74:Privilege-separated SSH:/var/empty/sshd:/sbin/nologin
postfix:x:89:89::/var/spool/postfix:/sbin/nologin
chrony:x:998:996::/var/lib/chrony:/sbin/nologin
apache:x:48:48:Apache:/usr/share/httpd:/sbin/nologin
[root@ansible2 test]#
```

Ansible Ad Hoc Commands | Ansible Modules : File, Ping, Copy, Yum, User, Service, Shell | Th...



```
[root@ansible1 ~]# ansible testservers -m user -a "name=test"
192.168.145.145 | CHANGED => {
    "ansible_facts": {
        "discovered_interpreter_python": "/usr/bin/python"
    },
    "changed": true,
    "comment": "",
    "create_home": true,
    "group": 1000,
    "home": "/home/test",
    "name": "test",
    "shell": "/bin/bash",
    "state": "present",
    "system": false,
    "uid": 1000
}
```

```
[root@ansible1 ~]#
```

1

Ansible Ad Hoc Commands | Ansible Modules : File, Ping, Copy, Yum, User, Service, Shell |Th...



```
[root@ansible2 test]# more /etc/passwd
root:x:0:0:root:/root:/bin/bash
bin:x:1:1:bin:/bin/nologin
daemon:x:2:2:daemon:/sbin/nologin
adm:x:3:4:adm:/var/adm:/sbin/nologin
lp:x:4:7:lp:/var/spool/lpd:/sbin/nologin
sync:x:5:0:sync:/sbin:/bin/sync
shutdown:x:6:0:shutdown:/sbin:/sbin/shutdown
halt:x:7:0:halt:/sbin:/sbin/halt
mail:x:8:12:mail:/var/spool/mail:/sbin/nologin
operator:x:11:0:operator:/root:/sbin/nologin
games:x:12:100:games:/usr/games:/sbin/nologin
ftp:x:14:50:FTP User:/var/ftp:/sbin/nologin
nobody:x:99:99:Nobody:/sbin/nologin
systemd-network:x:192:192:systemd Network Management:/sbin/nologin
dbus:x:81:81:System message bus:/sbin/nologin
polkitd:x:999:998:User for polkitd:/sbin/nologin
sshd:x:74:74:Privilege-separated SSH:/var/empty/sshd:/sbin/nologin
postfix:x:89:89::/var/spool/postfix:/sbin/nologin
chrony:x:998:996::/var/lib/chrony:/sbin/nologin
apache:x:48:48:Apache:/usr/share/httpd:/sbin/nologin
test:x:1000:1000::/home/test:/bin/bash
[root@ansible2 test]#
```

Ansible Ad Hoc Commands | Ansible Modules : File, Ping, Copy, Yum, User, Service, Shell | Th...



```
[root@ansible1 ~]# ansible testservers -m user -a "name=test"
192.168.145.145 | CHANGED => {
    "ansible_facts": {
        "discovered_interpreter_python": "/usr/bin/python"
    },
    "changed": true,
    "comment": "",
    "create_home": true,
    "group": 1000,
    "home": "/home/test",
    "name": "test",
    "shell": "/bin/bash",
    "state": "present",
    "system": false,
    "uid": 1000
}
```

```
[root@ansible1 ~]# ansible testservers -m user -a "name=test state=absent"
```

Ansible Ad Hoc Commands | Ansible Modules : File, Ping, Copy, Yum, User, Service, Shell | Th...



```
[root@ansible1 ~]# ansible testservers -m user -a "name=test"
192.168.145.145 | CHANGED => {
    "ansible_facts": {
        "discovered_interpreter_python": "/usr/bin/python"
    },
    "changed": true,
    "comment": "",
    "create_home": true,
    "group": 1000,
    "home": "/home/test",
    "name": "test",
    "shell": "/bin/bash",
    "state": "present",
    "system": false,
    "uid": 1000
}
[root@ansible1 ~]# ansible testservers -m user -a "name=test state=absent"
192.168.145.145 | CHANGED => {
    "ansible_facts": {
        "discovered_interpreter_python": "/usr/bin/python"
    },
    "changed": true,
    "force": false,
    "name": "test",
    "remove": false,
    "state": "absent"
}
[root@ansible1 ~]#
```

Ansible Ad Hoc Commands | Ansible Modules : File, Ping, Copy, Yum, User, Service, Shell |Th...



```
games:x:12:100:games:/usr/games:/sbin/nologin
ftp:x:14:50:FTP User:/var/ftp:/sbin/nologin
nobody:x:99:99:Nobody:/sbin/nologin
systemd-network:x:192:192:systemd Network Management:/sbin/nologin
dbus:x:81:81:System message bus:/sbin/nologin
polkitd:x:999:998:User for polkitd:/sbin/nologin
sshd:x:74:74:Privilege-separated SSH:/var/empty/sshd:/sbin/nologin
postfix:x:89:89::/var/spool/postfix:/sbin/nologin
chrony:x:998:996::/var/lib/chrony:/sbin/nologin
apache:x:48:48:Apache:/usr/share/httpd:/sbin/nologin
test:x:1000:1000::/home/test:/bin/bash
[root@ansible2 test]# more /etc/passwd
root:x:0:0:root:/root:/bin/bash
bin:x:1:1:bin:/bin:/sbin/nologin
daemon:x:2:2:daemon:/sbin:/sbin/nologin
adm:x:3:4:adm:/var/adm:/sbin/nologin
lp:x:4:7:lp:/var/spool/lpd:/sbin/nologin
sync:x:5:0:sync:/sbin:/sync
shutdown:x:6:0:shutdown:/sbin:/sbin/shutdown
halt:x:7:0:halt:/sbin:/sbin/halt
mail:x:8:12:mail:/var/spool/mail:/sbin/nologin
operator:x:11:0:operator:/root:/sbin/nologin
games:x:12:100:games:/usr/games:/sbin/nologin
ftp:x:14:50:FTP User:/var/ftp:/sbin/nologin
nobody:x:99:99:Nobody:/sbin/nologin
systemd-network:x:192:192:systemd Network Management:/sbin/nologin
dbus:x:81:81:System message bus:/sbin/nologin
polkitd:x:999:998:User for polkitd:/sbin/nologin
sshd:x:74:74:Privilege-separated SSH:/var/empty/sshd:/sbin/nologin
postfix:x:89:89::/var/spool/postfix:/sbin/nologin
chrony:x:998:996::/var/lib/chrony:/sbin/nologin
apache:x:48:48:Apache:/usr/share/httpd:/sbin/nologin
[root@ansible2 test]#
```

[root@ansible2 test]#

Scroll for details



Ansible Ad Hoc Commands | Ansible Modules : File, Ping, Copy, Yum, User, Service, Shell | Th... i b ↗

```
[root@ansible1 ~]# ansible testservers -a "/sbin/reboot"
```

◀ ▶ ⏪ ⏩ 🔍 28:47 / 30:03 Scroll for details CC HD 🔍

Ansible Ad Hoc Commands | Ansible Modules : File, Ping, Copy, Yum, User, Service, Shell | Th... i t ↗

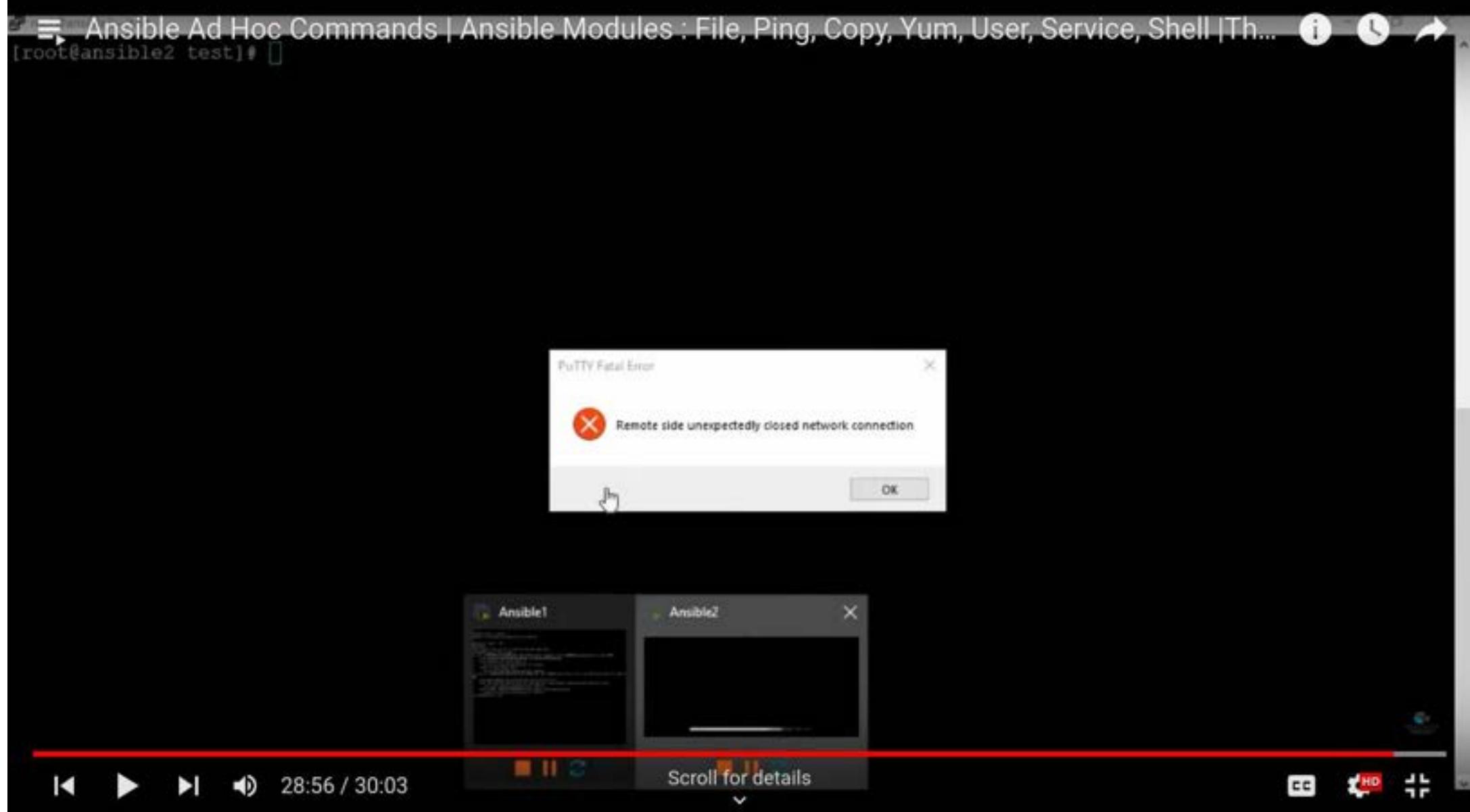
```
[root@ansible1 ~]# ansible testservers -a "/sbin/reboot"
192.168.145.145 | FAILED | rc=1 >>
Failed to connect to the host via ssh: ssh: connect to host 192.168.145.145 port 22: Connection refused
[root@ansible1 ~]#
```

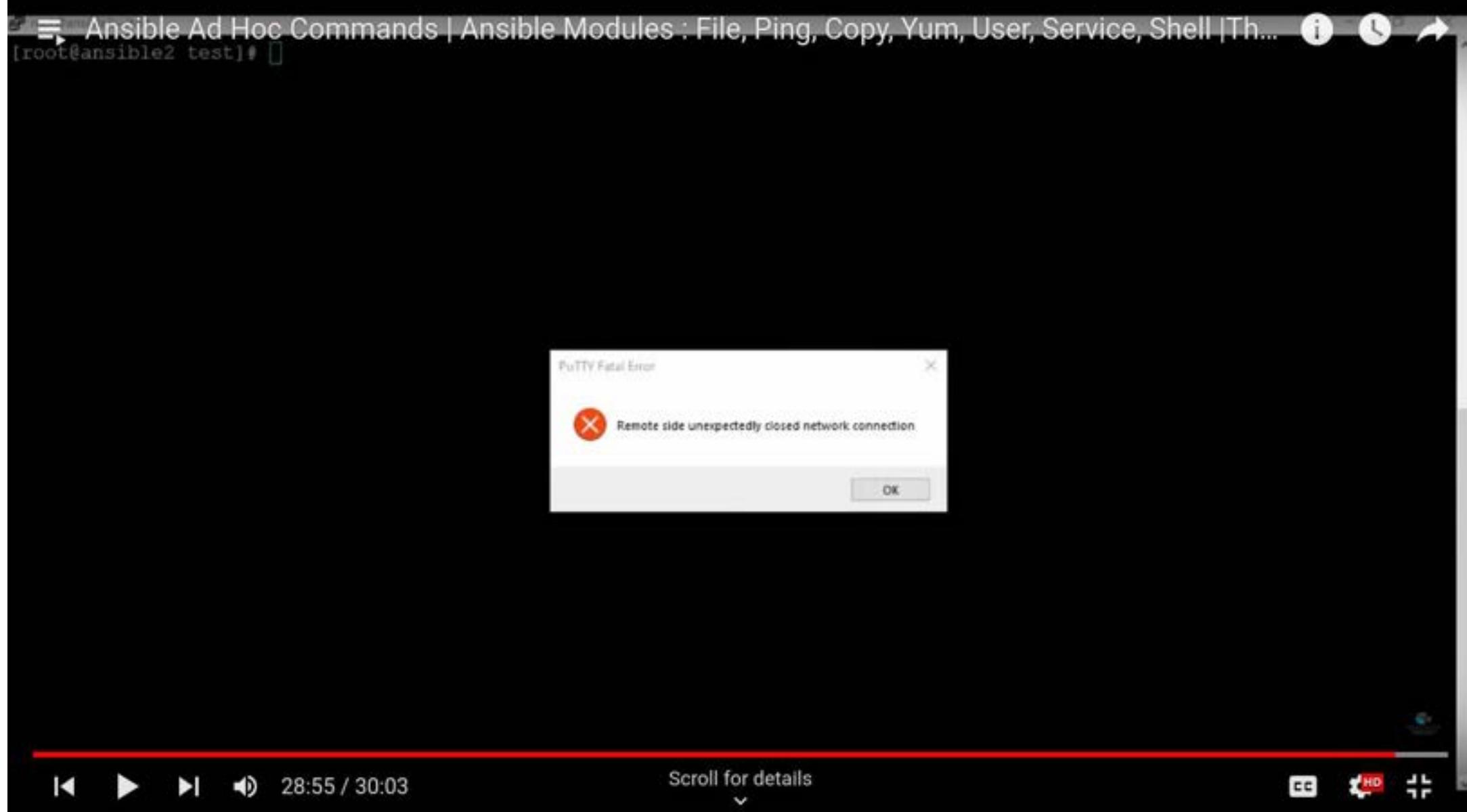
1

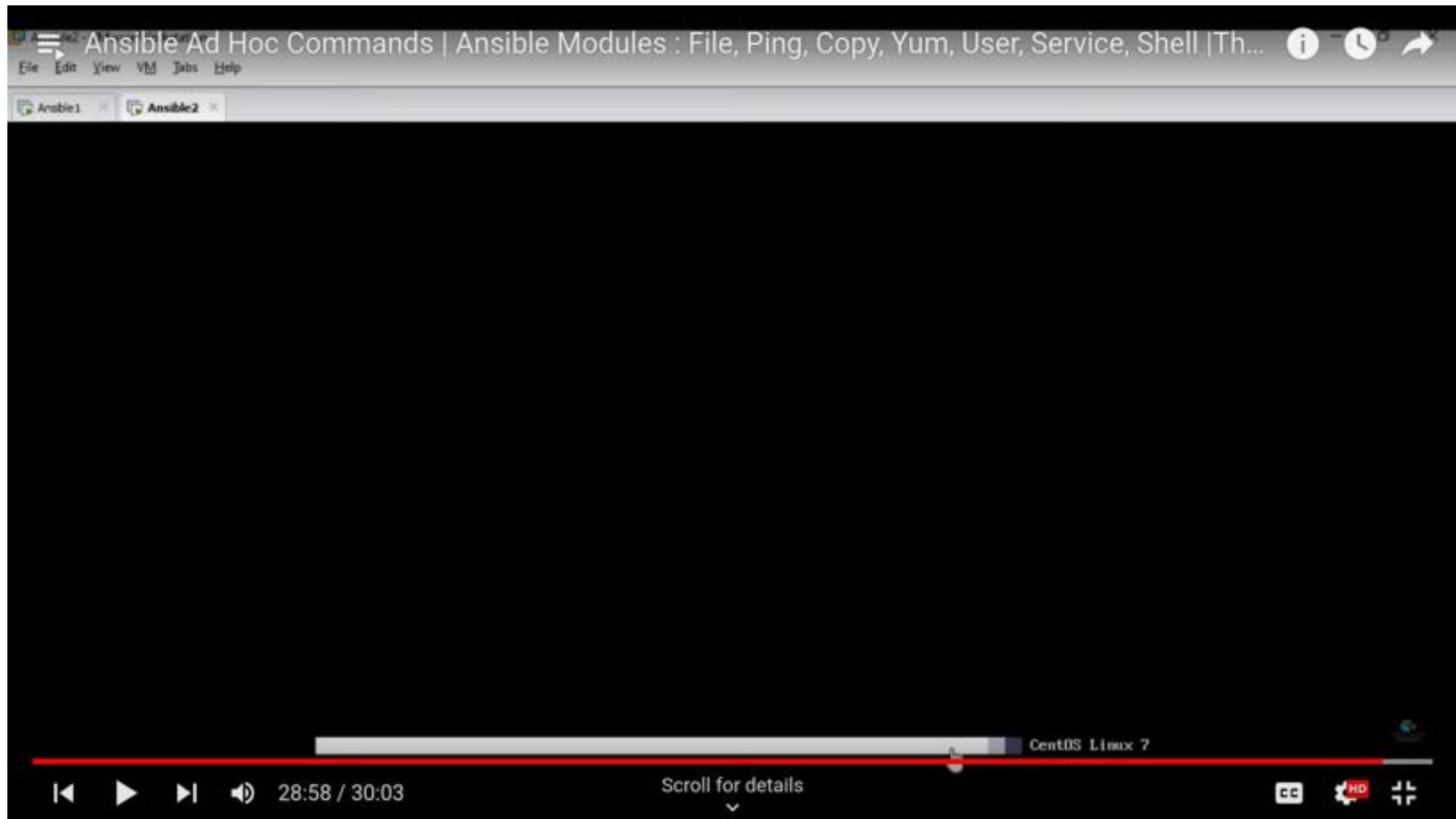
◀ ▶ ⏪ ⏩ 🔍 28:52 / 30:03

Scroll for details

CC HD







Ansible Ad Hoc Commands | Ansible Modules : File, Ping, Copy, Yum, User, Service, Shell | Th...



File Edit View VM Jobs Help

Ansible1 Ansible2

CentOS Linux 7 (Core)
Kernel 3.10.0-957.el7.x86_64 on an x86_64

ansible2 login: _



◀ ▶ ⏪ ⏩ 🔍 29:09 / 30:03

Scroll for details

CC HD

Hello Friends, Welcome back to my channel

- Today we are going to see another tutorial on Ansible.
- In my previous videos on ansible , we have setup ansible on centos and ubuntu machines, we have also seen some of the ad-hoc commands that can be used for one time activities.
- In this video, we will see how we can create ansible playbook. We will start with simple playbooks and as we go on we will create complex playbooks that can be used to run the tasks repeatedly by just running the ansible playbooks.

What is Ansible Playbook?

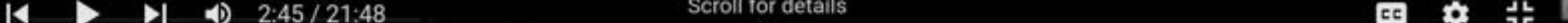
Ansible Playbook Beginners Tutorial // Create Your First Playbook To Install Apache Server -...



```
[root@ansible1 ~]# ansible
usage: ansible [-h] [--version] [-v] [-b] [--become-method BECOME_METHOD]
                [--become-user BECOME_USER] [-K] [-i INVENTORY] [--list-hosts]
                [-l SUBSET] [-P POLL_INTERVAL] [-S SECONDS] [-o] [-t TREE] [-k]
                [--private-key PRIVATE_KEY_FILE] [-u REMOTE_USER]
                [-c CONNECTION] [-T TIMEOUT]
                [--ssh-common-args SSH_COMMON_ARGS]
                [--sftp-extra-args SFTP_EXTRA_ARGS]
                [--scp-extra-args SCP_EXTRA_ARGS]
                [--ssh-extra-args SSH_EXTRA_ARGS] [-C] [--syntax-check] [-D]
                [-e EXTRA_VARS] [--vault-id VAULT_IDS]
                [--ask-vault-pass | --vault-password-file VAULT_PASSWORD_FILES]
                [-f FORKS] [-M MODULE_PATH] [--playbook-dir BASEDIR]
                [-a MODULE_ARGS] [-m MODULE_NAME]
                pattern
ansible: error: too few arguments
[root@ansible1 ~]#
```

Ansible Playbook Beginners Tutorial // Create Your First Playbook To Install Apache Server -... i b ↗

```
[root@ansible ~]# cd /etc/ansible/
[root@ansible ansible]# ls
ansible.cfg  hosts  roles
[root@ansible ansible]#
```



2:45 / 21:48

Scroll for details

CC gear grid

Ansible Playbook Beginners Tutorial // Create Your First Playbook To Install Apache Server -... i b ↗

```
[root@ansible ~]# cd /etc/ansible/  
[root@ansible ansible]# ls  
ansible.cfg hosts roles  
[root@ansible ansible]# cat hosts
```

◀ ▶ ⏪ ⏩ 🔍 2:48 / 21:48

Scroll for details

CC ⚙ ⌂

Ansible Playbook Beginners Tutorial // Create Your First Playbook To Install Apache Server -...



```
[testservers]
192.168.145.145

## [webservers]
## alpha.example.org
## beta.example.org
## 192.168.1.100
## 192.168.1.110

# If you have multiple hosts following a pattern you can specify
# them like this:

## www[001:006].example.com

# Ex 3: A collection of database servers in the 'dbservers' group

## [dbservers]
##
## db01.intranet.mydomain.net
## db02.intranet.mydomain.net
## 10.25.1.56
## 10.25.1.57

# Here's another example of host ranges, this time there are no
# leading 0s:

## db-[99:101]-node.example.com
```

[root@ansibile ~]#

Scroll for details



Ansible Playbook Beginners Tutorial // Create Your First Playbook To Install Apache Server -...

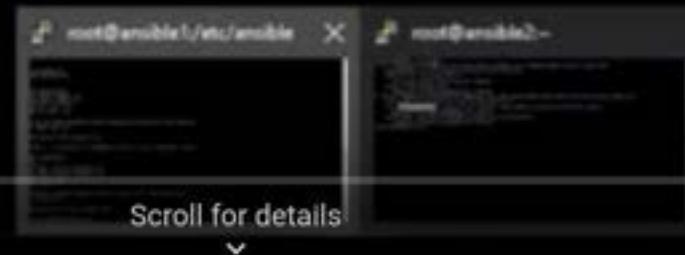


```
[root@ansible ~]# ip addr
1: lo: <LOOPBACK,UP,LOWER_UP> mtu 65536 qdisc noqueue state UNKNOWN group default qlen 1000
    link/loopback 00:00:00:00:00:00 brd 00:00:00:00:00:00
    inet 127.0.0.1/8 scope host lo
        valid_lft forever preferred_lft forever
    inet6 ::1/128 scope host
        valid_lft forever preferred_lft forever
2: ens33: <BROADCAST,MULTICAST,UP,LOWER_UP> mtu 1500 qdisc pfifo_fast state UP group default qlen 1000
    link/ether 00:0c:29:23:a2:10 brd ff:ff:ff:ff:ff:ff
    inet 192.168.145.145/24 brd 192.168.145.255 scope global noprefixroute dynamic ens33
        valid_lft 1146sec preferred_lft 1146sec
    inet6 fe80::5c6e:f999:e04:80a5/64 scope link noprefixroute
        valid_lft forever preferred_lft forever
[root@ansible2 ~]#
```

Ansible Playbook Beginners Tutorial // Create Your First Playbook To Install Apache Server -...



```
[root@ansible ~]# ip addr
1: lo: <LOOPBACK,UP,LOWER_UP> mtu 65536 qdisc noqueue state UNKNOWN group default qlen 1000
    link/loopback 00:00:00:00:00:00 brd 00:00:00:00:00:00
    inet 127.0.0.1/8 scope host lo
        valid_lft forever preferred_lft forever
    inet6 ::1/128 scope host
        valid_lft forever preferred_lft forever
2: ens33: <BROADCAST,MULTICAST,UP,LOWER_UP> mtu 1500 qdisc pfifo_fast state UP group default qlen 1000
    link/ether 00:0c:29:23:a2:10 brd ff:ff:ff:ff:ff:ff
    inet 192.168.145.145/24 brd 192.168.145.255 scope global noprefixroute dynamic ens33
        valid_lft 1146sec preferred_lft 1146sec
    inet6 fe80::5c6e:f999:e04:80a5/64 scope link noprefixroute
        valid_lft forever preferred_lft forever
[root@ansible2 ~]#
```



◀ ▶ ⏪ ⏩ 3:02 / 21:48

Scroll for details



Ansible Playbook Beginners Tutorial // Create Your First Playbook To Install Apache Server -...



```
[root@ansible ~]# wget  
-bash: wget: command not found
```

```
[root@ansible ~]#
```



4:20 / 21:48

Scroll for details





```
[root@ansible ~]# ansible testservers -m yum -a "name=wget state=present"
```

Ansible Playbook Beginners Tutorial // Create Your First Playbook To Install Apache Server -...



```
[root@ansible ~]# ansible testserver -m yum -a "name=wget state=present"
192.168.145.145 | CHANGED => (
    "ansible_facts": {
        "discovered_interpreter_python": "/usr/bin/python"
    },
    "changed": true,
    "changes": {
        "installed": [
            "wget"
        ]
    },
    "msg": "",
    "rc": 0,
    "results": [
        "Loaded plugins: fastestmirror\nLoading mirror speeds from cached hostfile\n * base: centos.excellimmedia.net\n * extras: mirror.dhakacom.com\n * updates: mirrors.nhanhoa.com\nResolving Dependencies\n--> Running transaction check\n--> Package wget.x86_64 0:1.14-18.el7_6.1 will be installed\n--> Finished Dependency Resolution\n\nDependencies Resolved\n\n=====\n           \n          Package           Arch      Version\n          Repository      Size\n\nInstalling:\n  wget           x86_64     1.14-18.el7_6.1      base      547 k\n\nTransaction Summary\n=====\n\nInstall  1 Package\n\nTotal download size: 547 k\n\nInstalled size: 2.0 M\n\nDownloading packages:\nRunning transaction check\nRunning transaction test\nTransaction test succeeded\nRunning transaction\n  Installing : wget-1.14-18.el7_6.1.x86_64                                1/1\n  Verifying  : wget-1.14-18.el7_6.1.x86_64                                1/1\n\nInstalled:\n  wget.x86_64 0:1.14-18.el7_6.1\n\nComplete!\n"
    ]
)
[root@ansible ~]#
```

Ansible Playbook Beginners Tutorial // Create Your First Playbook To Install Apache Server -...



```
[root@ansible ~]# wget  
wget: missing URL  
Usage: wget [OPTION]... [URL]...  
  
Try 'wget --help' for more options.  
[root@ansible2 ~]#
```

Ansible Playbook Beginners Tutorial // Create Your First Playbook To Install Apache Server -...



```
[root@ansible ~]# ansible testservers -m yum -a "name=wget state=present"
192.168.145.145 | CHANGED => {
    "ansible_facts": {
        "discovered_interpreter_python": "/usr/bin/python"
    },
    "changed": true,
    "changes": {
        "installed": [
            "wget"
        ]
    },
    "msg": "",
    "rc": 0,
    "results": [
        "Loaded plugins: fastestmirror\nLoading mirror speeds from cached hostfile\n * base: centos.excellimmedia.net\n * extras: mirror.dhakacom.com\n * updates: mirrors.nhanhoa.com\nResolving Dependencies\n--> Running transaction check\n--> Package wget.x86_64 0:1.14-18.el7_6.1 will be installed\n--> Finished Dependency Resolution\n\nDependencies Resolved\n\n=====\n           \n          Package           Arch      Version           \n          Repository      Size\n\n-----\nInstalling:\n  wget           x86_64     1.14-18.el7_6.1      base      547 k\n\nTransaction Summary\n=====\n\nInstall  1 Package\n\nTotal download size: 547 k\n\nInstalled size: 2.0 M\n\nDownloading packages:\nRunning transaction check\nRunning transaction test\nTransaction test succeeded\nRunning transaction\n  Installing : wget-1.14-18.el7_6.1.x86_64                                1/1\n  Verifying  : wget-1.14-18.el7_6.1.x86_64                                1/1\n\nInstalled:\n  wget.x86_64 0:1.14-18.el7_6.1\n\nComplete!\n"
    ]
}
[root@ansible ~]# ansible testservers -m yum -a "name=wget state=present"
```

Ansible Playbook Beginners Tutorial // Create Your First Playbook To Install Apache Server -...

```
[root@ansible1 ~]# ansible testservers -m yum -a "name=wget state=present"
192.168.145.145 | CHANGED => {
    "ansible_facts": {
        "discovered_interpreter_python": "/usr/bin/python"
    },
    "changed": true,
    "changes": [
        "installed": [
            "wget"
        ]
    ],
    "msg": "",
    "rc": 0,
    "results": [
        "Loaded plugins: fastestmirror\nLoading mirror speeds from cached hostfile\n * base: centos.excellimmedia.net\n * extras: mirror.dhakacom.com\n * updates: mirrors.nhanhoa.com\nResolving Dependencies\n--> Running transaction check\n--> Package wget.x86_64 0:1.14-18.el7_6.1 will be installed\n--> Finished Dependency Resolution\n\nDependencies Resolved\n\n=====\n                                               \n Package          Arch      Version\n                                               \nInsa\n       Repository      Size\n-----\ntalling:\n wget           x86_64     1.14-18.el7_6.1      base      547 k\nTransaction Summary\n-----\n\nInstall  1 Package\n\nTotal download size: 547 k\n\nInstalled size: 2.0 M\nDownloading packages:\nRunning transaction check\nRunning transaction test\nTransaction test succeeded\nRunning transaction\n  Installing : wget-1.14-18.el7_6.1.x86_64                                1/1\n  Verifying   : wget-1.14-18.el7_6.1.x86_64                                1/1\n\nInstalled:\n  wget.x86_64 0:1.14-18.el7_6.1\n\nComplete!\n"
    ]
}
[root@ansible1 ~]# ansible testservers -m yum -a "name=wget state=absent"
```

Ansible Playbook Beginners Tutorial // Create Your First Playbook To Install Apache Server -... i b ↗

```
[root@ansible ~]# wget
wget: missing URL
Usage: wget [OPTION]... [URL]...
Try 'wget --help' for more options.
[root@ansible2 ~]# wget
-bash: /usr/bin/wget: No such file or directory
[root@ansible2 ~]#
```

6:09 / 21:48

Scroll for details

CC gear grid

Ansible Playbook Beginners Tutorial // Create Your First Playbook To Install Apache Server – ...

```
: wget-1.14-18.el7_6.1.x86_64          1/1 \n\nInstalled:\n  wget.x86_64 0:1.14-18.el7_6.1\n\nComplete!\n\n}\n\n[root@ansible1 ~]# ansible testservers -m yum -a "name=wget state=absent"\n192.168.145.145 | CHANGED => {\n    "ansible_facts": {\n        "discovered_interpreter_python": "/usr/bin/python"\n    },\n    "changed": true,\n    "changes": {\n        "removed": [\n            "wget"\n        ]\n    },\n    "msg": "",\n    "rc": 0,\n    "results": {\n        "Loaded plugins: fastestmirror\nResolving Dependencies\n--> Running transaction check\n---> Package wget.x86_64 0:1.14-18.el7_6.1 will be erased\n--> Finished Dependency Resolution\n\nDependencies Resolved\n\n=====\n\n           \n      Package          Arch          Version          Repository\nSize\n\nx86_64      1.14-18.el7_6.1      @base          2.0 M\n\nTransaction Summary\n\n\nRemove 1 Package\n\n\nInstalled size: 2.0 M\n\nDownloading packages:\n\nRunning transaction check\n\nRunning transaction test\n\nTransaction test succeeded\n\nRunning transaction\n\n  Erasing   : wget-1.14-18.el7_6.1.x86_64\n\n  1/1 \n\n  Verifying  : wget-1.14-18.el7_6.1.x86_64\n\n  1/1 \n\nRemoved:\n  wget.x86_64 0:1.14-18.el7_6.1\n\n\nComplete!\n\n}
```

Ansible Playbook Beginners Tutorial // Create Your First Playbook To Install Apache Server – ...



FOLDERS

└ demapp

```
1
2   ---+
3   2   + name: Server Details
4   3   hosts: testservers
5   4   remote_user: root
6
7   tasks:
8     - name: Installing wget
9       yum: pkg=wget state=present
```

◀ ▶ ⏪ ⏩ 🔍 10:42 / 21:48

Scroll for details

CC ⚙️ ⌂



Ansible Playbook Beginners Tutorial // Create Your First Playbook To Install Apache Server -...



```
---
```

```
- name: Server Details
hosts: testservers
remote_user: root

tasks:
- name: Installing wget
  yum: pkg=wget state=present
```

INSERT ▶ | ◀ 🔍 11:05 / 21:48

Scroll for details



Ansible Playbook Beginners Tutorial // Create Your First Playbook To Install Apache Server -...



```
[root@ansible1 ansibleplaybook]# vi wget.yml  
[root@ansible1 ansibleplaybook]# ls  
wget.yml  
[root@ansible1 ansibleplaybook]#
```

Ansible Playbook Beginners Tutorial // Create Your First Playbook To Install Apache Server -...



```
[root@ansible1 ansibleplaybook]# vi wget.yml  
[root@ansible1 ansibleplaybook]# ls  
wget.yml  
[root@ansible1 ansibleplaybook]# ansible-playbook wget.yml
```

Ansible Playbook Beginners Tutorial // Create Your First Playbook To Install Apache Server -...



```
[root@ansible1 ansibleplaybook]# vi wget.yml
[root@ansible1 ansibleplaybook]# ls
wget.yml
[root@ansible1 ansibleplaybook]# ansible-playbook wget.yml

PLAY [Server Details] *****

TASK [Gathering Facts] *****
ok: [192.168.145.145]

TASK [Installing wget] *****
changed: [192.168.145.145]

PLAY RECAP *****
192.168.145.145      : ok=2    changed=1    unreachable=0    failed=0    skipped=0    rescued=0    ignored=0

[root@ansible1 ansibleplaybook]#
```

Ansible Playbook Beginners Tutorial // Create Your First Playbook To Install Apache Server -...



```
[root@ansible ~]# wget  
wget: missing URL  
Usage: wget [OPTION]... [URL]...  
  
Try 'wget --help' for more options.  
[root@ansible2 ~]#
```

◀ ▶ ▶| 🔍 12:41 / 21:48

Scroll for details

CC ⚙️ 📺

Ansible Playbook Beginners Tutorial // Create Your First Playbook To Install Apache Server -...



```
---
```

```
- name: Server Details
  hosts: testservers
  remote_user: root

  tasks:
    - name: Installing wget
      yum: pkg=wget state=present
```

Ansible Playbook Beginners Tutorial // Create Your First Playbook To Install Apache Server

Suggest i Splunk

```
---
```

```
- name: Server Details
hosts: testservers
remote_user: root

tasks:
- name: Installing wget
  yum: pkg=wget state=absent
```

Ansible Playbook Beginners Tutorial // Create Your First Playbook To Install Apache Server -...



```
[root@ansible1 ansibleplaybook]# vi wget.yml
[root@ansible1 ansibleplaybook]# ls
wget.yml
[root@ansible1 ansibleplaybook]# ansible-playbook wget.yml

PLAY [Server Details] *****

TASK [Gathering Facts] *****
ok: [192.168.145.145]

TASK [Installing wget] *****
changed: [192.168.145.145]

PLAY RECAP *****
192.168.145.145      : ok=2    changed=1    unreachable=0    failed=0    skipped=0    rescued=0    ignored=0

[root@ansible1 ansibleplaybook]# vi wget.yml
[root@ansible1 ansibleplaybook]# ansible-playbook wget.yml

PLAY [Server Details] *****

TASK [Gathering Facts] *****
ok: [192.168.145.145]

TASK [Installing wget] *****
changed: [192.168.145.145]

PLAY RECAP *****
192.168.145.145      : ok=2    changed=1    unreachable=0    failed=0    skipped=0    rescued=0    ignored=0
```

Ansible Playbook Beginners Tutorial // Create Your First Playbook To Install Apache Server – Part 1



```
[root@ansible ~]# wget  
wget: missing URL  
Usage: wget [OPTION]... [URL]...  
  
Try 'wget --help' for more options.  
[root@ansible ~]# wget  
-bash: /usr/bin/wget: No such file or directory  
[root@ansible ~]#
```

Ansible Playbook Beginners Tutorial // Create Your First Playbook To Install Apache Server – Part 1

FOLDERS

demapp

```
1  #!/usr/bin/python
2
3  - name: Server Details
4    hosts: testservers
5    remote_user: root
6
7  tasks:
8    - name: Installing httpd
9      yum: pkg=httpd state=latest
10
```



Ansible Playbook Beginners Tutorial // Create Your First Playbook To Install Apache Server – Part 1

```
---
```

```
- name: Server Details
hosts: testservers
remote_user: root

tasks:
- name: Installing httpd
  yum: pkg=httpd state=latest
```

Ansible Playbook Beginners Tutorial // Create Your First Playbook To Install Apache Server – Part 1

```
[root@ansible ~]# wget
wget: missing URL
Usage: wget [OPTION]... [URL]...

Try 'wget --help' for more options.
[root@ansible ~]# wget
-bash: /usr/bin/wget: No such file or directory
[root@ansible ~]# systemctl
```

Ansible Playbook Beginners Tutorial // Create Your First Playbook To Install Apache Server – Part 1



```
[root@ansible ~]# wget  
wget: missing URL  
Usage: wget [OPTION]... [URL]...  
  
Try 'wget --help' for more options.  
[root@ansible2 ~]# wget  
-bash: /usr/bin/wget: No such file or directory  
[root@ansible2 ~]# systemctl status httpd  
Unit httpd.service could not be found.  
[root@ansible2 ~]#
```

Ansible Playbook Beginners Tutorial // Create Your First Playbook To Install Apache Server – Part 1

```
[root@ansible1 ansibleplaybook]# vi httpd.yml
[root@ansible1 ansibleplaybook]# ansible-playbook httpd.yml

PLAY [Server Details] *****

TASK [Gathering Facts] *****
ok: [192.168.145.145]

TASK [Installing httpd] *****
changed: [192.168.145.145]

PLAY RECAP *****
192.168.145.145 : ok=2    changed=1    unreachable=0    failed=0    skipped=0    rescued=0    ignored=0

[root@ansible1 ansibleplaybook]#
```

Ansible Playbook Beginners Tutorial // Create Your First Playbook To Install Apache Server – Part 1

```
[root@ansible ~]# wget
wget: missing URL
Usage: wget [OPTION]... [URL]...

Try 'wget --help' for more options.
[root@ansible2 ~]# wget
-bash: /usr/bin/wget: No such file or directory
[root@ansible2 ~]# systemctl status httpd
Unit httpd.service could not be found.
[root@ansible2 ~]# systemctl status httpd
● httpd.service - The Apache HTTP Server
   Loaded: loaded (/usr/lib/systemd/system/httpd.service; disabled; vendor preset: disabled)
   Active: inactive (dead)
     Docs: man:httpd(8)
           man:apachectl(8)
[root@ansible2 ~]#
```

Ansible Playbook Beginners Tutorial // Create Your First Playbook To Install Apache Server – Part 1

```
---
```

```
- name: Server Details
hosts: testservers
remote_user: root

tasks:
- name: Installing httpd
  yum: pkg=httpd state=absent
```

Ansible Playbook Beginners Tutorial // Create Your First Playbook To Install Apache Server – Part 1

```
[root@ansible1 ansibleplaybook]# vi httpd.yml
[root@ansible1 ansibleplaybook]# ansible-playbook httpd.yml
```

16:15 / 21:48

Scroll for details

Ansible Playbook Beginners Tutorial // Create Your First Playbook To Install Apache Server – Part 1

```
[root@ansible1 ansibleplaybook]# vi httpd.yml  
[root@ansible1 ansibleplaybook]# ansible-playbook httpd.yml
```

```
PLAY [Server Details] *****
```

```
TASK [Gathering Facts] *****
```

Ansible Playbook Beginners Tutorial // Create Your First Playbook To Install Apache Server – Part 1

```
[root@ansible1 ansibleplaybook]# vi httpd.yml
[root@ansible1 ansibleplaybook]# ansible-playbook httpd.yml

PLAY [Server Details] *****

TASK [Gathering Facts] *****
ok: [192.168.145.145]

TASK [Installing httpd] *****
changed: [192.168.145.145]

PLAY RECAP *****
192.168.145.145      : ok=2    changed=1    unreachable=0    failed=0    skipped=0    rescued=0    ignored=0

[root@ansible1 ansibleplaybook]#
```

Ansible Playbook Beginners Tutorial // Create Your First Playbook To Install Apache Server – Part 1



```
[root@ansible ~]# wget
wget: missing URL
Usage: wget [OPTION]... [URL]...

Try 'wget --help' for more options.
[root@ansible ~]# wget
-bash: /usr/bin/wget: No such file or directory
[root@ansible ~]# systemctl status httpd
Unit httpd.service could not be found.
[root@ansible ~]# systemctl status httpd
● httpd.service - The Apache HTTP Server
   Loaded: loaded (/usr/lib/systemd/system/httpd.service; disabled; vendor preset: disabled)
   Active: inactive (dead)
     Docs: man:httpd(8)
           man:apachectl(8)
[root@ansible ~]# systemctl status httpd
Unit httpd.service could not be found.
[root@ansible ~]#
```

Ansible Playbook Beginners Tutorial // Create Your First Playbook To Install Apache Server – Part 1

FOLDERS

demoapp

```
1  ---
2  - name: Server Details
3  hosts: testservers
4  remote_user: root
5
6  tasks:
7  - name: Installing httpd
8  yum: pkg={{item}} state=present
9
10
11
```

Ansible Playbook Beginners Tutorial // Create Your First Playbook To Install Apache Server – Part 1

FOLDERS

demoapp

```
1  ---
2  -> main: Server Details
3    hosts: testservers
4    remote_user: root
5
6    tasks:
7      - name: Installing httpd
8        yum: pkg={{item}} state=present
9        with_items:
10          - wget
11          - httpd
```

Ansible Playbook Beginners Tutorial // Create Your First Playbook To Install Apache Server – Part 1

```
[root@ansible1 ansibleplaybook]# vi httpd.yml
[root@ansible1 ansibleplaybook]# ansible-playbook httpd.yml

PLAY [Server Details] *****

TASK [Gathering Facts] *****
ok: [192.168.145.145]

TASK [Installing httpd] *****
changed: [192.168.145.145]

PLAY RECAP *****
192.168.145.145 : ok=2    changed=1    unreachable=0    failed=0    skipped=0    rescued=0    ignored=0

[root@ansible1 ansibleplaybook]# vi httpdwget.yml
```

Ansible Playbook Beginners Tutorial // Create Your First Playbook To Install Apache Server – Part 1

```
---
```

```
- name: Server Details
hosts: testservers
remote_user: root

tasks:
- name: Installing httpd
  yum: pkg={{item}} state=present
  with_items:
  - wget
  - httpd
```

Ansible Playbook Beginners Tutorial // Create Your First Playbook To Install Apache Server – Part 1

```
[root@ansible1 ansibleplaybook]# ansible-playbook httpd.yml

PLAY [Server Details] *****

TASK [Gathering Facts] *****
ok: [192.168.145.145]

TASK [Installing httpd] *****
changed: [192.168.145.145]

PLAY RECAP *****
192.168.145.145 : ok=2    changed=1    unreachable=0    failed=0    skipped=0    rescued=0    ignored=0

[root@ansible1 ansibleplaybook]# vi httpdwget.yml
[root@ansible1 ansibleplaybook]# ansible-playbook httpdwget.yml

PLAY [Server Details] *****

TASK [Gathering Facts] *****
ok: [192.168.145.145]

TASK [Installing httpd] *****
[DEPRECATION WARNING]: Invoking "yum" only once while using a loop via squash_actions is deprecated. Instead of using a loop to supply multiple items and specifying `pkg: "({{item}})"`, please use `pkg: ['wget', 'httpd']` and remove the loop. This feature will be removed in version 2.11. Deprecation warnings can be disabled by setting deprecation_warnings=False in ansible.cfg.
changed: [192.168.145.145] => (item=[u'wget', u'httpd'])

PLAY RECAP *****
192.168.145.145 : ok=2    changed=1    unreachable=0    failed=0    skipped=0    rescued=0    ignored=0
```

[root@ansible1 ansibleplaybook]# 19:08 / 21:48

Scroll for details



Ansible Playbook Beginners Tutorial // Create Your First Playbook To Install Apache Server – Part 1

```
[root@ansible2 ~]# wget
wget: missing URL
Usage: wget [OPTION]... [URL]...

Try 'wget --help' for more options.
[root@ansible2 ~]# systemctl status httpd
● httpd.service - The Apache HTTP Server
   Loaded: loaded (/usr/lib/systemd/system/httpd.service; disabled; vendor preset: disabled)
   Active: inactive (dead)
     Docs: man:httpd(8)
           man:apachectl(8)
[root@ansible2 ~]#
```

Ansible Playbook Beginners Tutorial // Create Your First Playbook To Install Apache Server – Part 1

FOLDERS

demoapp

```
1  ---
2  - name: Server Details
3  hosts: testservers
4  remote_user: root
5
6  tasks:
7  - name: Installing httpd & Wget
8  yum: pkg={{item}} state=present
9  with_items:
10 - wget
11 - httpd
```

Ansible Playbook Beginners Tutorial // Create Your First Playbook To Install Apache Server – Part 1

```
1 ---  
2   - name: Server Details  
3     hosts: testservers  
4     become_user: root  
5  
6     tasks:  
7       - name: Installing Wget & httpd  
8         yum: pkg={{item}} state=present  
9         with_items:  
10           - wget  
11           - httpd
```

Ansible Playbook Beginners Tutorial // Create Your First Playbook To Install Apache Server – Part 1

FOLDERS

demapp

```
1  #!/usr/bin/python
2
3  #> vars:
4  #>   name: Server Details
5  #>   hosts: testservers
6  #>   remote_user: root
7
8  tasks:
9  - name: Installing Wget & Httpd
10    yum: pkg={{item}} state=present
11    with_items:
12      - wget
13      - httpd
14
```

Hello Friends, Welcome back to my channel

- We are here with another tutorial on Ansible. This will be continuation of ansible basic tutorial series.
- In todays tutorial we are going to see how we can write ansible playbook for creating folder and copying the files.
- Check out my previous tutorial on setting up ansible on Centos, ubuntu, and ansible beginner tutorial on ad hoc commands.

Ansible Basics | Ansible Playbook Tutorial For Beginners | Automate Folder Creation And File...



```
[root@ansible1 ~]# ansible --version
ansible 2.9.10
  config file = /etc/ansible/ansible.cfg
  configured module search path = [u'/root/.ansible/plugins/modules', u'/usr/share/ansible/plugins/modules']
  ansible python module location = /usr/lib/python2.7/site-packages/ansible
  executable location = /usr/bin/ansible
  python version = 2.7.5 (default, Oct 30 2018, 23:45:53) [GCC 4.8.5 20150623 (Red Hat 4.8.5-36)]
[root@ansible1 ~]#
```

Ansible Basics | Ansible Playbook Tutorial For Beginners | Automate Folder Creation And File...



From Thetips4you

```
[root@ansible ~]# ls  
anaconda-ks.cfg  mount  
[root@ansible ~]#
```



Scroll for details



Ansible Basics | Ansible Playbook Tutorial For Beginners | Automate Folder Creation And File...



From Thetips4you

```
[root@ansible2 ~]# anaconda-ks.cfg mount  
[root@ansible2 ~]# ansible --version  
-bash: ansible: command not found  
[root@ansible2 ~]#
```



Scroll for details



Ansible Basics | Ansible Playbook Tutorial For Beginners | Automate Folder Creation And File... i b ↗

[root@ansible ~]# ansible --version
ansible 2.9.10
 config file = /etc/ansible/ansible.cfg
 configured module search path = [u'/root/.ansible/plugins/modules', u'/usr/share/ansible/plugins/modules']
 ansible python module location = /usr/lib/python2.7/site-packages/ansible
 executable location = /usr/bin/ansible
 python version = 2.7.5 (default, Oct 30 2018, 23:45:53) [GCC 4.8.5 20150623 (Red Hat 4.8.5-36)]
[root@ansible ~]#

From Thetips4you

2:37 / 18:34

Scroll for details

CC gear grid

Ansible Basics | Ansible Playbook Tutorial For Beginners | Automate Folder Creation And File...



From Thetips4you

```
[root@ansible ~]# ansible --version
ansible 2.9.10
  config file = /etc/ansible/ansible.cfg
  configured module search path = [u'/root/.ansible/plugins/modules', u'/usr/share/ansible/plugins/modules']
  ansible python module location = /usr/lib/python2.7/site-packages/ansible
  executable location = /usr/bin/ansible
  python version = 2.7.5 (default, Oct 30 2018, 23:45:53) [GCC 4.8.5 20150623 (Red Hat 4.8.5-36)]
[root@ansible ~]# cd /etc/ansible/
[root@ansible ansible]# ls
```

Ansible Basics | Ansible Playbook Tutorial For Beginners | Automate Folder Creation And File...



From Thetips4you

```
[root@ansible ~]# ansible --version
ansible 2.9.10
  config file = /etc/ansible/ansible.cfg
  configured module search path = [u'/root/.ansible/plugins/modules', u'/usr/share/ansible/plugins/modules']
  ansible python module location = /usr/lib/python2.7/site-packages/ansible
  executable location = /usr/bin/ansible
  python version = 2.7.5 (default, Oct 30 2018, 23:45:53) [GCC 4.8.5 20150623 (Red Hat 4.8.5-36)]
[root@ansible ~]# cd /etc/ansible/
[root@ansible ansible]# ls
ansible.cfg  hosts  roles
[root@ansible ansible]#
```

Ansible Basics | Ansible Playbook Tutorial For Beginners | Automate Folder Creation And File...



```
[root@ansible ~]# ansible --version
ansible 2.9.10
  config file = /etc/ansible/ansible.cfg
  configured module search path = [u'/root/.ansible/plugins/modules', u'/usr/share/ansible/plugins/modules']
  ansible python module location = /usr/lib/python2.7/site-packages/ansible
  executable location = /usr/bin/ansible
  python version = 2.7.5 (default, Oct 30 2018, 23:45:53) [GCC 4.8.5 20150623 (Red Hat 4.8.5-36)]
[root@ansible ~]# cd /etc/ansible/
[root@ansible ansible]# ls
ansible.cfg  hosts  roles
[root@ansible ansible]# cat hosts
```

Ansible Basics | Ansible Playbook Tutorial For Beginners | Automate Folder Creation And File...



- Comments begin with the '#' character
- Blank lines are ignored
- Groups of hosts are delimited by [header] elements
- You can enter hostnames or ip addresses
- A hostname/ip can be a member of multiple groups

Ex 1: Ungrouped hosts, specify before any group headers.

```
## green.example.com
## blue.example.com
## servers]
## alpha.example.org
```

Ex 2: A collection of hosts belonging to the 'webservers' group

```
[testservers]
```

```
192.168.145.145
```

```
## [webservers]
## alpha.example.org
## beta.example.org
## 192.168.1.100
## 192.168.1.110
```

If you have multiple hosts following a pattern you can specify
them like this:

Ansible Basics | Ansible Playbook Tutorial For Beginners | Automate Folder Creation And File...



```
[root@ansible ~]# ip a
1: lo: <LOOPBACK,UP,LOWER_UP> mtu 65536 qdisc noqueue state UNKNOWN group default qlen 1000
    link/loopback 00:00:00:00:00:00 brd 00:00:00:00:00:00
    inet 127.0.0.1/8 scope host lo
        valid_lft forever preferred_lft forever
    inet6 ::1/128 scope host
        valid_lft forever preferred_lft forever
2: ens33: <BROADCAST,MULTICAST,UP,LOWER_UP> mtu 1500 qdisc pfifo_fast state UP group default qlen 1000
    link/ether 00:0c:29:23:a2:10 brd ff:ff:ff:ff:ff:ff
    inet 192.168.145.145/24 brd 192.168.145.255 scope global noprefixroute dynamic ens33
        valid_lft 1107sec preferred_lft 1107sec
    inet6 fe80::5c6e:f999:e04:80a5/64 scope link noprefixroute
        valid_lft forever preferred_lft forever
[root@ansible2 ~]#
```



Ansible Basics | Ansible Playbook Tutorial For Beginners | Automate Folder Creation And File... ⓘ 🔍 ↗

```
[root@ansible ~]# anaconda-ks.cfg mount
[root@ansible ~]#
```

The video player interface at the bottom of the screen includes the following elements:

- Playback controls: back, forward, previous, next, and volume.
- Progress bar: A red line indicates the current position at 4:37 / 18:34.
- Text: "Scroll for details" with a dropdown arrow.
- Icons: CC, settings, and other video-related symbols.

Ansible Basics | Ansible Playbook Tutorial For Beginners | Automate Folder Creation And File...



```
[root@ansible ~]# anaconda-ks.cfg mount  
[root@ansible ~]# pwd  
/root  
[root@ansible ~]#
```



Scroll for details



Ansible Basics | Ansible Playbook Tutorial For Beginners | Automate Folder Creation And File...



```
[root@ansible ansible]# ansible testserver -m file -a "dest=/root/file state=directory"
192.168.145.145 | CHANGED => {
    "ansible_facts": {
        "discovered_interpreter_python": "/usr/bin/python"
    },
    "changed": true,
    "gid": 0,
    "group": "root",
    "mode": "0755",
    "owner": "root",
    "path": "/root/file",
    "secontext": "unconfined_u:object_r:admin_home_t:s0",
    "size": 6,
    "state": "directory",
    "uid": 0
}
[root@ansible ansible]#
```

Ansible Basics | Ansible Playbook Tutorial For Beginners | Automate Folder Creation And File...



```
[root@ansible ~]# ls  
anaconda-ks.cfg  mount  
[root@ansible ~]# pwd  
/root  
[root@ansible ~]# ls  
anaconda-ks.cfg  file  mount  
[root@ansible ~]#
```

Ansible Basics | Ansible Playbook Tutorial For Beginners | Automate Folder Creation And File...



```
[root@ansible1 ansible]# ansible testservers -m file -a "dest=/root/file state=directory"
192.168.145 | CHANGED => {
    "ansible_facts": {
        "discovered_interpreter_python": "/usr/bin/python"
    },
    "changed": true,
    "gid": 0,
    "group": "root",
    "mode": "0755",
    "owner": "root",
    "path": "/root/file",
    "secontext": "unconfined_u:object_r:admin_home_t:s0",
    "size": 6,
    "state": "directory",
    "uid": 0
}
[root@ansible1 ansible]# ansible testservers -m file -a "dest=/root/file state=absent"
192.168.145 | CHANGED => {
    "ansible_facts": {
        "discovered_interpreter_python": "/usr/bin/python"
    },
    "changed": true,
    "path": "/root/file",
    "state": "absent"
}
[root@ansible1 ansible]#
```

Ansible Basics | Ansible Playbook Tutorial For Beginners | Automate Folder Creation And File... ⓘ 🔍 ↗

```
[root@ansible ~]# ls  
anaconda-ks.cfg  mount  
[root@ansible ~]# pwd  
/root  
[root@ansible ~]# ls  
anaconda-ks.cfg  [REDACTED]  mount  
[root@ansible ~]# ls  
anaconda-ks.cfg  mount  
[root@ansible ~]# [REDACTED]
```

5:15 / 18:34

Scroll for details

CC ⚙️ 📈

Ansible Basics | Ansible Playbook Tutorial For Beginners | Automate Folder Creation And File...



```
    "path": "/root/file",
    "secontext": "unconfined_u:object_r:admin_home_t:s0",
    "size": 6,
    "state": "directory",
    "uid": 0
}
[root@ansible1 ansible]# ansible testservers -m file -a "dest=/root/file state=absent"
192.168.145.145 | CHANGED => {
    "ansible_facts": {
        "discovered_interpreter_python": "/usr/bin/python"
    },
    "changed": true,
    "path": "/root/file",
    "state": "absent"
}
[root@ansible1 ansible]# ansible testservers -m file -a "dest=/root/file.txt state=touch"
192.168.145.145 | CHANGED => {
    "ansible_facts": {
        "discovered_interpreter_python": "/usr/bin/python"
    },
    "changed": true,
    "dest": "/root/file.txt",
    "gid": 0,
    "group": "root",
    "mode": "0644",
    "owner": "root",
    "secontext": "unconfined_u:object_r:admin_home_t:s0",
    "size": 0,
    "state": "file",
    "uid": 0
}
```

Ansible Basics | Ansible Playbook Tutorial For Beginners | Automate Folder Creation And File...



```
[root@ansible ~]# ls  
anaconda-ks.cfg  mount  
[root@ansible ~]# pwd  
/root  
[root@ansible ~]# ls  
anaconda-ks.cfg  file  mount  
[root@ansible ~]# ls  
anaconda-ks.cfg  mount  
[root@ansible ~]# ls  
anaconda-ks.cfg  file.txt  mount  
[root@ansible ~]#
```

Ansible Basics | Ansible Playbook Tutorial For Beginners | Automate Folder Creation And File...



```
[root@ansible2 ~]# ls  
anaconda-ks.cfg  mount  
[root@ansible2 ~]# pwd  
/root  
[root@ansible2 ~]# ls  
anaconda-ks.cfg  file  mount  
[root@ansible2 ~]# ls  
anaconda-ks.cfg  mount  
[root@ansible2 ~]# ls  
anaconda-ks.cfg  file.txt  mount  
[root@ansible2 ~]# ls  
anaconda-ks.cfg  mount  
[root@ansible2 ~]# c
```

Ansible Basics | Ansible Playbook Tutorial For Beginners | Automate Folder Creation And File...



```
[root@ansible1 ansible]# ansible testservers -m file -a "dest=/root/file.txt state=touch"
192.168.145.145 | CHANGED => {
    "ansible_facts": {
        "discovered_interpreter_python": "/usr/bin/python"
    },
    "changed": true,
    "dest": "/root/file.txt",
    "gid": 0,
    "group": "root",
    "mode": "0644",
    "owner": "root",
    "secontext": "unconfined_u:object_r:admin_home_t:s0",
    "size": 0,
    "state": "file",
    "uid": 0
}
[root@ansible1 ansible]# ansible testservers -m file -a "dest=/root/file.txt state=absent"
192.168.145.145 | CHANGED => {
    "ansible_facts": {
        "discovered_interpreter_python": "/usr/bin/python"
    },
    "changed": true,
    "path": "/root/file.txt",
    "state": "absent"
}
```

[root@ansible1 ansible]# 5:48 / 18:34

Scroll for details



Ansible Basics | Ansible Playbook Tutorial For Beginners | Automate Folder Creation And File...



FOLDERS

- `src/demoapp`
- `src/node_modules`
- `src/routes`
- `src/views`
- + `src/app.js`
- + `src/package-lock.json`
- + `src/package.json`

```
1  ---
2  - name: create folder server and user details
3  hosts: testservers
4  remote_user: root
5
6  tasks:
7  - name: creating folder
8    file: dest=/root/file state=directory
```

Ansible Basics | Ansible Playbook Tutorial For Beginners | Automate Folder Creation And File...



```
[root@ansible1 ansible]# ls  
ansible.cfg  hosts  roles  
[root@ansible1 ansible]# vi ansiblefolder.yml
```



8:29 / 18:34

Scroll for details



Ansible Basics | Ansible Playbook Tutorial For Beginners | Automate Folder Creation And File...



```
---
```

```
- name: create folder server and user details
  hosts: testservers
  remote_user: root

  tasks:
    - name: creating folder
      file: dest=/root/file state=directory
```

Ansible Basics | Ansible Playbook Tutorial For Beginners | Automate Folder Creation And File...



```
[root@ansible1 ansible]# ls  
ansible.cfg hosts roles  
[root@ansible1 ansible]# vi ansiblefolder.yml  
[root@ansible1 ansible]# ls  
ansible.cfg ansiblefolder.yml hosts roles  
[root@ansible1 ansible]#
```

Ansible Basics | Ansible Playbook Tutorial For Beginners | Automate Folder Creation And File...



```
[root@ansible1 ansible]# ls
ansible.cfg hosts roles
[root@ansible1 ansible]# vi ansiblefolder.yml
[root@ansible1 ansible]# ls
ansible.cfg ansiblefolder.yml hosts roles
[root@ansible1 ansible]# pwd
/etc/ansible
[root@ansible1 ansible]# mv ./ansiblefolder.yml /root/
[root@ansible1 ansible]# cd ~
[root@ansible1 ~]# ls
anaconda-ks.cfg ansiblefolder.yml nfs
[root@ansible1 ~]# pwd
/root
[root@ansible1 ~]# cle
```

Ansible Basics | Ansible Playbook Tutorial For Beginners | Automate Folder Creation And File... ⓘ ⓘ ⌂

```
[root@ansible ~]# ls  
anaconda-ks.cfg ansiblefolder.yml nfs  
[root@ansible ~]#
```

The video player interface at the bottom of the screen includes the following elements:

- Playback controls: back, forward, previous, next, and volume.
- Time display: 9:35 / 18:34.
- Progress bar: A horizontal bar indicating the video's progress, with a red segment on the left and a grey segment on the right.
- Text input field: "Scroll for details" with a dropdown arrow.
- Control icons: CC, settings, and other media-related icons.

Ansible Basics | Ansible Playbook Tutorial For Beginners | Automate Folder Creation And File... ⓘ 🔍 ↗

```
[root@ansible ~]# anaconda-ks.cfg ansiblefolder.yml nfs
[root@ansible ~]# ansible-playbook ansiblefolder.yml
```

9:50 / 18:34

Scroll for details

CC ⚙️ 📺

Ansible Basics | Ansible Playbook Tutorial For Beginners | Automate Folder Creation And File...



```
[root@ansible ~]# ls  
anaconda-ks.cfg  ansiblefolder.yml  nfs  
[root@ansible ~]# ansible-playbook ansiblefolder.yml  
  
PLAY [create folder server and user details] *****  
  
TASK [Gathering Facts] *****  
ok: [192.168.145.145]  
  
TASK [creating folder] *****  
changed: [192.168.145.145]  
  
PLAY RECAP *****  
192.168.145.145 : ok=2    changed=1    unreachable=0    failed=0    skipped=0    rescued=0    ignored=0  
  
[root@ansible ~]#
```

Ansible Basics | Ansible Playbook Tutorial For Beginners | Automate Folder Creation And File...



```
[root@ansible ~]# ls  
anaconda-ks.cfg  file  mount
```

```
[root@ansible ~]#
```

Ansible Basics | Ansible Playbook Tutorial For Beginners | Automate Folder Creation And File...



```
[root@ansible ~]# ls
anaconda-ks.cfg  ansiblefolder.yml  nfs
[root@ansible ~]# ansible-playbook ansiblefolder.yml

PLAY [create folder server and user details] *****

TASK [Gathering Facts] *****
ok: [192.168.145.145]

TASK [creating folder] *****
changed: [192.168.145.145]

PLAY RECAP *****
192.168.145.145      : ok=2    changed=1    unreachable=0    failed=0    skipped=0    rescued=0    ignored=0

[root@ansible ~]# ls
anaconda-ks.cfg  ansiblefolder.yml  nfs
[root@ansible ~]# v
```

Ansible Basics | Ansible Playbook Tutorial For Beginners | Automate Folder Creation And File...



```
[root@ansible1 ~]# ls
anaconda-ks.cfg  ansiblefolder.yml  nfs
[root@ansible1 ~]# ansible-playbook ansiblefolder.yml

PLAY [create folder server and user details] *****

TASK [Gathering Facts] *****
ok: [192.168.145.145]

TASK [creating folder] *****
changed: [192.168.145.145]

PLAY RECAP *****
192.168.145.145      : ok=2    changed=1    unreachable=0    failed=0    skipped=0    rescued=0    ignored=0

[root@ansible1 ~]# ls
anaconda-ks.cfg  ansiblefolder.yml  nfs
[root@ansible1 ~]# vi ansiblefolder.yml
```

```
---
```

```
- name: create folder server and user details
  hosts: testservers
  remote_user: root

  tasks:
    - name: creating folder
      file: dest=/root/file state=absent
```

Ansible Basics | Ansible Playbook Tutorial For Beginners | Automate Folder Creation And File...



```
[root@ansible ~]# ls
anaconda-ks.cfg  ansiblefolder.yml  nfs
[root@ansible ~]# ansible-playbook ansiblefolder.yml

PLAY [create folder server and user details] *****

TASK [Gathering Facts] *****
ok: [192.168.145.145]

TASK [creating folder] *****
changed: [192.168.145.145]

PLAY RECAP *****
192.168.145.145      : ok=2    changed=1    unreachable=0    failed=0    skipped=0    rescued=0    ignored=0

[root@ansible ~]# ls
anaconda-ks.cfg  ansiblefolder.yml  nfs
[root@ansible ~]# vi ansiblefolder.yml
[root@ansible ~]# ansible-playbook ansiblefolder.yml
```

Ansible Basics | Ansible Playbook Tutorial For Beginners | Automate Folder Creation And File...



```
[root@ansible1 ~]# ls
anaconda-ks.cfg  ansiblefolder.yml  nfs
[root@ansible1 ~]# ansible-playbook ansiblefolder.yml

PLAY [create folder server and user details] *****

TASK [Gathering Facts] *****
ok: [192.168.145.145]

TASK [creating folder] *****
changed: [192.168.145.145]

PLAY RECAP *****
192.168.145.145      : ok=2    changed=1    unreachable=0    failed=0    skipped=0    rescued=0    ignored=0

[root@ansible1 ~]# ls
anaconda-ks.cfg  ansiblefolder.yml  nfs
[root@ansible1 ~]# vi ansiblefolder.yml
[root@ansible1 ~]# ansible-playbook ansiblefolder.yml

PLAY [create folder server and user details] *****

TASK [Gathering Facts] *****
ok: [192.168.145.145]

TASK [creating folder] *****
changed: [192.168.145.145]

PLAY RECAP *****
192.168.145.145      : ok=2    changed=1    unreachable=0    failed=0    skipped=0    rescued=0    ignored=0
```

[root@ansible1 ~]# 11:19 / 18:34

Scroll for details



Ansible Basics | Ansible Playbook Tutorial For Beginners | Automate Folder Creation And File...



```
[root@ansible ~]# ls  
anaconda-ks.cfg  file  mount  
[root@ansible ~]# ls  
anaconda-ks.cfg  mount  
[root@ansible ~]#
```

Ansible Basics | Ansible Playbook Tutorial For Beginners | Automate Folder Creation And File...



```
[root@ansible1 ~]# ls
anaconda-ks.cfg  ansiblefolder.yml  nfs
[root@ansible1 ~]# ansible-playbook ansiblefolder.yml

PLAY [create folder server and user details] *****

TASK [Gathering Facts] *****
ok: [192.168.145.145]

TASK [creating folder] *****
changed: [192.168.145.145]

PLAY RECAP *****
192.168.145.145      : ok=2    changed=1    unreachable=0    failed=0    skipped=0    rescued=0    ignored=0

[root@ansible1 ~]# ls
anaconda-ks.cfg  ansiblefolder.yml  nfs
[root@ansible1 ~]# vi ansiblefolder.yml
[root@ansible1 ~]# ansible-playbook ansiblefolder.yml

PLAY [create folder server and user details] *****

TASK [Gathering Facts] *****
ok: [192.168.145.145]

TASK [creating folder] *****
changed: [192.168.145.145]

PLAY RECAP *****
192.168.145.145      : ok=2    changed=1    unreachable=0    failed=0    skipped=0    rescued=0    ignored=0
```

Ansible Basics | Ansible Playbook Tutorial For Beginners | Automate Folder Creation And File...



```
[root@ansible ~]# ls  
anaconda-ks.cfg  ansiblefolder.yml  nfs  
[root@ansible ~]#
```

◀ ▶ ▶| 🔍 12:18 / 18:34

Scroll for details



Ansible Basics | Ansible Playbook Tutorial For Beginners | Automate Folder Creation And File... ⓘ 🔍 ↗

```
[root@ansible1 ~]# ls  
anaconda-ks.cfg ansiblefolder.yml nfs  
[root@ansible1 ~]# mkdir demo  
[root@ansible1 ~]# ls  
anaconda-ks.cfg ansiblefolder.yml demo nfs  
[root@ansible1 ~]#
```

12:23 / 18:34

Scroll for details

CC ⚙️ 🔍

Ansible Basics | Ansible Playbook Tutorial For Beginners | Automate Folder Creation And File... ⓘ 🔍 ↗

```
[root@ansible1 ~]# ls  
anaconda-ks.cfg ansiblefolder.yml nfs  
[root@ansible1 ~]# mkdir demo  
[root@ansible1 ~]# ls  
anaconda-ks.cfg ansiblefolder.yml demo nfs  
[root@ansible1 ~]# cd demo/  
[root@ansible1 demo]# ls  
[root@ansible1 demo]#
```

12:28 / 18:34

Scroll for details

CC

Ansible Basics | Ansible Playbook Tutorial For Beginners | Automate Folder Creation And File...



```
[root@ansible1 ~]# ls  
anaconda-ks.cfg  ansiblefolder.yml  nfs  
[root@ansible1 ~]# mkdir demo  
[root@ansible1 ~]# ls  
anaconda-ks.cfg  ansiblefolder.yml  demo  nfs  
[root@ansible1 ~]# cd demo/  
[root@ansible1 demo]# ls  
[root@ansible1 demo]# vi test.txt
```

Ansible Basics | Ansible Playbook Tutorial For Beginners | Automate Folder Creation And File...



This is a demo text file for ansible



Scroll for details



Ansible Basics | Ansible Playbook Tutorial For Beginners | Automate Folder Creation And File...



```
[root@ansible1 ~]# ls  
anaconda-ks.cfg  ansiblefolder.yml  nfs  
[root@ansible1 ~]# mkdir demo  
[root@ansible1 ~]# ls  
anaconda-ks.cfg  ansiblefolder.yml  demo  nfs  
[root@ansible1 ~]# cd demo/  
[root@ansible1 demo]# ls  
[root@ansible1 demo]# vi test.txt  
[root@ansible1 demo]# cat test.txt
```

```
This is a demo text file for ansible  
[root@ansible1 demo]#
```

Ansible Basics | Ansible Playbook Tutorial For Beginners | Automate Folder Creation And File... ⓘ ⓘ ⌂

```
[root@ansible1 ~]# cd ..
[root@ansible1 ~]# ls
anaconda-ks.cfg  ansiblefolder.yml  demo  nfs
[root@ansible1 ~]#
```

13:00 / 18:34

Scroll for details

Ansible Basics | Ansible Playbook Tutorial For Beginners | Automate Folder Creation And File...



```
[root@ansible ~]# ls  
anaconda-ks.cfg  file  mount  
[root@ansible ~]# ls  
anaconda-ks.cfg  mount  
[root@ansible ~]# ls  
anaconda-ks.cfg  mount  
[root@ansible ~]#
```

Ansible Basics | Ansible Playbook Tutorial For Beginners | Automate Folder Creation And File... ⓘ 🔍 ↗

```
[root@ansible2 ~]# ls  
anaconda-ks.cfg  file  mount  
[root@ansible2 ~]# ls  
anaconda-ks.cfg  mount  
[root@ansible2 ~]# ls  
anaconda-ks.cfg  mount  
[root@ansible2 ~]# mkdir demoonclient  
[root@ansible2 ~]# ls  
anaconda-ks.cfg  demoonclient  mount  
[root@ansible2 ~]#
```

13:16 / 18:34

Scroll for details

Ansible Basics | Ansible Playbook Tutorial For Beginners | Automate Folder Creation And File...



FOLDERS

- demoapp
- node_modules
- routes
- views
- / app.js
- / package-lock.json
- / package.json

```
1  ---  
2  -> name: create folder server and user details  
3  hosts: testservers  
4  remote_user: root  
5  
6  tasks:  
7  - name: creating folder  
8    file: dest=/root/file state=directory  
9  
10
```

Ansible Basics | Ansible Playbook Tutorial For Beginners | Automate Folder Creation And File...



```
[root@ansible ~]# ls  
anaconda-ks.cfg  file  mount  
[root@ansible2 ~]# ls  
anaconda-ks.cfg  mount  
[root@ansible2 ~]# ls  
anaconda-ks.cfg  mount  
[root@ansible2 ~]# mkdir demoonclient  
[root@ansible2 ~]# ls  
anaconda-ks.cfg  demoonclient  mount  
[root@ansible2 ~]# rm -rf demoonclient/  
[root@ansible2 ~]#
```

Ansible Basics | Ansible Playbook Tutorial For Beginners | Automate Folder Creation And File...



```
[root@ansible ~]# ls  
anaconda-ks.cfg  file  mount  
[root@ansible2 ~]# ls  
anaconda-ks.cfg  mount  
[root@ansible2 ~]# ls  
anaconda-ks.cfg  mount  
[root@ansible2 ~]# mkdir demoonclient  
[root@ansible2 ~]# ls  
anaconda-ks.cfg  demoonclient  mount  
[root@ansible2 ~]# rm -rf demoonclient/
```

Ansible Basics | Ansible Playbook Tutorial For Beginners | Automate Folder Creation And File...



```
[root@ansible ~]# ls  
anaconda-ks.cfg  file  mount  
[root@ansible ~]# ls  
anaconda-ks.cfg  mount  
[root@ansible ~]# ls  
anaconda-ks.cfg  mount  
[root@ansible ~]# mkdir demoonclient  
[root@ansible ~]# ls  
anaconda-ks.cfg  demoonclient  mount  
[root@ansible ~]# rm -rf demoonclient/  
[root@ansible ~]# ls  
anaconda-ks.cfg  mount  
[root@ansible ~]#
```

Ansible Basics | Ansible Playbook Tutorial For Beginners | Automate Folder Creation And File...



FOLDERS

- |- .gitignore
- |- demoapp
- |- node_modules
- |- routes
- |- views
- |- app.js
- |- package-lock.json
- |- package.json

```
1  -----
2  2   - name: create folder server and user details
3  3     BHOSTS: testservers
4  4     remote_user: root
5  5
6  6     tasks:
7  7       - name: creating folder
8  8         file: dest=/root/file state=directory
```

Ansible Basics | Ansible Playbook Tutorial For Beginners | Automate Folder Creation And File...



FOLDERS

- > demoapp
- > node_modules
- > routes
- > views
- /> app.js
- /> package-lock.json
- /> package.json

Ansibleplaybook.yml

```
3  ---
4  - name: create folder server and user details
5    hosts: testservers
6    remote_user: root
7
8    tasks:
9      - name: creating folder
10        file: dest=/root/demconcilient state=directory
```

Ansible Basics | Ansible Playbooks

youtube.com is now full screen

Exit Full Screen (Esc)

Creation And File...



```
[root@ansible1 ~]# ls  
anaconda-ks.cfg  ansiblefolder.yml  demo  nfs  
[root@ansible1 ~]#
```

◀ ▶ ⏪ ⏩ 🔍 14:47 / 18:34

Scroll for details

CC ⚙ ⌂

Ansible Basics | Ansible Playbook Tutorial For Beginners | Automate Folder Creation And File...



FOLDERS

- app
- demoapp
 - node_modules
 - routes
 - views
- > app.js
- > package-lock.json
- > package.json

```
1: ---
2:   - name: create folder server and user details
3:     hosts: testservers
4:     remote_user: root
5:
6:     tasks:
7:       - name: creating folder
8:         file: dest=/root/democlient state=directory
9:
10:      - name: copying the file from demo folder to democlient
11:        copy: src=/root/demo/test.txt
```

Ansible Basics | Ansible Playbook Tutorial For Beginners | Automate Folder Creation

Suggested: Kubernetes

```
[root@ansible1 demo]# cd ..
[root@ansible1 ~]# ls
anaconda-ks.cfg  ansiblefolder.yml  demo  nfs
[root@ansible1 ~]# pwd
/rOOT
[root@ansible1 ~]#
```

Play (k)

◀ ▶ ▶ 🔍 14:50 / 18:34

Scroll for details



Ansible Basics | Ansible Playbook Tutorial For Beginners | Automate Folder Creation And File... ⓘ 🔍 ↗

```
[root@ansible1 demo]# cd ..
[root@ansible1 ~]# ls
anaconda-ks.cfg  ansiblefolder.yml  demo  nfs
[root@ansible1 ~]# pwd
/root
[root@ansible1 ~]# cd demo/
[root@ansible1 demo]# ls
test.txt
[root@ansible1 demo]#
```

◀ ▶ ⏪ ⏹ 14:56 / 18:34

Scroll for details

CC ⚙ ⌂

Ansible Basics | Ansible Playbook Tutorial For Beginners | Automate Folder Creation And File...



FOLDERS

- |- demoapp
- |- node_modules
- |- routes
- |- views
- |- app.js
- |- package-lock.json
- |- package.json

Ansibleplaybook.yml

```
1  ---  
2  - name: create folder server and user details  
3    hosts: testservers  
4    remote_user: root  
5  
5  tasks:  
6    - name: creating folder  
7      file: dest=/root/democlient state=directory  
8  
8    - name: copying the file from demo folder to democlient  
9      copy: src=/root/demo/test.txt dest=/root/democlient/
```

Ansible Basics | Ansible Playbook Tutorial For Beginners | Automate Folder Creation And File...



```
[root@ansible1 demo]# cd ..
[root@ansible1 ~]# ls
anaconda-ks.cfg  ansiblefolder.yml  demo  nfs
[root@ansible1 ~]# pwd
/root
[root@ansible1 ~]# cd demo/
[root@ansible1 demo]# ls
test.txt
[root@ansible1 demo]# pwd
/root/demo
[root@ansible1 demo]# cd ..
[root@ansible1 ~]# ls
anaconda-ks.cfg  ansiblefolder.yml  demo  nfs
[root@ansible1 ~]# vi ansiblecopy.yml
```

Ansible Basics | Ansible Playbook Tutorial For Beginners | Automate Folder Creation And File...



```
---
```

```
- name: create folder server and user details
  hosts: testservers
  remote_user: root

  tasks:
    - name: creating folder
      file: dest=/root/demoonclient state=directory

    - name: copying the file from demo folder to demoonclient
      copy: src=/root/demo/test.txt dest=/root/demoonclient/
```

Ansible Basics | Ansible Playbook Tutorial For Beginners | Automate Folder Creation And File...



```
[root@ansible1 demo]# cd ..
[root@ansible1 ~]# ls
anaconda-ks.cfg  ansiblefolder.yml  demo  nfs
[root@ansible1 ~]# pwd
/root
[root@ansible1 ~]# cd demo/
[root@ansible1 demo]# ls
test.txt
[root@ansible1 demo]# pwd
/root/demo
[root@ansible1 demo]# cd ..
[root@ansible1 ~]# ls
anaconda-ks.cfg  ansiblefolder.yml  demo  nfs
[root@ansible1 ~]# vi ansiblecopy.yml
[root@ansible1 ~]# ls
anaconda-ks.cfg  ansiblecopy.yml  ansiblefolder.yml  demo  nfs
[root@ansible1 ~]# c
```

Ansible Basics | Ansible Playbook Tutorial For Beginners | Automate Folder Creation And File...



```
[root@ansible1 demo]# cd ..
[root@ansible1 ~]# ls
anaconda-ks.cfg  ansiblefolder.yml  demo  nfs
[root@ansible1 ~]# pwd
/root
[root@ansible1 ~]# cd demo/
[root@ansible1 demo]# ls
test.txt
[root@ansible1 demo]# pwd
/root/demo
[root@ansible1 demo]# cd ..
[root@ansible1 ~]# ls
anaconda-ks.cfg  ansiblefolder.yml  demo  nfs
[root@ansible1 ~]# vi ansiblecopy.yml
[root@ansible1 ~]# ls
anaconda-ks.cfg  ansiblecopy.yml  ansiblefolder.yml  demo  nfs
[root@ansible1 ~]# cat ansiblecopy.yml
```

```
---
- name: create folder server and user details
  hosts: testservers
  remote_user: root

  tasks:
    - name: creating folder
      file: dest=/root/demoonclient state=directory

    - name: copying the file from demo folder to demoonclient
      copy: src=/root/demo/test.txt dest=/root/demoonclient/
[root@ansible1 ~]
```

Ansible Basics | Ansible Playbook Tutorial For Beginners | Automate Folder Creation And File...



```
[root@ansible ~]# ansible-playbook ansiblecopy.yml

PLAY [create folder server and user details] *****

TASK [Gathering Facts] *****
ok: [192.168.145.145]

TASK [creating folder] *****
changed: [192.168.145.145]

TASK [copying the file from demo folder to demoonclient] *****
changed: [192.168.145.145]

PLAY RECAP *****
192.168.145.145      : ok=3    changed=2    unreachable=0    failed=0    skipped=0    rescued=0    ignored=0

[root@ansible ~]#
```

Ansible Basics | Ansible Playbook Tutorial For Beginners | Automate Folder Creation And File...



```
[root@ansible ~]# ls  
anaconda-ks.cfg  file  mount  
[root@ansible ~]# ls  
anaconda-ks.cfg  mount  
[root@ansible ~]# ls  
anaconda-ks.cfg  mount  
[root@ansible ~]# mkdir demoonclient  
[root@ansible ~]# ls  
anaconda-ks.cfg  demoonclient  mount  
[root@ansible ~]# rm -rf demoonclient/  
[root@ansible ~]# ls  
anaconda-ks.cfg  mount  
[root@ansible ~]# ls  
anaconda-ks.cfg  mount  
[root@ansible ~]# ls  
anaconda-ks.cfg  demoonclient  mount  
[root@ansible ~]#
```

Ansible Basics | Ansible Playbook Tutorial For Beginners | Automate Folder Creation And File...



```
[root@ansible ~]# ls  
anaconda-ks.cfg  file  mount  
[root@ansible ~]# ls  
anaconda-ks.cfg  mount  
[root@ansible ~]# ls  
anaconda-ks.cfg  mount  
[root@ansible ~]# mkdir demoonclient  
[root@ansible ~]# ls  
anaconda-ks.cfg  demoonclient  mount  
[root@ansible ~]# rm -rf demoonclient/  
[root@ansible ~]# ls  
anaconda-ks.cfg  mount  
[root@ansible ~]# ls  
anaconda-ks.cfg  mount  
[root@ansible ~]# ls  
anaconda-ks.cfg  demoonclient  mount  
[root@ansible ~]# cd demoonclient/  
[root@ansible demoonclient]#
```

Ansible Basics | Ansible Playbook Tutorial For Beginners | Automate Folder Creation And File...



```
[root@ansible ~]# ls  
anaconda-ks.cfg  file  mount  
[root@ansible ~]# ls  
anaconda-ks.cfg  mount  
[root@ansible ~]# ls  
anaconda-ks.cfg  mount  
[root@ansible ~]# mkdir demoonclient  
[root@ansible ~]# ls  
anaconda-ks.cfg  demoonclient  mount  
[root@ansible ~]# rm -rf demoonclient/  
[root@ansible ~]# ls  
anaconda-ks.cfg  mount  
[root@ansible ~]# ls  
anaconda-ks.cfg  mount  
[root@ansible ~]# ls  
anaconda-ks.cfg  demoonclient  mount  
[root@ansible ~]# cd demoonclient/  
[root@ansible demoonclient]# ls  
test.txt  
[root@ansible demoonclient]#
```

Ansible Basics | Ansible Playbook Tutorial For Beginners | Automate Folder Creation And File...



```
[root@ansible ~]# ls  
anaconda-ks.cfg  file  mount  
[root@ansible ~]# ls  
anaconda-ks.cfg  mount  
[root@ansible ~]# ls  
anaconda-ks.cfg  mount  
[root@ansible ~]# mkdir demoonclient  
[root@ansible ~]# ls  
anaconda-ks.cfg  demoonclient  mount  
[root@ansible ~]# rm -rf demoonclient/  
[root@ansible ~]# ls  
anaconda-ks.cfg  mount  
[root@ansible ~]# ls  
anaconda-ks.cfg  mount  
[root@ansible ~]# ls  
anaconda-ks.cfg  demoonclient  mount  
[root@ansible ~]# cd demoonclient/  
[root@ansible demoonclient]# ls  
test.txt  
[root@ansible demoonclient]# cat test.txt  
  
This is a demo text file for ansible  
[root@ansible demoonclient]#
```

Hello Friends, Welcome back to my channel

- We are going to see interesting Ansible Playbook tutorial in this video.
- In this video, we are going to see how we can deploy a LAMP stack and publish your website using Ansible.

Hello Friends, Welcome back to my channel

- We are going to see interesting Ansible Playbook tutorial in this video.
- In this video, we are going to see how we can deploy a LAMP stack and publish your website using Ansible.
- We are going to write the complete Ansible Playbook to Install Apache, MYSQL database, PHP on CentOS machine and deploy your webpage.

Test your knowledge!

Q) What this Ansible Ad Hoc command do ?
`ansible testservers -a "/sbin/reboot"`

Test your knowledge!

Q) What this Ansible Ad Hoc command do ?
`ansible testservers -a "/sbin/reboot"`

- A) Open Command Prompt
- B) Create folder reboot
- C) Reboot the testservers collection
- D) Install ansible and reboot

Test your knowledge!

Q) What this Ansible Ad Hoc command do ?
`ansible testservers -a "/sbin/reboot"`

- A) Open Command Prompt
- B) Create folder reboot
- C) Reboot the testservers collection
- D) Install ansible and reboot

Enter your answer in Comment section

Ansible LAMP Stack Playbook Tutorial | How To Install LAMP Stack On Centos Using Ansible

```
[root@ansible1 ~]# cd /etc/ansible/
[root@ansible1 ansible]# ls
ansible.cfg  hosts  roles
[root@ansible1 ansible]#
```

2:37 / 17:03

Scroll for details

Ansible LAMP Stack Playbook Tutorial | How To Install LAMP Stack On Centos Using Ansible

```
[root@ansible ~]# cd /etc/ansible/  
[root@ansible ansible]# ls  
ansible.cfg hosts roles  
[root@ansible ansible]# cat hosts
```

2:41 / 17:03

Scroll for details

Ansible LAMP Stack Playbook Tutorial | How To Install LAMP Stack On Centos Using Ansible



```
[testservers]
192.168.145.146

## [webservers]
## alpha.example.org
## beta.example.org
## 192.168.1.100
## 192.168.1.110

# If you have multiple hosts following a pattern you can specify
# them like this:

## www[001:006].example.com

# Ex 3: A collection of database servers in the 'dbservers' group

## [dbservers]
##
## db01.intranet.mydomain.net
## db02.intranet.mydomain.net
## 10.25.1.56
## 10.25.1.57

# Here's another example of host ranges, this time there are no
# leading 0s:

## db-[99:101]-node.example.com
```

[root@ansibile ~]# 2:48 / 17:03

Scroll for details



Ansible LAMP Stack Playbook Tutorial | How To Install LAMP Stack On Centos Using Ansible



```
[root@localhost ~]# ip addr
1: lo: <LOOPBACK,UP,LOWER_UP> mtu 65536 qdisc noqueue state UNKNOWN group default qlen 1000
    link/loopback 00:00:00:00:00:00 brd 00:00:00:00:00:00
    inet 127.0.0.1/8 scope host lo
        valid_lft forever preferred_lft forever
    inet6 ::1/128 scope host
        valid_lft forever preferred_lft forever
2: ens33: <BROADCAST,MULTICAST,UP,LOWER_UP> mtu 1500 qdisc pfifo_fast state UP group default qlen 1000
    link/ether 00:0c:29:ee:1c:fe brd ff:ff:ff:ff:ff:ff
    inet 192.168.145.146/24 brd 192.168.145.255 scope global noprefixroute dynamic ens33
        valid_lft 1702sec preferred_lft 1702sec
    inet6 fe80::1245:15f8:cae4:d2e/64 scope link noprefixroute
        valid_lft forever preferred_lft forever
[root@localhost ~]#
```

Ansible LAMP Stack Playbook Tutorial | How To Install LAMP Stack On Centos Using Ansible

POLARIS

•  [democracy](#)



Scroll for details



Ansible LAMP Stack Playbook Tutorial | How To Install LAMP Stack On Centos Using Ansible

FOLDERS

```
1  ---  
2  - hosts: testservers  
3  remote_user: root  
4  
5  
6  tasks:  
7      - name: Installing Apache  
8          yum:  
9              name: Httpd  
10             status: installed  
11  
12      - name: Enabling httpd service and start the service  
13          service:  
14              name: Httpd  
15              enabled: yes  
16              state: started  
17  
18      - name: Install MySQL DB Community edition  
19          yum:  
20              name:  
21                  - mariadb-server  
22                  - mariadb  
23              status: latest  
24  
25      - name: Start the MySQL service  
26          service:  
27              name: mysqld  
28              enabled: yes  
29              state: started  
30  
31  
32      - name: Installing PHP  
33          yum:  
34              name:  
35                  - php  
36                  - php-mysql  
37              status: latest  
38  
39      - name: Copying your php file in the /var/www/html  
40          copy:  
41              src: /root/ansible/leaveweb/index  
42              dest: /var/www/html
```

Ansible LAMP Stack Playbook Tutorial | How To Install LAMP Stack On Centos Using Ansible



FOLDERS

demapp

```
19    name: httpd
20      state: installed
21
22    - name: Enabling httpd service and start the service
23      service:
24        name: httpd
25        enabled: yes
26        state: started
27
28    - name: Install MySQL community edition
29      yum:
30        name:
31          - mariadb-server
32          - mariadb
33        state: latest
34
35    - name: Start the MySQL service
36      service:
37        name: mysqld
38        enabled: yes
39        state: started
40
41    - name: Installing PHP
42      yum:
43        name:
44          - php
45          - php-mysql
46        state: latest
47
48    - name: Copying your php file in the /var/www/html
49      copy:
50        src: /root/ansibleplaybook/index.php
51        dest: /var/www/html/index.php
```

Ansible LAMP Stack Playbook Tutorial | How To Install LAMP Stack On Centos Using Ansible



FOLDERS

demoapp

```
1: <?php>
2: <?php>
3: <?><?php>Ansible Application</?><?>
4: <?php>
5: <?php>
6:
7: </?php>
8: echo "Hello From the tips4you";
9:
10: </?php>
11: </?php>
```

Ansible LAMP Stack Playbook Tutorial | How To Install LAMP Stack On Centos Using Ansible



FOLDERS

demapp

```
1 ---  
2   - name: Installing Apache  
3     yum:  
4       name: httpd  
5       state: installed  
6  
7   - name: Enabling httpd service and start the service  
8     service:  
9       name: Httpd  
10      enabled: yes  
11      state: started  
12  
13   - name: Install MySQL DB Community edition  
14     yum:  
15       name:  
16         - mariadb-server  
17         - mariadb  
18       state: latest  
19  
20   - name: Start the MySQL service  
21     service:  
22       name: mysqld  
23       enabled: yes  
24       state: started  
25  
26  
27   - name: Installing PHP  
28     yum:  
29       name:  
30         - php  
31         - php-mysql  
32       state: latest  
33  
34   - name: Copying your php file in the /var/www/html  
35     copy:  
36       src: /root/ansibleplaybook/index.php  
37       dest: /var/www/html/index.php  
38  
39  
40  
41  
42  
43  
44  
45
```

◀ ▶ ⏪ ⏩ 🔍 11:13 / 17:03

Scroll for details



Ansible LAMP Stack Playbook Tutorial | How To Install LAMP Stack On Centos Using Ansible



```
[root@ansible1 ansible]# cd /root/ansibleplaybook/  
[root@ansible1 ansibleplaybook]# ls  
[root@ansible1 ansibleplaybook]#
```

◀ ▶ ⏪ ⏩ 🔍 11:37 / 17:03

Scroll for details





Ansible LAMP Stack Playbook Tutorial | How To Install LAMP Stack On Centos Using Ansible



```
state: started

- name: Installing MySQL Community Server
  yum:
    name:
      - mariadb-server
      - mariadb
    state: latest

- name: Starting the MySQL DB service
  service:
    name: mysqld
    enabled: yes
    state: started

- name: Installing PHP
  yum:
    name:
      - php
      - php-mysql
    state: latest

- name: Copying the PHP test.php file
  copy:
    src: /root/ansibleplaybook/index.php
    dest: /var/www/html/index.php
```

Ansible LAMP Stack Playbook Tutorial | How To Install LAMP Stack On Centos Using Ansible



```
---
```

```
- hosts: testservers
  remote_user: root

tasks:

- name: Installing Apache2
  yum:
    name: httpd
    state: installed

- name: Enabling and starting the apache service
  service:
    name: httpd
    enabled: yes
    state: started

- name: Installing MYSQL Community Server
  yum:
    name:
      - mariadb-server
      - mariadb
    state: latest

- name: Starting the MySQL DB service
  service:
    name: mysqld
    enabled: yes
    state: started
```

INSERT ►|◀ 12:02 / 17:03

Scroll for details



Ansible LAMP Stack Playbook Tutorial | How To Install LAMP Stack On Centos Using Ansible



```
[root@ansible1 ansibleplaybook]# vi lampstack.yml  
[root@ansible1 ansibleplaybook]# ls  
lampstack.yml  
[root@ansible1 ansibleplaybook]#
```

Ansible LAMP Stack Playbook Tutorial | How To Install LAMP Stack On Centos Using Ansible



```
[root@ansible1 ansibleplaybook]# vi lampstack.yml  
[root@ansible1 ansibleplaybook]# ls  
lampstack.yml  
[root@ansible1 ansibleplaybook]# vi index.php
```

Ansible LAMP Stack Playbook Tutorial | How To Install LAMP Stack On Centos Using Ansible



```
<html>
<head>
  <title>Ansible Application</title>
</head>
<body>

  <?php
    echo " Hello From Thetips4you"
  ?>

</body>
</html>
```

Ansible LAMP Stack Playbook Tutorial | How To Install LAMP Stack On Centos Using Ansible



```
[root@ansible1 ansibleplaybook]# vi lampstack.yml
[root@ansible1 ansibleplaybook]# ls
lampstack.yml
[root@ansible1 ansibleplaybook]# vi index.php
[root@ansible1 ansibleplaybook]# ls
index.php  lampstack.yml
[root@ansible1 ansibleplaybook]#
```

Ansible LAMP Stack Playbook Tutorial | How To Install LAMP Stack On Centos Using Ansible



```
[root@ansible1 ansibleplaybook]# vi index.php  
[root@ansible1 ansibleplaybook]# ls  
index.php  lampstack.yml  
[root@ansible1 ansibleplaybook]# ansible-playbook lampstack.yml
```

```
PLAY [testservers] *****  
  
TASK [Gathering Facts] *****  
ok: [192.168.145.146]  
  
TASK [Installing Apache2] *****  
changed: [192.168.145.146]  
  
TASK [Enabling and starting the apache service] *****  
changed: [192.168.145.146]  
  
TASK [Installing MYSQL Community Server] *****  
changed: [192.168.145.146]  
  
TASK [Starting the MySQL DB service] *****  
changed: [192.168.145.146]  
  
TASK [Installing PHP] *****  
changed: [192.168.145.146]  
  
TASK [Copying the PHP test.php file] *****  
changed: [192.168.145.146]  
  
PLAY RECAP *****  
192.168.145.146 : ok=7    changed=6    unreachable=0    failed=0    skipped=0    rescued=0    ignored=0
```

[root@ansible1 ansibleplaybook]#

Scroll for details



```
[root@localhost ~]# ip addr
1: lo: <LOOPBACK,UP,LOWER_UP> mtu 65536 qdisc noqueue state UNKNOWN group default qlen 1000
    link/loopback 00:00:00:00:00:00 brd 00:00:00:00:00:00
        inet 127.0.0.1/8 scope host lo
            valid_lft forever preferred_lft forever
        inet6 ::1/128 scope host
            valid_lft forever preferred_lft forever
2: ens33: <BROADCAST,MULTICAST,UP,LOWER_UP> mtu 1500 qdisc pfifo_fast state UP group default qlen 1000
    link/ether 00:0c:29:ee:1c:fe brd ff:ff:ff:ff:ff:ff
        inet 192.168.145.146/24 brd 192.168.145.255 scope global noprefixroute dynamic ens33
            valid_lft 1702sec preferred_lft 1702sec
        inet6 fe80::1245:15f8:cae4:d2e/64 scope link noprefixroute
            valid_lft forever preferred_lft forever
[root@localhost ~]#
```

Ansible LAMP Stack Playbook Tutorial | How To Install LAMP Stack On Centos Using Ansible

```
[root@localhost ~]# php -v
PHP 5.4.16 (cli) (built: Apr 1 2020 04:07:17)
Copyright (c) 1997-2013 The PHP Group
Zend Engine v2.4.0, Copyright (c) 1998-2013 Zend Technologies
[root@localhost ~]#
```

14:01 / 17:03

Scroll for details

Ansible LAMP Stack Playbook Tutorial | How To Install LAMP Stack On Centos Using Ansible



```
[root@localhost ~]# php -v
PHP 5.4.16 (cli) (built: Apr 1 2020 04:07:17)
Copyright (c) 1997-2013 The PHP Group
Zend Engine v2.4.0, Copyright (c) 1998-2013 Zend Technologies
[root@localhost ~]# systemctl status httpd
● httpd.service - The Apache HTTP Server
   Loaded: loaded (/usr/lib/systemd/system/httpd.service; enabled; vendor preset: disabled)
   Active: active (running) since Sun 2020-11-08 12:40:14 EST; 5min ago
     Docs: man:httpd(8)
           man:apachectl(8)
   Main PID: 7625 (httpd)
      Status: "Total requests: 0; Current requests/sec: 0; Current traffic: 0 B/sec"
   CGroup: /system.slice/httpd.service
           └─7625 /usr/sbin/httpd -DFOREGROUND
              ├─7626 /usr/sbin/httpd -DFOREGROUND
              ├─7627 /usr/sbin/httpd -DFOREGROUND
              ├─7628 /usr/sbin/httpd -DFOREGROUND
              ├─7629 /usr/sbin/httpd -DFOREGROUND
              └─7631 /usr/sbin/httpd -DFOREGROUND

Nov 08 12:40:13 localhost.localdomain systemd[1]: Starting The Apache HTTP Server...
Nov 08 12:40:14 localhost.localdomain httpd[7625]: AH000558: httpd: Could not reliably determine the server's fully...ssage
Nov 08 12:40:14 localhost.localdomain systemd[1]: Started The Apache HTTP Server.
Hint: Some lines were ellipsized, use -l to show in full.
[root@localhost ~]#
```

Ansible LAMP Stack Playbook Tutorial | How To Install LAMP Stack On Centos Using Ansible



```
[root@localhost ~]# php -v
PHP 5.4.16 (cli) (built: Apr 1 2020 04:07:17)
Copyright (c) 1997-2013 The PHP Group
Zend Engine v2.4.0, Copyright (c) 1998-2013 Zend Technologies
[root@localhost ~]# systemctl status httpd
● httpd.service - The Apache HTTP Server
   Loaded: loaded (/usr/lib/systemd/system/httpd.service; enabled; vendor preset: disabled)
   Active: active (running) since Sun 2020-11-08 12:40:14 EST; 5min ago
     Docs: man:httpd(8)
           man:apachectl(8)
Main PID: 7625 (httpd)
   Status: "Total requests: 0; Current requests/sec: 0; Current traffic: 0 B/sec"
   CGroup: /system.slice/httpd.service
           └─7625 /usr/sbin/httpd -DFOREGROUND
               ├─7626 /usr/sbin/httpd -DFOREGROUND
               ├─7627 /usr/sbin/httpd -DFOREGROUND
               ├─7628 /usr/sbin/httpd -DFOREGROUND
               ├─7629 /usr/sbin/httpd -DFOREGROUND
               └─7631 /usr/sbin/httpd -DFOREGROUND

Nov 08 12:40:13 localhost.localdomain systemd[1]: Starting The Apache HTTP Server...
Nov 08 12:40:14 localhost.localdomain httpd[7625]: AH00558: httpd: Could not reliably determine the server's fully...ssage
Nov 08 12:40:14 localhost.localdomain systemd[1]: Started The Apache HTTP Server.
Hint: Some lines were ellipsized, use -l to show in full.
[root@localhost ~]# systemctl status mysql
```

Ansible LAMP Stack Playbook Tutorial | How To Install LAMP Stack On Centos Using Ansible



```
[root@localhost ~]# php -v
PHP 5.4.16 (cli) (built: Apr 1 2020 04:07:17)
Copyright (c) 1997-2013 The PHP Group
Zend Engine v2.4.0, Copyright (c) 1998-2013 Zend Technologies
[root@localhost ~]# systemctl status httpd
● httpd.service - The Apache HTTP Server
   Loaded: loaded (/usr/lib/systemd/system/httpd.service; enabled; vendor preset: disabled)
   Active: active (running) since Sun 2020-11-08 12:40:14 EST; 5min ago
     Docs: man:httpd(8)
           man:apachectl(8)
  Main PID: 7625 (httpd)
    Status: "Total requests: 0; Current requests/sec: 0; Current traffic: 0 B/sec"
   CGroup: /system.slice/httpd.service
           ├─7625 /usr/sbin/httpd -DFOREGROUND
           ├─7626 /usr/sbin/httpd -DFOREGROUND
           ├─7627 /usr/sbin/httpd -DFOREGROUND
           ├─7628 /usr/sbin/httpd -DFOREGROUND
           ├─7629 /usr/sbin/httpd -DFOREGROUND
           └─7631 /usr/sbin/httpd -DFOREGROUND

Nov 08 12:40:13 localhost.localdomain systemd[1]: Starting The Apache HTTP Server...
Nov 08 12:40:14 localhost.localdomain httpd[7625]: AH00558: httpd: Could not reliably determine the server's fully...ssage
Nov 08 12:40:14 localhost.localdomain systemd[1]: Started The Apache HTTP Server.
Hint: Some lines were ellipsized, use -l to show in full.
[root@localhost ~]# systemctl status mysqld
```

Ansible LAMP Stack Playbook Tutorial | How To Install LAMP Stack On Centos Using Ansible



```
Docs: man:htpd(8)
man:apachectl(8)
Main PID: 7625 (httpd)
Status: "Total requests: 0; Current requests/sec: 0; Current traffic: 0 B/sec"
CGroup: /system.slice/httpd.service
└─7625 /usr/sbin/httpd -DFOREGROUND
   ├─7626 /usr/sbin/httpd -DFOREGROUND
   ├─7627 /usr/sbin/httpd -DFOREGROUND
   ├─7628 /usr/sbin/httpd -DFOREGROUND
   ├─7629 /usr/sbin/httpd -DFOREGROUND
   └─7631 /usr/sbin/httpd -DFOREGROUND

Nov 08 12:40:13 localhost.localdomain systemd[1]: Starting The Apache HTTP Server...
Nov 08 12:40:14 localhost.localdomain httpd[7625]: AH00558: httpd: Could not reliably determine the server's fully...ssage
Nov 08 12:40:14 localhost.localdomain systemd[1]: Started The Apache HTTP Server.
Hint: Some lines were ellipsized, use -l to show in full.
[root@localhost ~]# systemctl status mysqld
● mysqld.service - MySQL Server
   Loaded: loaded (/usr/lib/systemd/system/mysqld.service; enabled; vendor preset: disabled)
   Active: active (running) since Sun 2020-11-08 12:45:04 EST; 1min 18s ago
     Docs: man:mysqld(8)
           http://dev.mysql.com/doc/refman/en/using-systemd.html
   Process: 7781 ExecStart=/usr/sbin/mysqld --daemonize --pid-file=/var/run/mysqld/mysqld.pid $MYSQLD_OPTS (code=exited, st
   Main PID: 7784 (mysqld)
      CGroup: /system.slice/mysqld.service
              └─7784 /usr/sbin/mysqld --daemonize --pid-file=/var/run/mysqld/mysqld.pid

Nov 08 12:45:02 localhost.localdomain systemd[1]: Starting MySQL Server...
Nov 08 12:45:03 localhost.localdomain mysqld[7784]: 2020-11-08 12:45:03 [Note] /usr/sbin/mysqld (mysqld 8.0.23) starting as process 7784 ...
Nov 08 12:45:03 localhost.localdomain mysqld[7784]: 2020-11-08 12:45:03 [Warning] InnoDB: Using log buffer size 16MB is smaller than recommended for InnoDB tables.
Nov 08 12:45:03 localhost.localdomain mysqld[7784]: 2020-11-08 12:45:03 [Warning] InnoDB: You can set it to at least 25% of total data + index size or 256MB, whichever is larger.
Nov 08 12:45:03 localhost.localdomain mysqld[7784]: 2020-11-08 12:45:03 [Warning] InnoDB: Using log buffer size 16MB is smaller than recommended for InnoDB tables.
Nov 08 12:45:03 localhost.localdomain mysqld[7784]: 2020-11-08 12:45:03 [Warning] InnoDB: You can set it to at least 25% of total data + index size or 256MB, whichever is larger.
```

[root@localhost ~]# 14:20 / 17:03

Scroll for details



Ansible LAMP Stack Playbook Tutorial | How To Install LAMP Stack On Centos Using Ansible



```
[root@localhost ~]# cd /var/www/html/  
[root@localhost html]# ls  
index.php  
[root@localhost html]#
```

◀ ▶ ▶| 🔍 14:28 / 17:03

Scroll for details



Ansible LAMP Stack Playbook Tutorial | How To Install LAMP Stack On Centos Using Ansible



```
[root@localhost ~]# cd /var/www/html/
[root@localhost html]# ls
index.php
[root@localhost html]# cat index.php

<html>
<head>
<title>Ansible Application</title>
</head>
<body>

<?php
echo " Hello From Thetips4you"
?>

</body>
</html>
[root@localhost html]#
```

Testing 123..

This page is used to test the proper operation of the [Apache HTTP server](#) after it has been installed. If you can read this page it means that this site is working properly. This server is powered by [CentOS](#).

Just visiting?

The website you just visited is either experiencing problems or is undergoing routine maintenance.

If you would like to let the administrators of this website know that you've seen this page instead of the page you expected, you should send them e-mail. In general, mail sent to the name "webmaster" and directed to the website's domain should reach the responsible person.

◀ ▶ ⏪ ⏩ 🔍 14:48 / 17:03

Are you the Administrator?

You should add your website content to the directory `/var/www/html/`.

To prevent this page from ever being used, follow the instructions in the file `/etc/httpd/conf.d/welcome.conf`.

Promoting Apache and CentOS

YOU ARE FREE TO USE THE IMAGES BELOW ON APACHE AND CENTOS' LAMP-POWERED

WEBSITES. THANKS FOR USING APACHE AND CENTOS!



Ansible LAMP Stack Playbook Tutorial | How To Install LAMP Stack On Centos Using Ansible

Not secure | 192.168.149.146/index.php

Hello From Thetps4you

◀ ▶ ⏪ ⏩ 🔍 14:53 / 17:03

Scroll for details

CC ⚙️ 📺

Ansible LAMP Stack Playbook Tutorial | How To Install LAMP Stack On Centos Using Ansible



```
[root@localhost html]# cat index.php\n\n<html>\n  <head>\n    <title>Ansible Application</title>\n  </head>\n  <body>\n    <?php\n      echo \" Hello From Thetips4you\";\n    ?>\n  </body>\n</html>\n[root@localhost html]#
```



Test your knowledge!

Q) What this Ansible Ad Hoc command do ?
`ansible testservers -a "/sbin/reboot"`

- A) Open Command Prompt
- B) Create folder reboot
- C) Reboot the testservers collection
- D) Install ansible and reboot

Test your knowledge!

Q) What this Ansible Ad Hoc command do ?
`ansible testservers -a "/sbin/reboot"`

- A) Open Command Prompt
- B) Create folder reboot
- C) Reboot the testservers collection
- D) Install ansible and reboot

Enter your answer in Comment section

Ansible LAMP Stack Playbook Tutorial | How To Install LAMP Stack On Centos Using Ansible

Ansible LAMP Stack Playbook Tutorial | How To Install LAMP Stack On Centos Using Ansible



FOLDERS

demoapp

```
10    name: httpd
11      state: installed
12
13      - name: Enabling httpd service and start the service
14        services:
15          - name: httpd
16            enabled: yes
17            state: started
18
19      - name: Install MySQL DB Community edition
20        yum:
21          name:
22            - mariadb-server
23            - mariadb
24          state: latest
25
26      - name: Start the MySQL service
27        services:
28          - name: mysqld
29            enabled: yes
30            state: started
31
32
33      - name: Installing PHP
34        yum:
35          name:
36            - php
37            - php-mysql
38          state: latest
39
40      - name: Copying your php file in the /var/www/html
41        copy:
42          src: /root/ansibleplaybook/Index.php
43          dest: /var/www/html/index.php
44
45
```

≡ How to create simple ansible playbook to setup lamp stack on centos | Ansible lamp stack playb... 🔍

```
[root@ansible1 ansibleplaybook]# ansible-playbook howtostack.yml

PLAY [testservers] *****

TASK [Gathering Facts] *****
ok: [192.168.145.146]

TASK [Installing Apache, MySQL and PHP] *****
changed: [192.168.145.146]

TASK [Enabling and Starting the service httpd] *****
changed: [192.168.145.146]

TASK [Enabling and Starting the service MySQL] *****
changed: [192.168.145.146]

TASK [Copying the php file] *****
ok: [192.168.145.146]

PLAY RECAP *****
192.168.145.146 : ok=5    changed=3    unreachable=0    failed=0    skipped=0    rescued=0    ignored=0
```

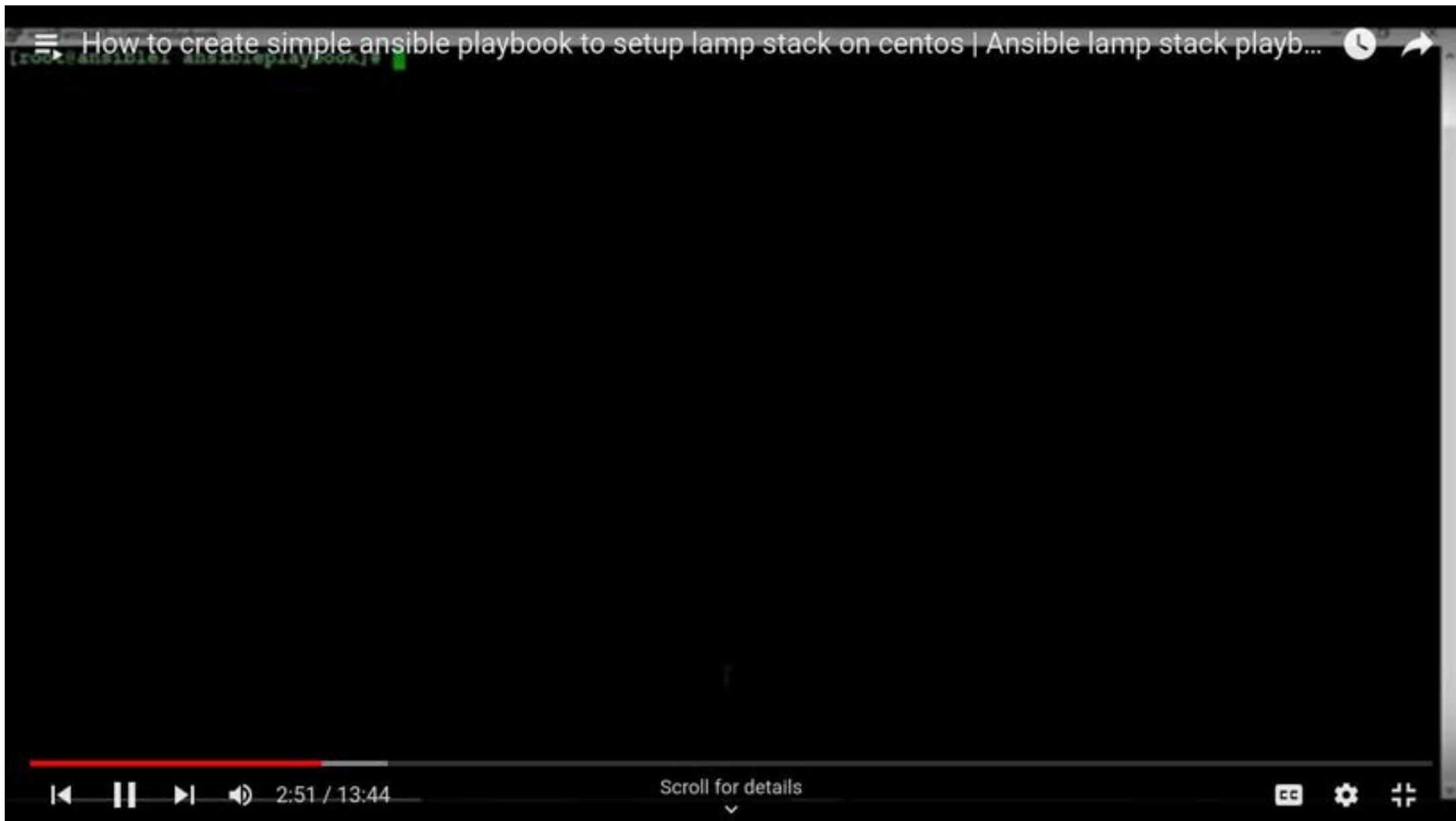
[root@ansible1 ansibleplaybook]#

Hello Friends, Welcome back to my channel

An Easy and simple way to setup Lamp stack using Ansible Playbook.

What is lamp stack ?





How to create simple ansible playbook to setup lamp stack on centos | Ansible lamp stack playb...

```
FOLDERS: newlampstack.yml  
1 ...  
2 - hosts: testservers
```

≡ How to create simple ansible playbook to setup lamp stack on centos | Ansible lamp stack playb... 🔍

```
[root@ansible1 ansible]# cd /etc/ansible/
[root@ansible1 ansible]# ls
ansible.cfg  hosts  roles
[root@ansible1 ansible]#
```

◀ ▶ ⏪ ⏩ 2:57 / 13:44

Scroll for details

CC ⚙ ⌂

≡ How to create simple ansible playbook to setup lamp stack on centos | Ansible lamp stack playb... 🔍

```
[root@ansible1 ansible]# cd /etc/ansible/
[root@ansible1 ansible]# ls
ansible.cfg  hosts  roles
[root@ansible1 ansible]# cat hosts
```

3:00 / 13:44

Scroll for details

CC

≡ How to create simple ansible playbook to setup lamp stack on centos | Ansible lamp stack playb...



```
# Ex 1: Ungrouped hosts, specify before any group headers.

## green.example.com
## blue.example.com
## crvers]
## alpha.example.org

# Ex 2: A collection of hosts belonging to the 'webservers' group

[testservers]
192.168.145.146

## [webservers]
## alpha.example.org
## beta.example.org
## 192.168.1.100
## 192.168.1.110

# If you have multiple hosts following a pattern you can specify
# them like this:

## www[001:006].example.com

# Ex 3: A collection of database servers in the 'dbservers' group

## [dbservers]
## 0.0.1 internet.mydomain.net
## 0.0.2.1 intranet.mydomain.net
```

Scroll for details



How to create simple ansible playbook to setup lamp stack on centos | Ansible lamp stack playb...

```
1  ---
2  - hosts: testservers
3    remote_user: root
4    become: yes
5
6  tasks:
7    - name: Installing Apache, MySQL and PHP
8      yum:
9        name:
10       - httpd
11       - mysql-community-server
12       - mysql-community-client
13       - php
14       - php-mysql
15     state: installed
16
17    - name: Enabling and Starting the service httpd
18      service:
19        name: httpd
20        enabled: yes
21        state: started
22
23    - name: Enabling and Starting the service MySQL
24      service:
25        name: mysqld
```

How to create simple ansible playbook to setup lamp stack on centos | Ansible lamp stack playb...

```
1  #!/usr/bin/ansible-playbook -i hosts testlampstack.yml
2
3  - hosts: testservers
4    remote_user: root
5    become: yes
6
7    tasks:
8      - name: Installing Apache, MySQL and PHP
9        yum:
10          name:
11            - httpd
12            - mysql-community-server
13            - mysql-community-client
14            - php
15            - php-mysql
16          state: installed
17
18      - name: Enabling and Starting the service httpd
19        service:
20          name: httpd
21          enabled: yes
22          state: started
23
24      - name: Enabling and Starting the service MySQL
25        service:
26          name: mysqld
27          enabled: yes
28          state: started
```

How to create simple ansible playbook to setup lamp stack on centos | Ansible lamp stack playb...

```
6 tasks:
7   - name: Installing Apache, MySQL and PHP
8     yum:
9       name:
10      - httpd
11      - mysql-community-server
12      - mysql-community-client
13      - php
14      - php-mysql
15      state: installed
16
17   - name: Enabling and Starting the service httpd
18     service:
19       name: httpd
20       enabled: yes
21       state: started
22
23   - name: Enabling and Starting the service MySQL
24     service:
25       name: mysqld
26       enabled: yes
27       state: started
28
29   - name: Copying the php file
30     copy:
31       src:
```

≡ How to create simple ansible playbook to setup lamp stack on centos | Ansible lamp stack playb...

```
[root@ansible1 ansible]# cd ~  
[root@ansible1 ~]# cd ansibleplaybook/  
[root@ansible1 ansibleplaybook]# ls  
index.php  lampstack.yml  removelampstack.yml  
[root@ansible1 ansibleplaybook]# █
```

How to create simple ansible playbook to setup lamp stack on centos | Ansible lamp stack playb...

```
9      name:
10     - httpd
11     - mysql-community-server
12     - mysql-community-client
13     - php
14     - php-mysql
15     state: installed
16
17   - name: Enabling and Starting the service httpd
18     service:
19       name: httpd
20       enabled: yes
21       state: started
22
23   - name: Enabling and Starting the service MySQL
24     service:
25       name: mysqld
26       enabled: yes
27       state: started
28
29   - name: Copying the php file
30     copy:
31       src: /root/ansibleplaybook/index.php
32       dest: /var/www/html/index.php
33       force: yes
34
35
```

How to create simple ansible playbook to setup lamp stack on centos | Ansible lamp stack playb...

```
FOLDERS  newlampstack.yml  *
1 ...
2 - hosts: testservers
3   remote_user: root
4   become: yes
5
6   tasks:
7     - name: Installing Apache, MySQL and PHP
8       yum:
9         name:
10        - httpd
11        - mysql-community-server
12        - mysql-community-client
13        - php
14        - php-mysql
15       state: installed
16
17     - name: Enabling and Starting the service httpd
18       service:
19         name: httpd
20         enabled: yes
21         state: started
22
23     - name: Enabling and Starting the service MySQL
24       service:
25         name: mysqld
26         enabled: yes
27         state: started
```

≡ How to create simple ansible playbook to setup lamp stack on centos | Ansible lamp stack playb...

```
[root@ansible ~]# cd ansibleplaybook/  
[root@ansible ansibleplaybook]# ls  
index.php lampstack.yml removelampstack.yml  
[root@ansible ansibleplaybook]# vi newlampstack.yml
```

How to create simple ansible playbook to setup lamp stack on centos | Ansible lamp stack play...

```
  pip-mysql
    state: installed

- name: Enabling and Starting the service httpd
  service:
    name: httpd
    enabled: yes
    state: started

- name: Enabling and Starting the service MySQL
  service:
    name: mysqld
    enabled: yes
    state: started

- name: Copying the php file
  copy:
    src: /root/ansibleplaybook/index.php
    dest: /var/www/html/index.php
    force: yes
```

How to create simple ansible playbook to setup lamp stack on centos | Ansible lamp stack play...

```
---
```

```
- hosts: testservers
  remote_user: root
  become: yes

  tasks:
    - name: Installing Apache, MySQL and PHP
      yum:
        name:
          - httpd
          - mysql-community-server
          - mysql-community-client
          - php
          - php-mysql
        state: installed

    - name: Enabling and Starting the service httpd
      service:
        name: httpd
        enabled: yes
        state: started

    - name: Enabling and Starting the service MySQL
      service:
        name: mysqld
        enabled: yes
        state: started

    - name: Copying the php file
```

≡ How to create simple ansible playbook to setup lamp stack on centos | Ansible lamp stack playb... ⏱ ↗

```
[root@localhost ~]# systemctl status httpd
Unit httpd.service could not be found.
[root@localhost ~]#
```

◀ ▶ ⏪ ⏩ 10:57 / 13:44

Scroll for details

CC ⚙ ⌂

≡ How to create simple ansible playbook to setup lamp stack on centos | Ansible lamp stack playb...



```
[root@localhost ~]# systemctl status httpd  
Unit httpd.service could not be found.  
[root@localhost ~]# systemctl status mysqld  
Unit mysqld.service could not be found.  
[root@localhost ~]#
```

How to create simple ansible playbook to setup lamp stack on centos | Ansible lamp stack playb...

◀ ▶ ▶| 🔍 11:14 / 13:44

Scroll for details

CC ⚙️ 🎫

```
≡ How to create simple ansible playbook to setup lamp stack on centos | Ansible lamp stack playb... ⏪ ↗  
[root@ansible1 ansibleplaybook]# ansible-playbook newlampstack.yaml  
  
PLAY [testservers] *****  
  
TASK [Gathering Facts] *****  
ok: [192.168.145.146]  
  
TASK [Installing Apache, MySQL and PHP] *****  
changed: [192.168.145.146]  
  
TASK [Enabling and Starting the service httpd] *****  
changed: [192.168.145.146]  
  
TASK [Enabling and Starting the service MySQL] *****  
changed: [192.168.145.146]  
  
TASK [Copying the php file] *****  
ok: [192.168.145.146]  
  
PLAY RECAP *****  
192.168.145.146 : ok=5    changed=3    unreachable=0    failed=0    skipped=0    rescued=0    ignored=0  
  
[root@ansible1 ansibleplaybook]#
```

≡ How to create simple ansible playbook to setup lamp stack on centos | Ansible lamp stack play...



```
[root@localhost ~]# systemctl status httpd
Unit httpd.service could not be found.
[root@localhost ~]# systemctl status mysqld
Unit mysqld.service could not be found.
[root@localhost ~]# systemctl status httpd
● httpd.service - The Apache HTTP Server
   Loaded: loaded (/usr/lib/systemd/system/httpd.service; enabled; vendor preset: disabled)
   Active: active (running) since Thu 2021-01-14 13:18:26 EST; 21s ago
     Docs: man:httpd(8)
           man:apachectl(8)
 Main PID: 7684 (httpd)
    Status: "Total requests: 0; Current requests/sec: 0; Current traffic: 0 B/sec"
   CGrou...
```

```
└─7684 /usr/sbin/httpd -DFOREGROUND
└─7685 /usr/sbin/httpd -DFOREGROUND
└─7686 /usr/sbin/httpd -DFOREGROUND
└─7687 /usr/sbin/httpd -DFOREGROUND
└─7688 /usr/sbin/httpd -DFOREGROUND
└─7689 /usr/sbin/httpd -DFOREGROUND

Jan 14 13:18:25 localhost.localdomain systemd[1]: Starting The Apache HTTP Server...
Jan 14 13:18:26 localhost.localdomain httpd[7684]: AH00558: httpd: Could not reliably determine the server's fully...ssage
Jan 14 13:18:26 localhost.localdomain systemd[1]: Started The Apache HTTP Server.
Hint: Some lines were ellipsized, use -l to show in full.
[root@localhost ~]#
```

≡ How to create simple ansible playbook to setup lamp stack on centos | Ansible lamp stack play...



```
[root@localhost ~]# systemctl status httpd
Unit httpd.service could not be found.
[root@localhost ~]# systemctl status mysqld
Unit mysqld.service could not be found.
[root@localhost ~]# systemctl status httpd
● httpd.service - The Apache HTTP Server
   Loaded: loaded (/usr/lib/systemd/system/httpd.service; enabled; vendor preset: disabled)
   Active: active (running) since Thu 2021-01-14 13:18:26 EST; 21s ago
     Docs: man:httpd(8)
           man:apachectl(8)
Main PID: 7684 (httpd)
   Status: "Total requests: 0; Current requests/sec: 0; Current traffic: 0 B/sec"
   CGrou.../system.slice/httpd.service
      └─7684 /usr/sbin/httpd -DFOREGROUND
         ├─7685 /usr/sbin/httpd -DFOREGROUND
         ├─7686 /usr/sbin/httpd -DFOREGROUND
         ├─7687 /usr/sbin/httpd -DFOREGROUND
         ├─7688 /usr/sbin/httpd -DFOREGROUND
         └─7689 /usr/sbin/httpd -DFOREGROUND
```

```
Jan 14 13:18:25 localhost.localdomain systemd[1]: Starting The Apache HTTP Server...
Jan 14 13:18:26 localhost.localdomain httpd[7684]: AH00558: httpd: Could not reliably determine the server's fully...ssage
Jan 14 13:18:26 localhost.localdomain systemd[1]: Started The Apache HTTP Server.
Hint: Some lines were ellipsized, use -l to show in full.
[root@localhost ~]# systemctl status mysqld
```

≡ How to create simple ansible playbook to setup lamp stack on centos | Ansible lamp stack play...



```
Docs: man:apachectl(8)
man:apachectl(8)
Main PID: 7684 (httpd)
Status: "Total requests: 0; Current requests/sec: 0; Current traffic: 0 B/sec"
CGroup: /system.slice/httpd.service
└─7684 /usr/sbin/httpd -DFOREGROUND
   ├─7685 /usr/sbin/httpd -DFOREGROUND
   ├─7686 /usr/sbin/httpd -DFOREGROUND
   ├─7687 /usr/sbin/httpd -DFOREGROUND
   ├─7688 /usr/sbin/httpd -DFOREGROUND
   └─7689 /usr/sbin/httpd -DFOREGROUND

Jan 14 13:18:25 localhost.localdomain systemd[1]: Starting The Apache HTTP Server...
Jan 14 13:18:26 localhost.localdomain httpd[7684]: AH00558: httpd: Could not reliably determine the server's fully...ssage
Jan 14 13:18:26 localhost.localdomain systemd[1]: Started The Apache HTTP Server.
Hint: Some lines were ellipsized, use -l to show in full.
[root@localhost ~]# systemctl status mysqld
● mysqld.service - MySQL Server
   Loaded: loaded (/usr/lib/systemd/system/mysqld.service; enabled; vendor preset: disabled)
   Active: active (running) since Thu 2021-01-14 13:18:31 EST; 22s ago
     Docs: man:mysqld(8)
           http://dev.mysql.com/doc/refman/en/using-systemd.html
   Process: 7737 ExecStartPre=/usr/bin/mysqld_pre_systemd (code=exited, status=0/SUCCESS)
 Main PID: 7760 (mysqld)
   Status: "Server is operational"
  CGroup: /system.slice/mysqld.service
          └─7760 /usr/sbin/mysqld

Jan 14 13:18:27 localhost.localdomain systemd[1]: Starting MySQL Server...
Jan 14 13:18:31 localhost.localdomain systemd[1]: Started MySQL Server.
[r...]
```

≡ How to create simple ansible playbook to setup lamp stack on centos | Ansible lamp stack playb... ⏴ ↗

```
[root@localhost ~]# cd /var/www/html/
[root@localhost html]# ls
index.php
[root@localhost html]# cat index.php

```



◀ ▶ ▶| 🔍 12:14 / 13:44



Scroll for details



≡ How to create simple ansible playbook to setup lamp stack on centos | Ansible lamp stack playb...



```
[root@localhost ~]# cd /var/www/html/
[root@localhost html]# ls
index.php
[root@localhost html]# cat index.php

<html>
<head>
<title>Ansible Application</title>
</head>
<body>

    <?php
        echo " Hello From Thetips4you, this tutorial is part of handlers learning in ansible"
    ?>

</body>
</html>
[root@localhost html]#
```

≡ How to create simple ansible playbook to setup lamp stack on centos | Ansible lamp stack playb...



```
[root@ansible1 ansibleplaybook]# ansible-playbook newlampstack.yaml

PLAY [testservers] *****

TASK [Gathering Facts] *****
ok: [192.168.145.146]

TASK [Installing Apache, MySQL and PHP] *****
changed: [192.168.145.146]

TASK [Enabling and Starting the service httpd] *****
changed: [192.168.145.146]

TASK [Enabling and Starting the service MySQL] *****
changed: [192.168.145.146]

TASK [Copying the php file] *****
ok: [192.168.145.146]

PLAY RECAP *****
192.168.145.146      : ok=5    changed=3    unreachable=0    failed=0    skipped=0    rescued=0    ignored=0

[root@ansible1 ansibleplaybook]#
```

How to create simple ansible playbook to setup lamp stack on centos | Ansible lamp stack playb...

```
1  ---
2  - hosts: testservers
3    remote_user: root
4    become: yes
5
6    tasks:
7      - name: Installing Apache, MySQL and PHP
8        yum:
9          packages:
10            - httpd
11            - mysql-community-server
12            - mysql-community-client
13            - php
14            - php-mysql
15          state: installed
16
17      - name: Enabling and Starting the service httpd
18        service:
19          name: httpd
20          enabled: yes
21          state: started
22
23      - name: Enabling and Starting the service MySQL
24        service:
25          name: mysqld
26          enabled: yes
27          state: started
28
29      - name: Copying the php file
30        copy:
31          src: /root/ansibleplaybook/index.php
32          dest: /var/www/html/index.php
```

How to create simple ansible playbook to setup lamp stack on centos | Ansible lamp stack playb...

```
File: newlampstack.yml
10      - httpd
11      - mysql-community-server
12      - mysql-community-client
13      - php
14      - php-mysql
15      state: installed
16
17      - name: Enabling and Starting the service httpd
18        service:
19          name: httpd
20          enabled: yes
21          state: started
22
23      - name: Enabling and Starting the service MySQL
24        service:
25          name: mysqld
26          enabled: yes
27          state: started
28
29      - name: Copying the php file
30        copy:
31          src: /root/ansibleplaybook/index.php
32          dest: /var/www/html/index.php
33          force: yes
34
35
36
37
38
39
40
41
```

Handlers in ansible playbook.

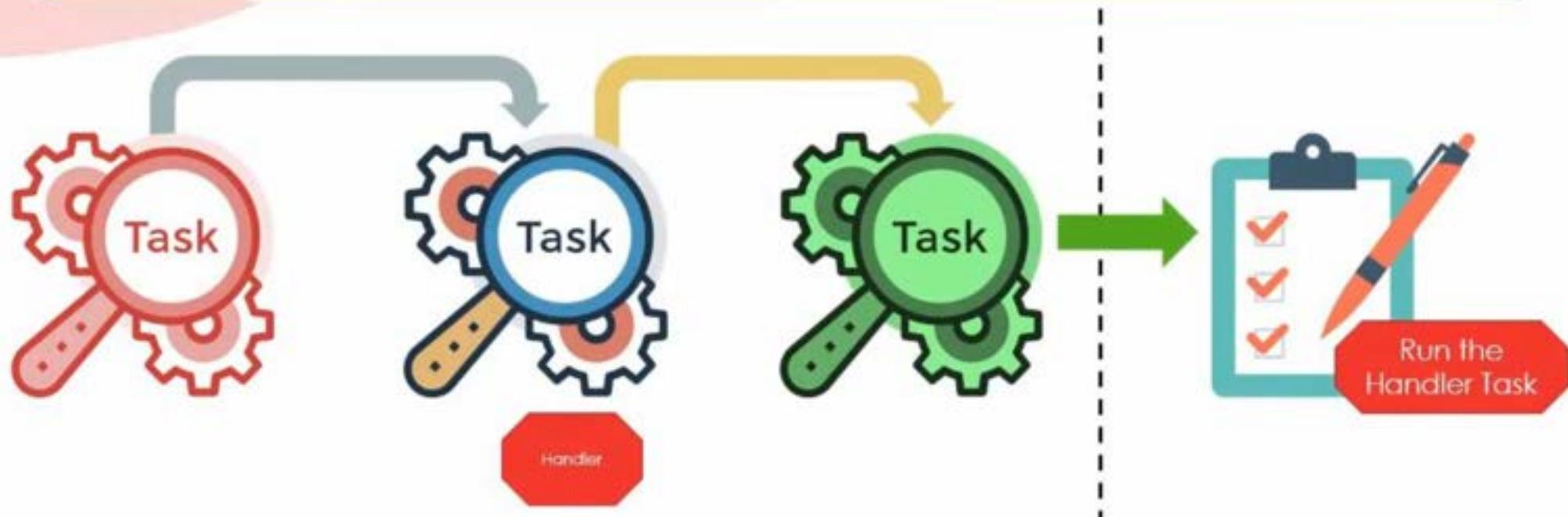


- ✓ Handlers run after all the tasks in a particular play have been completed. Handler only runs once, regardless of how many tasks notify it.

Handlers in ansible playbook.

- ✓ **Handlers are tasks that only run when notified.**
- ✓ **Handlers are used in cases when you want a task to run only when a change is made on a machine.**

Handlers in ansible playbook.



- ✓ Handlers run after all the tasks in a particular play have been completed. Handler only runs once, regardless of how many tasks notify it.

Handlers in ansible playbook.



- ✓ Handlers run after all the tasks in a particular play have been completed. Handler only runs once, regardless of how many tasks notify it.



Ansible Handlers Tutorial | Ansible Notify And Handler



```
- hosts: testservers
  remote_user: root

  tasks:
    - name: Installing Apache, MySQL DB Community edition & PHP
      yum:
        name:
          - httpd
          - mysql-community-server
          - mysql-community-client
          - php
          - php-mysql
        state: installed

    - name: Enabling httpd service and start the service
      service:
        name: httpd
        enabled: yes
        state: started

    - name: Enabling mysql service and start the service
      service:
        name: mysqld
        enabled: yes
        state: started

    - name: Copying your php file in the /var/www/html
      copy:
        src: /root/ansibleplaybook/index.php
        dest: /var/www/html/index.php
        force: yes
```

```
- hosts: t
  remote_user: root

  tasks:
    - name: Installing Apache, MySQL DB Community edition & PHP
      yum:
        name:
          - httpd
          - mysql-community-server
          - mysql-community-client
          - php
          - php-mysql
        state: installed

    - name: Enabling httpd service and start the service
      service:
        name: httpd
        enabled: yes
        state: started

    - name: Enabling mysql service and start the service
      service:
        name: mysqld
        enabled: yes
        state: started

    - name: Copying your php file in the /var/www/html
      copy:
        src: /root/ansibleplaybook/index.php
        dest: /var/www/html/index.php
        force: yes
```

Ansible Handlers Tutorial | Ansible Notify And Handler



```
- hosts: testservers
  remote_user: root

  tasks:
    - name: Installing Apache, MySQL DB Community edition & PHP
      yum:
        name:
          - httpd
          - mysql-community-server
          - mysql-community-client
          - php
          - php-mysql
        state: installed

    - name: Enabling httpd service and start the service
      service:
        name: httpd
        enabled: yes
        state: started

    - name: Enabling mysql service and start the service
      service:
        name: mysqld
        enabled: yes
        state: started

    - name: Copying your php file in the /var/www/html
      copy:
        src: /root/ansibleplaybook/index.php
        dest: /var/www/html/index.php
        force: yes

    notify:
      - restart httpd service

  handlers:
    - name: restart httpd service
      service:
        name: httpd
        state: restarted
```

Ansible Handlers Tutorial | Ansible Notify And Handler



```
- hosts: testservers
  remote_user: root

  tasks:
    - name: Installing Apache, MySQL DB Community edition & PHP
      yum:
        name:
          - httpd
          - mysql-community-server
          - mysql-community-client
          - php
          - php-mysql
        state: installed

    - name: Enabling httpd service and start the service
      service:
        name: httpd
        enabled: yes
        state: started

    - name: Enabling mysql service and start the service
      service:
        name: mysqld
        enabled: yes
        state: started

    - name: Copying your php file in the /var/www/html
      copy:
        src: /root/ansibleplaybook/index.php
        dest: /var/www/html/index.php
        force: yes

    notify:
      - restart httpd service

  handlers:
    - name: restart httpd service
      service:
        name: httpd
        state: restarted
```

Ansible Handlers Tutorial | Ansible Notify And Handler
[root@ansible1 ansibleplaybook]# cat handlers.yml



I

Ansible Handlers Tutorial | Ansible Notify And Handler

```
- mysql-community-client
  - php
  - php-mysql
  state: installed

- name: Enabling httpd service and start the service
  service:
    name: httpd
    enabled: yes
    state: started

- name: Enabling mysql service and start the service
  service:
    name: mysqld
    enabled: yes
    state: started

- name: Copying your php file in the /var/www/html
  copy:
    src: /root/ansibleplaybook/index.php
    dest: /var/www/html/index.php
    force: yes
  notify:
    - restart httpd service

handlers:
```

```
- name: restart httpd service
  service:
    name: httpd
    state: restarted
```

```
[root@ansibile ansible]#
```

Scroll for details

Ansible Handlers Tutorial | Ansible Notify And Handler



```
[root@ansible1 ansibleplaybook]# clear  
[root@ansible1 ansibleplaybook]# cat handlers.yml
```

```
---
```

```
- hosts: testservers
  remote_user: root
```



```
tasks:
```

```
  - name: Installing Apache, MYSQL DB Community edition & PHP
    yum:
      name:
        - httpd
        - mysql-community-server
        - mysql-community-client
        - php
        - php-mysql
      state: installed
```

```
  - name: Enabling httpd service and start the service
    service:
      name: httpd
      enabled: yes
      state: started
```

```
  - name: Enabling mysql service and start the service
    service:
      name: mysqld
      enabled: yes
      state: started
```

Ansible Handlers Tutorial | Ansible Notify And Handler

```
- mysql-community-client
  - php
  - php-mysql
  state: installed

- name: Enabling httpd service and start the service
  service:
    name: httpd
    enabled: yes
    state: started

- name: Enabling mysql service and start the service
  service:
    name: mysqld
    enabled: yes
    state: started

- name: Copying your php file in the /var/www/html
  copy:
    src: /root/ansibleplaybook/index.php
    dest: /var/www/html/index.php
    force: yes
  notify:
    - restart httpd service

handlers:
  - name: restart httpd service
    service:
      name: httpd
      state: restarted
```

[root@ansibile1 ansibleplaybook]#

Scroll for details

Ansible Handlers Tutorial | Ansible Notify And Handler

```
- mysql-community-client
  - php
  - php-mysql
  state: installed

- name: Enabling httpd service and start the service
  service:
    name: httpd
    enabled: yes
    state: started

- name: Enabling mysql service and start the service
  service:
    name: mysqld
    enabled: yes
    state: started

- name: Copying your php file in the /var/www/html
  copy:
    src: /root/ansibleplaybook/index.php
    dest: /var/www/html/index.php
    force: yes
  notify:
    - restart httpd service

handlers:
  - name: restart httpd service
    service:
      name: httpd
      state: restarted
```

[root@ansibile1 ansibleplaybook]#

Scroll for details



Ansible Handlers Tutorial | Ansible Notify And Handler



```
- mysql-community-client
  - php
  - php-mysql
  state: installed

- name: Enabling httpd service and start the service
  service:
    name: httpd
    enabled: yes
    state: started

- name: Enabling mysql service and start the service
  service:
    name: mysqld
    enabled: yes
    state: started

- name: Copying your php file in the /var/www/html
  copy:
    src: /root/ansibleplaybook/index.php
    dest: /var/www/html/index.php
    force: yes
  notify:
    - restart httpd service
```

handlers:

```
- name: restart httpd service
  service:
    name: httpd
    state: restarted
```

[root@ansibile1 ansibleplaybook]#

Scroll for details



[root@ansible1 ansibleplaybook]# ansible-playbook handlers.yml

```
PLAY [testservers] *****  
TASK [Gathering Facts] *****  
ok: [192.168.145.146]  
  
TASK [Installing Apache, MYSQL DB Community edition & PHP] *****  
ok: [192.168.145.146]  
  
TASK [Enabling httpd service and start the service] *****  
ok: [192.168.145.146]  
  
TASK [Enabling mysql service and start the service] *****  
ok: [192.168.145.146]  
  
TASK [Copying your php file in the /var/www/html] *****  
ok: [192.168.145.146]  
  
PLAY RECAP *****  
192.168.145.146 : ok=5    changed=0    unreachable=0    failed=0    skipped=0    rescued=0    ignored=0  
  
[root@ansible1 ansibleplaybook]#
```

[root@ansible1 ansibleplaybook]# ansible-playbook handlers.yml

```
PLAY [testservers] *****  
TASK [Gathering Facts] *****  
ok: [192.168.145.146]  
  
TASK [Installing Apache, MYSQL DB Community edition & PHP] *****  
ok: [192.168.145.146]  
  
TASK [Enabling httpd service and start the service] *****  
ok: [192.168.145.146]  
  
TASK [Enabling mysql service and start the service] *****  
ok: [192.168.145.146]  
  
TASK [Copying your php file in the /var/www/html] *****  
ok: [192.168.145.146]  
  
PLAY RECAP *****  
192.168.145.146 : ok=5    changed=0    unreachable=0    failed=0    skipped=0    rescued=0    ignored=0
```

[root@ansible1 ansibleplaybook]# vi index.php

Ahsible Handlers Tutorial | Ansible Notify And Handler



```
<html>
<head>
  <title>Ansible Application</title>
</head>
<body>

<?php
echo " Hello From Thetips4you"

?>

</body>
</html>
```

I

Ahsible Handlers Tutorial | Ansible Notify And Handler



```
<html>
<head>
  <title>Ansible Application</title>
</head>
<body>

  <?php

    echo " Hello From Thetips4you, This is part of Handlers Ansible Tutorial"

  ?>

</body>
</html>
```

I

Ansible Handlers Tutorial | Ansible Notify And Handler

```
[root@ansible1 ansibleplaybook]# ansible-playbook handlers.yml

PLAY [testservers] *****

TASK [Gathering Facts] *****
ok: [192.168.145.146]

TASK [Installing Apache, MYSQL DB Community edition & PHP] *****
ok: [192.168.145.146]

TASK [Enabling httpd service and start the service] *****
ok: [192.168.145.146]

TASK [Enabling mysql service and start the service] *****
ok: [192.168.145.146]

TASK [Copying your php file in the /var/www/html] *****
ok: [192.168.145.146]

PLAY RECAP *****
192.168.145.146 : ok=5    changed=0    unreachable=0    failed=0    skipped=0    rescued=0    ignored=0

[root@ansible1 ansibleplaybook]# vi index.php
[root@ansible1 ansibleplaybook]# ansible-playbook handlers.yml
```

Ansible Handlers Tutorial | Ansible Notify And Handler



```
ok: [192.168.145.146]

PLAY RECAP ****
192.168.145.146 : ok=5     changed=0      unreachable=0    failed=0     skipped=0    rescued=0    ignored=0

[root@ansible1 ansibleplaybook]# vi index.php
[root@ansible1 ansibleplaybook]# ansible-playbook handlers.yml

PLAY [testservers] **

TASK [Gathering Facts] ****
ok: [192.168.145.146]

TASK [Installing Apache, MYSQL DB Community edition & PHP] ****
ok: [192.168.145.146]

TASK [Enabling httpd service and start the service] ****
ok: [192.168.145.146]

TASK [Enabling mysql service and start the service] ****
ok: [192.168.145.146]

TASK [Copying your php file in the /var/www/html] ****
changed: [192.168.145.146]

RUNNING HANDLER [restart httpd service] ****
changed: [192.168.145.146]

PLAY RECAP ****
192.168.145.146 : ok=6     changed=2      unreachable=0    failed=0     skipped=0    rescued=0    ignored=0
```

[root@ansible1 ansibleplaybook]#

Scroll for details



Ansible Handlers Tutorial | Ansible Notify And Handler



```
ok: [192.168.145.146]

PLAY RECAP ****
192.168.145.146 : ok=5     changed=0      unreachable=0    failed=0     skipped=0    rescued=0    ignored=0

[root@ansible1 ansibleplaybook]# vi index.php
[root@ansible1 ansibleplaybook]# ansible-playbook handlers.yml

PLAY [testservers] **

TASK [Gathering Facts] ****
ok: [192.168.145.146]

TASK [Installing Apache, MYSQL DB Community edition & PHP] ****
ok: [192.168.145.146]

TASK [Enabling httpd service and start the service] ****
ok: [192.168.145.146]

TASK [Enabling mysql service and start the service] ****
ok: [192.168.145.146]

TASK [Copying your php file in the /var/www/html] ****
changed: [192.168.145.146]

RUNNING HANDLER [restart httpd service] ****
changed: [192.168.145.146]

PLAY RECAP ****
192.168.145.146 : ok=6     changed=2      unreachable=0    failed=0     skipped=0    rescued=0    ignored=0
```

[root@ansible1 ansibleplaybook]#

Scroll for details



```
Ansible Handlers Tutorial | Ansible Notify And Handler
ok: [192.168.145.146]

PLAY RECAP ****
192.168.145.146 : ok=5    changed=0    unreachable=0    failed=0    skipped=0    rescued=0    ignored=0

[root@ansible1 ansibleplaybook]# vi index.php
[root@ansible1 ansibleplaybook]# ansible-playbook handlers.yml

PLAY [testservers] **

TASK [Gathering Facts] ****
ok: [192.168.145.146]

TASK [Installing Apache, MYSQL DB Community edition & PHP] ****
ok: [192.168.145.146]

TASK [Enabling httpd service and start the service] ****
ok: [192.168.145.146]

TASK [Enabling mysql service and start the service] ****
ok: [192.168.145.146]

TASK [Copying your php file in the /var/www/html] ****
changed: [192.168.145.146]

RUNNING HANDLER [restart httpd service] ****
changed: [192.168.145.146]

PLAY RECAP ****
192.168.145.146 : ok=6    changed=2    unreachable=0    failed=0    skipped=0    rescued=0    ignored=0

[root@ansible1 ansibleplaybook]#
```

[root@ansible1 ansibleplaybook]# ansible-playbook handlers.yml



◀ ▶ ⏪ ⏹ 8:28 / 11:14

Scroll for details

CC HD

Ansible Handlers Tutorial | Ansible Notify And Handler
[root@ansible1 ansibleplaybook]# ansible-playbook handlers.yml

```
PLAY [testservers] *****  
  
TASK [Gathering Facts] *****  
ok: [192.168.145.146]  
  
TASK [Installing Apache, MYSQL DB Community edition & PHP] *****  
ok: [192.168.145.146]  
  
TASK [Enabling httpd service and start the service] *****  
ok: [192.168.145.146]  
  
TASK [Enabling mysql service and start the service] *****  
ok: [192.168.145.146]  
  
TASK [Copying your php file in the /var/www/html] *****  
ok: [192.168.145.146]  
  
PLAY RECAP *****  
192.168.145.146 : ok=5    changed=0    unreachable=0    failed=0    skipped=0    rescued=0    ignored=0  
  
[root@ansible1 ansibleplaybook]#
```

Ansible Handlers Tutorial | Ansible Notify And Handler

```
[root@localhost ~]# cd /var/www/html
[root@localhost html]#
```

I

8:49 / 11:14

Scroll for details

CC HD

An Ansible Handlers Tutorial | Ansible Notify And Handler



```
[root@localhost ~]# cd /var/www/html
[root@localhost html]# cat index.php

<html>
<head>
<title>Ansible Application</title>
</head>
<body>

<?php
echo " Hello From Thetips4you, This is part of Handlers Ansible Tutorial"
?>

</body>
</html>
[root@localhost html]#
```

Ansible Handlers Tutorial | Ansible Notify And Handler
[root@ansible1 ansibleplaybook]# ansible-playbook handlers.yml

```
PLAY [testservers] *****  
  
TASK [Gathering Facts] *****  
ok: [192.168.145.146]  
  
TASK [Installing Apache, MYSQL DB Community edition & PHP] *****  
ok: [192.168.145.146]  
  
TASK [Enabling httpd service and start the service] *****  
ok: [192.168.145.146]  
  
TASK [Enabling mysql service and start the service] *****  
ok: [192.168.145.146]  
  
TASK [Copying your php file in the /var/www/html] *****  
ok: [192.168.145.146]  
  
PLAY RECAP *****  
192.168.145.146 : ok=5    changed=0    unreachable=0    failed=0    skipped=0    rescued=0    ignored=0  
  
[root@ansible1 ansibleplaybook]# vi index.php
```

Ahsible Handlers Tutorial | Ansible Notify And Handler



```
<html>
<head>
  <title>Ansible Application</title>
</head>
<body>

  <?php

    echo " Hello From Thetips4you, This is part of Handlers Ansible Tutorial"

  ?>

</body>
</html>
```

I

Ahsible Handlers Tutorial | Ansible Notify And Handler



```
<html>
<head>
  <title>Ansible Application</title>
</head>
<body>

<?php

echo " Hello From Thetips4yo[REDACTED]

?>

</body>
</html>
```

Ansible Handlers Tutorial | Ansible Notify And Handler
[root@ansible1 ansibleplaybook]# ansible-playbook handlers.yml



I

◀ ▶ ⏪ 🔍 9:17 / 11:14

Scroll for details

CC HD

Ansible Handlers Tutorial | Ansible Notify And Handler
[root@ansible1 ansibleplaybook]# ansible-playbook handlers.yml

```
PLAY [testservers] *****  
  
TASK [Gathering Facts] *****  
ok: [192.168.145.146]  
  
TASK [Installing Apache, MYSQL DB Community edition & PHP] *****  
ok: [192.168.145.146]  
  
TASK [Enabling httpd service and start the service] *****  
ok: [192.168.145.146]  
  
TASK [Enabling mysql service and start the service] *****  
ok: [192.168.145.146]  
  
TASK [Copying your php file in the /var/www/html] *****  
changed: [192.168.145.146]  
  
RUNNING HANDLER [restart httpd service] *****  
changed: [192.168.145.146]  
  
PLAY RECAP *****  
192.168.145.146 : ok=6    changed=2    unreachable=0    failed=0    skipped=0    rescued=0    ignored=0  
[root@ansible1 ansibleplaybook]#
```

Ansible Handlers Tutorial | Ansible Notify And Handler



```
[root@localhost ~]# cd /var/www/html
[root@localhost html]# cat index.php

<html>
<head>
<title>Ansible Application</title>
</head>
<body>

<?php
echo " Hello From Thetips4you, This is part of Handlers Ansible Tutorial"
?>

</body>
</html>
[root@localhost html]#
```

Ansible Handlers Tutorial | Ansible Notify And Handler



```
[root@localhost ~]# cd /var/www/html
[root@localhost html]# cat index.php

<html>
<head>
<title>Ansible Application</title>
</head>
<body>

<?php

echo " Hello From Thetips4you, This is part of Handlers Ansible Tutorial"

?>

</body>
</html>
[root@localhost html]# cat index.php
```

Ansible Handlers Tutorial | Ansible Notify And Handler



```
<html>
<head>
<title>Ansible Application</title>
</head>
<body>

<?php

echo " Hello From Thetips4you, This is part of Handlers Ansible Tutorial"

?>

</body>
</html>
[root@localhost html]# cat index.php

<html>
<head>
<title>Ansible Application</title>
</head>
<body>

<?php

echo " Hello From Thetips4you"

?>

</body>
</html>
[root@localhost html]# 9:33 / 11:14
```

Scroll for details



[root@ansible1 ansibleplaybook]# vi handlers.yml



I

Ahsible Handlers Tutorial | Ansible Notify And Handler



```
---  
- hosts: testservers  
  remote_user: root  
  
tasks:  
  - name: Installing Apache, MYSQL DB Community edition & PHP  
    yum:  
      name:  
        - httpd  
        - mysql-community-server  
        - mysql-community-client  
        - php  
        - php-mysql  
      state: installed  
  
  - name: Enabling httpd service and start the service  
    service:  
      name: httpd  
      enabled: yes  
      state: started  
  
  - name: Enabling mysql service and start the service  
    service:  
      name: mysqld  
      enabled: yes  
      state: started
```

Ansible Handlers Tutorial | Ansible Notify And Handler



```
- hosts: testservers
  remote_user: root

tasks:
  - name: Installing Apache, MYSQL DB Community edition & PHP
    yum:
      name:
        - httpd
        - mysql-community-server
        - mysql-community-client
        - php
        - php-mysql
      state: installed

  - name: Enabling httpd service and start the service
    service:
      name: httpd
      enabled: yes
      state: started

  - name: Enabling mysql service and start the service
    service:
      name: mysqld
      enabled: yes
      state: started

  - name: Copying your php file in the /var/www/html
    copy:
      src: /root/ansibleplaybook/index.php
```

◀ ▶ ⏪ ⏩ 🔍 10:00 / 11:14

Scroll for details

CC HD

Ansible Handlers Tutorial | Ansible Notify And Handler



```
- mysql-community-client
  - php
  - php-mysql
  state: installed

- name: Enabling httpd service and start the service
  service:
    name: httpd
    enabled: yes
    state: started

- name: Enabling mysql service and start the service
  service:
    name: mysqld
    enabled: yes
    state: started

- name: Copying your php file in the /var/www/html
  copy:
    src: /root/ansibleplaybook/index.php
    dest: /var/www/html/index.php
    force: yes
  notify:
    - restart httpd service

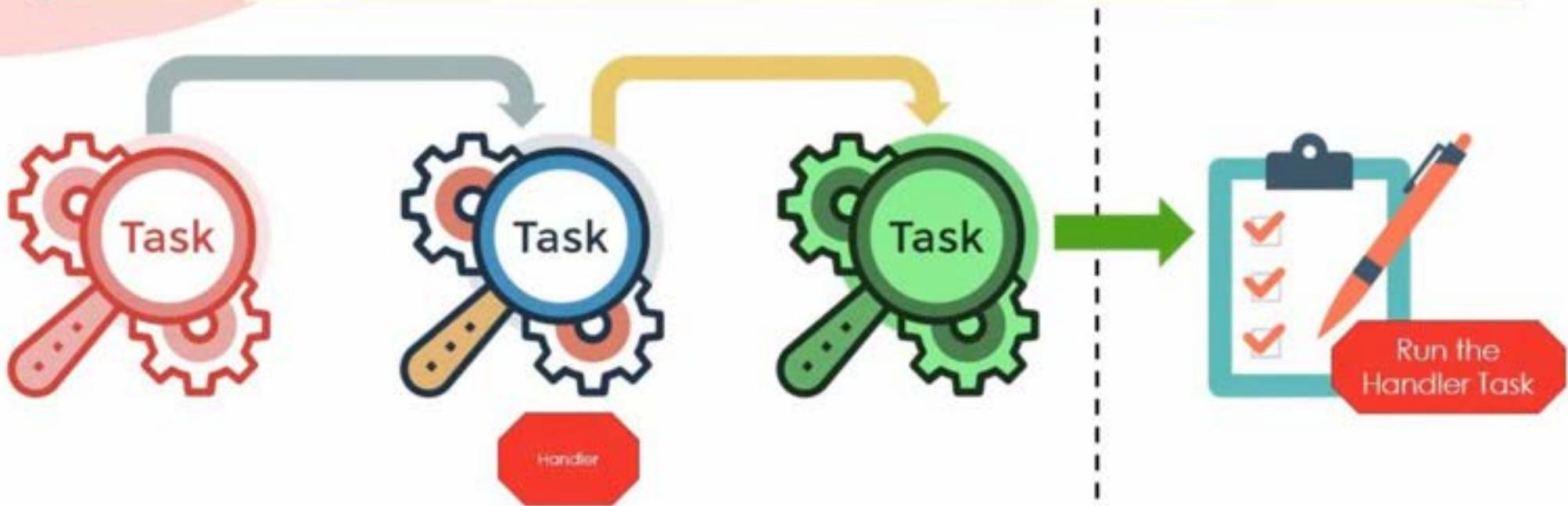
handlers:
  - name: restart httpd service
    service:
      name: httpd
      state: restarted
```

◀ ▶ ▶ 🔍 10:27 / 11:14

Scroll for details

CC HD

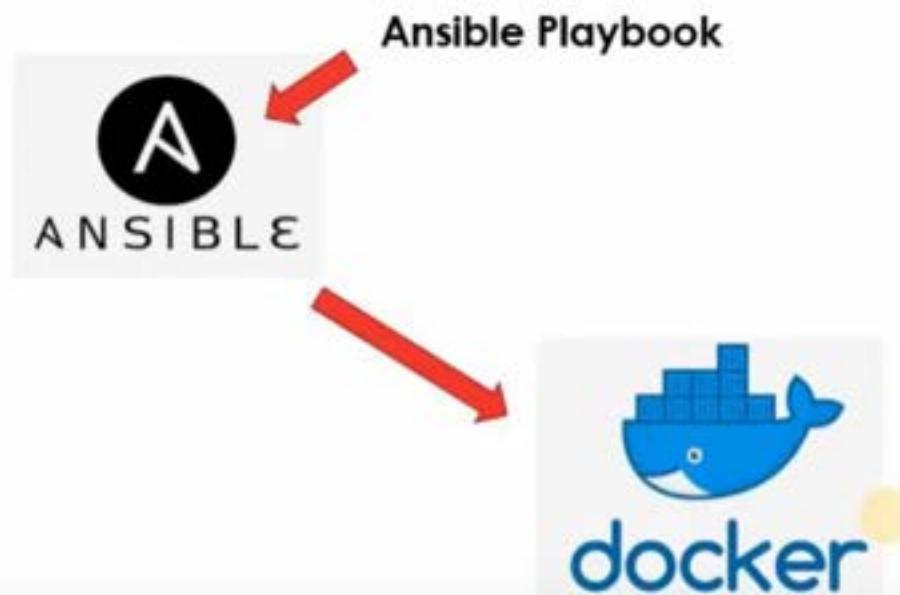
Handlers in ansible playbook.



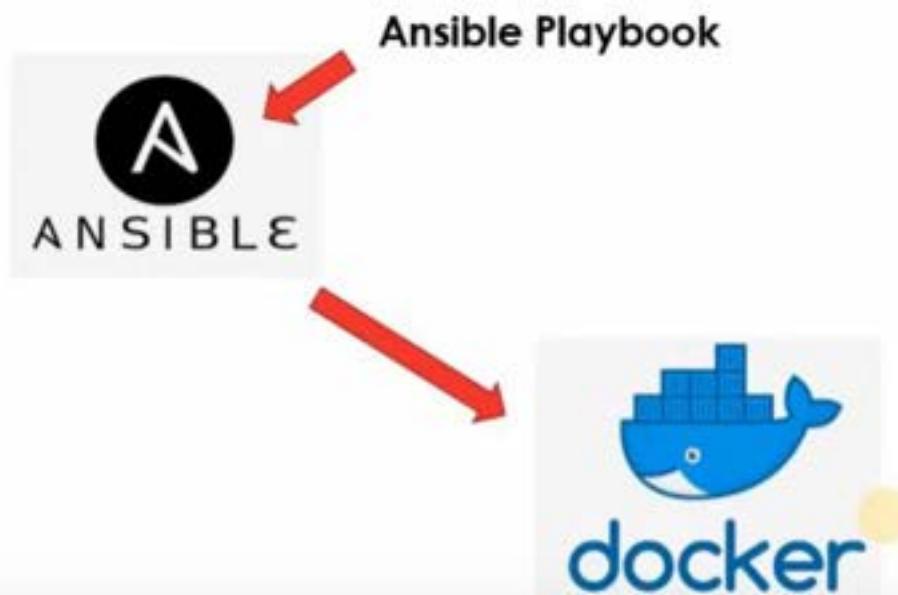
- ✓ Handlers run after all the tasks in a particular play have been completed. Handler only runs once, regardless of how many tasks notify it.



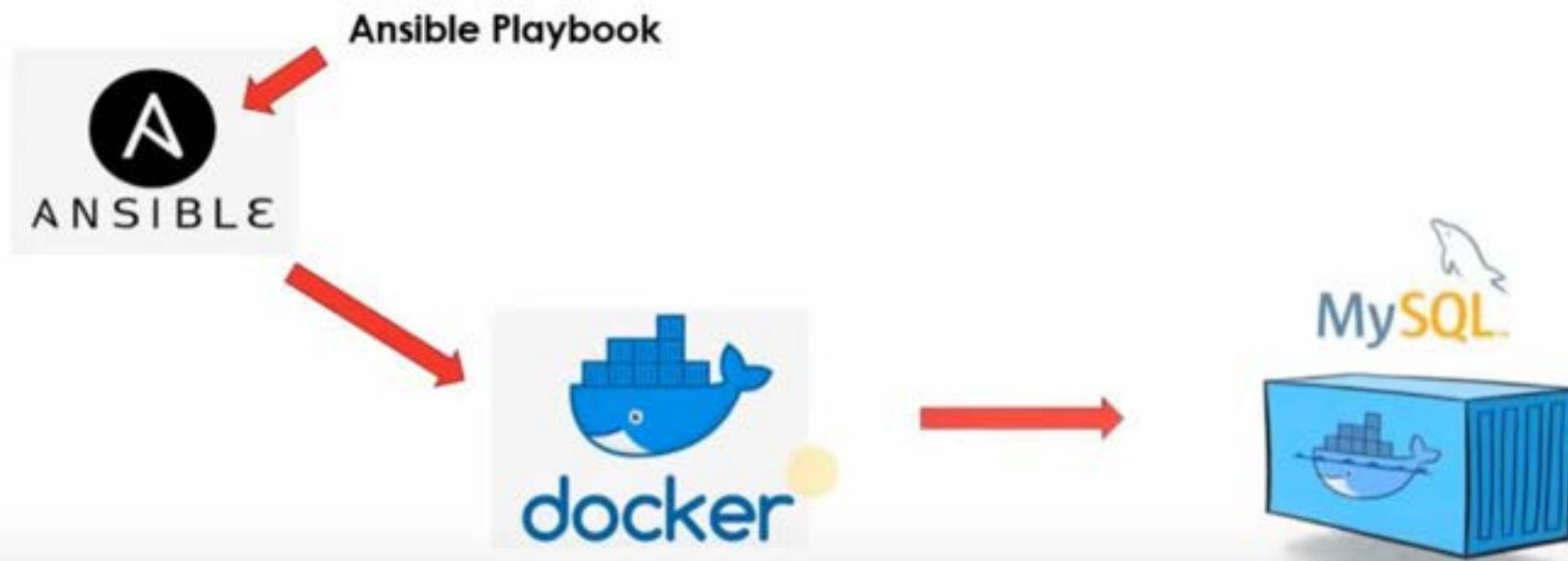
Ansible Playbook for creating MySQL Docker Container



Ansible Playbook for creating MYSQL Docker Container

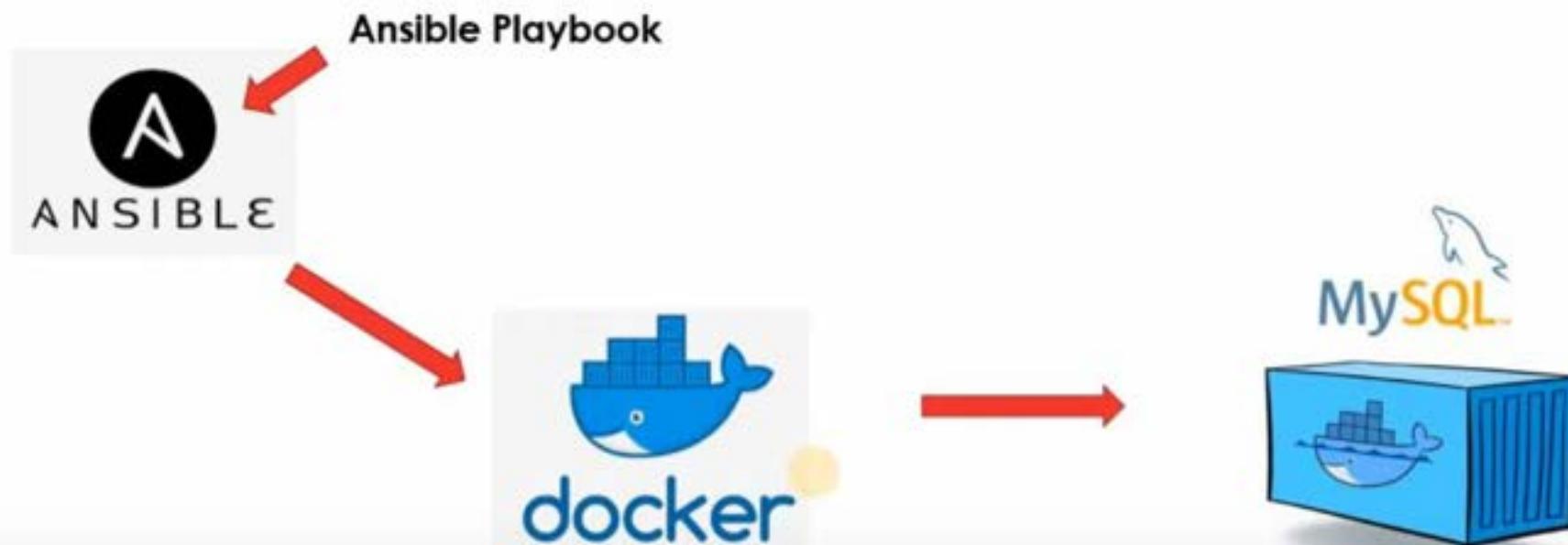


Ansible Playbook for creating MySQL Docker Container



```
[root@ansible1 ansible]# cd ~  
[root@ansible1 ~]# cd ansibleplaybook/  
[root@ansible1 ansibleplaybook]# vi mysqldocker.yml  
[root@ansible1 ansibleplaybook]# ansible-playbook mysqldocker.yml  
  
PLAY [Running MYSQL Database as Docker Container using Ansible.] *****  
  
TASK [Gathering Facts] *****  
ok: [192.168.145.146]  
  
TASK [Launch mysql database container] *****  
changed: [192.168.145.146]  
  
PLAY RECAP *****  
192.168.145.146 : ok=2    changed=1    unreachable=0    failed=0    skipped=0    rescued=0    ignored=0  
  
[root@ansible1 ansibleplaybook]#
```

Ansible Playbook for creating MySQL Docker Container



[root@ansible1 ansibleplaybook]# cd /etc/ansible/
[root@ansible1 ansible]# ls
ansible.cfg hosts roles
[root@ansible1 ansible]# cat hosts

I

Running MYSQL Docker Container Using Ansible Playbook | Ansible Deploy Docker Containe...



```
[testservers]
192.168.1.46

## [webservers]
## alpha.example.org
## beta.example.org
## 192.168.1.100
## 192.168.1.110

# If you have multiple hosts following a pattern you can specify
# them like this:

## www[001:006].example.com

# Ex 3: A collection of database servers in the 'dbservers' group

## [dbservers]
##
## db01.intranet.mydomain.net
## db02.intranet.mydomain.net
## 10.25.1.56
## 10.25.1.57

# Here's another example of host ranges, this time there are no
# leading 0s:

## db-[99:101]-node.example.com
```

[root@ansible1 ansible]# 2.12 / 9:06 • Subscribe >

Scroll for details



Running MYSQL Docker Container Using Ansible Playbook | Ansible Deploy Docker Containe...

```
[root@localhost ~]# ip addr
1: lo: <LOOPBACK,UP,LOWER_UP> mtu 65536 qdisc noqueue state UNKNOWN group default qlen 1000
    link/loopback 00:00:00:00:00:00 brd 00:00:00:00:00:00
    inet 127.0.0.1/8 scope host lo
        valid_lft forever preferred_lft forever
    inet6 ::1/128 scope host
        valid_lft forever preferred_lft forever
2: ens33: <BROADCAST,MULTICAST,UP,LOWER_UP> mtu 1500 qdisc pfifo_fast state UP group default qlen 1000
    link/ether 00:0c:29:ee:1c:fe brd ff:ff:ff:ff:ff:ff
    inet 192.168.145.146/24 brd 192.168.145.255 scope global noprefixroute dynamic ens33
        valid_lft 1764sec preferred_lft 1764sec
    inet6 fe80::1245:15f8:cae4:d2e/64 scope link noprefixroute
        valid_lft forever preferred_lft forever
3: docker0: <NO-CARRIER,BROADCAST,MULTICAST,UP> mtu 1500 qdisc noqueue state DOWN group default
    link/ether 02:42:72:b2:e4:24 brd ff:ff:ff:ff:ff:ff
    inet 172.17.0.1/16 brd 172.17.255.255 scope global docker0
        valid_lft forever preferred_lft forever
    inet6 fe80::42:72ff:feb2:e424/64 scope link
        valid_lft forever preferred_lft forever
[root@localhost ~]#
```

[root@localhost ~]# docker -v
Docker version 20.10.3, build 48d30b5
[root@localhost ~]#

[root@localhost ~]# docker -v
Docker version 20.10.3, build 48d30b5
[root@localhost ~]# docker ps

CONTAINER ID	IMAGE	COMMAND	CREATED	STATUS	PORTS	NAMES

[root@localhost ~]#

Running MYSQL Docker Container Using Ansible Playbook | Ansible Deploy Docker Containe...

```
[root@localhost ~]# docker -v
Docker version 20.10.3, build 48d30b5
[root@localhost ~]# docker ps
CONTAINER ID   IMAGE      COMMAND   CREATED     STATUS      PORTS     NAMES
[root@localhost ~]# docker ps -a
CONTAINER ID   IMAGE      COMMAND   CREATED     STATUS      PORTS     NAMES
[root@localhost ~]#
```

I

2:39 / 9:06 • Subscribe > Scroll for details CC HD

Running MYSQL Docker Container Using Ansible Playbook | Ansible Deploy Docker Containe...

```
1  ---
2  - name: Running MYSQL Database as Docker Container using Ansible.
3  hosts: testservers
4  remote_user: root
5
6
7  vars:
8    db_volume: db_data
9
10
11 tasks:
12
13 - name: Launch mysql database container
14   docker_container:
15     name: db
16     image: mysql:5.7
17     volumes:
18       - "{{ db_volume }}:/var/lib/mysql"
19     restart: true
20     env:
21       MYSQL_ROOT_PASSWORD: password
22       MYSQL_DATABASE: test
23       MYSQL_USER: wordpress
24       MYSQL_PASSWORD: wordpress
25
```

Running MYSQL Docker Container Using Ansible Playbook | Ansible Deploy Docker Containe...

```
1  ---
2  - name: Running MYSQL Database as Docker Container using Ansible.
3  hosts: testservers
4  remote_user: root
5
6
7  vars:
8      db_volume: db_data
9
10
11 tasks:
12
13 - name: Launch mysql database container
14   docker_container:
15     name: db
16     image: mysql:5.7
17     volumes:
18       - "{{ db_volume }}:/var/lib/mysql"
19     restart: true
20     env:
21       MYSQL_ROOT_PASSWORD: password
22       MYSQL_DATABASE: test
23       MYSQL_USER: wordpress
24       MYSQL_PASSWORD: wordpress
25     ports:
```

Running MySQL Docker Container Using Ansible Playbook

Suggested: DevOps | CI CD Pipeline | Setup Docker On

Intos 7

```
File Edit Selection Find View Goto Tools Project Preferences Help
FOLDERS mysqldockercontaineransible.yml
4 remote_user: root
5
6
7 vars:
8   db_volume: db_data
9
10
11 tasks:
12
13 - name: Launch mysql database container
14   docker_container:
15     name: db
16     image: mysql:5.7
17     volumes:
18       - "{{ db_volume }}:/var/lib/mysql"
19     restart: true
20     env:
21       MYSQL_ROOT_PASSWORD: password
22       MYSQL_DATABASE: test
23       MYSQL_USER: wordpress
24       MYSQL_PASSWORD: wordpress
25     ports:
26       - "33306:3306"
```

Running MySQL Docker Container Using Ansible Playbook | Ansible Deploy Docker Containe...

```
File Edit Selection Find View Goto Tools Project Preferences Help
FOLD mysqldockercontaineransible.yml
7 vars:
8   db_volume: db_data
9
10
11 tasks:
12
13 - name: Launch mysql database container
14   docker_container:
15     name: db
16     image: mysql:5.7
17     volumes:
18       - "{{ db_volume }}:/var/lib/mysql"
19     restart: true
20     env:
21       MYSQL_ROOT_PASSWORD: password
22       MYSQL_DATABASE: test
23       MYSQL_USER: wordpress
24       MYSQL_PASSWORD: wordpress
25     ports:
26       - "33306:3306"
```

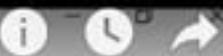
Running MYSQL Docker Container Using Ansible Playbook | Ansible Deploy Docker Containe...

```
1 ---
2 - name: Running MYSQL Database as Docker Container using Ansible.
3   hosts: testservers
4   remote_user: root
5
6
7   vars:
8     db_volume: db_data
9
10
11  tasks:
12
13  - name: Launch mysql database container
14    docker_container:
15      name: db
16      image: mysql:5.7
17      volumes:
18        - "{{ db_volume }}:/var/lib/mysql"
19      restart: true
20      env:
21        MYSQL_ROOT_PASSWORD: password
22        MYSQL_DATABASE: test
23        MYSQL_USER: wordpress
24        MYSQL_PASSWORD: wordpress
25      ports:
```

Running MYSQL Docker Container Using Ansible Playbook | Ansible Deploy Docker Containe...   
[root@ansible1 ansible]# cd ~
[root@ansible1 ~]# cd ansibleplaybook/
[root@ansible1 ansibleplaybook]# vi mysqldocker.yml

I

Running MYSQL Docker Container Using Ansible Playbook | Ansible Deploy Docker Containe...



```
---  
- name: Running MYSQL Database as Docker Container using Ansible.  
hosts: testservers  
remote_user: root  
  
vars:  
  db_volume: db_data  
  
tasks:  
  - name: Launch mysql database container  
    docker_container:  
      name: db  
      image: mysql:5.7  
      volumes:  
        - "{{ db_volume }}:/var/lib/mysql"  
      restart: true  
      env:  
        MYSQL_ROOT_PASSWORD: password  
        MYSQL_DATABASE: test  
        MYSQL_USER: wordpress  
        MYSQL_PASSWORD: wordpress  
      ports:  
        - "33306:3306"
```

```
root@ansible:~/ansibleplaybook
```

```
---
```

```
- name: Running MYSQL Database as Docker Container using Ansible.
  hosts: testservers
  remote_user: root
```

```
I
```

```
vars:
  db_volume: db_data
```

```
tasks:
```

```
- name: Launch mysql database container
  docker_container:
    name: db
    image: mysql:5.7
    volumes:
      - "{{ db_volume }}:/var/lib/mysql"
    restart: true
    env:
      MYSQL_ROOT_PASSWORD: password
      MYSQL_DATABASE: test
      MYSQL_USER: wordpress
      MYSQL_PASSWORD: wordpress
    ports:
      - "33306:3306"
```

[root@ansible1 ansible]# cd ~
[root@ansible1 ~]# cd ansibleplaybook/
[root@ansible1 ansibleplaybook]# vi mysqldocker.yml
[root@ansible1 ansibleplaybook]# ansible-playbook mysqldocker.yml

PLAY [Running MYSQL Database as Docker Container using Ansible.] *****

TASK [Gathering Facts] *****
ok: [192.168.145.146]

TASK [Launch mysql database container] *****
changed: [192.168.145.146]

PLAY RECAP *****
192.168.145.146 : ok=2 changed=1 unreachable=0 failed=0 skipped=0 rescued=0 ignored=0

[root@ansible1 ansibleplaybook]#

[root@ansible1 ansible]# cd ~
[root@ansible1 ~]# cd ansibleplaybook/
[root@ansible1 ansibleplaybook]# vi mysqldocker.yml
[root@ansible1 ansibleplaybook]# ansible-playbook mysqldocker.yml

PLAY [Running MYSQL Database as Docker Container using Ansible.] *****

TASK [Gathering Facts] *****
ok: [192.168.145.146]

TASK [Launch mysql database container] *****
changed: [192.168.145.146]

PLAY RECAP *****
192.168.145.146 : ok=2 changed=1 unreachable=0 failed=0 skipped=0 rescued=0 ignored=0

[root@ansible1 ansibleplaybook]#

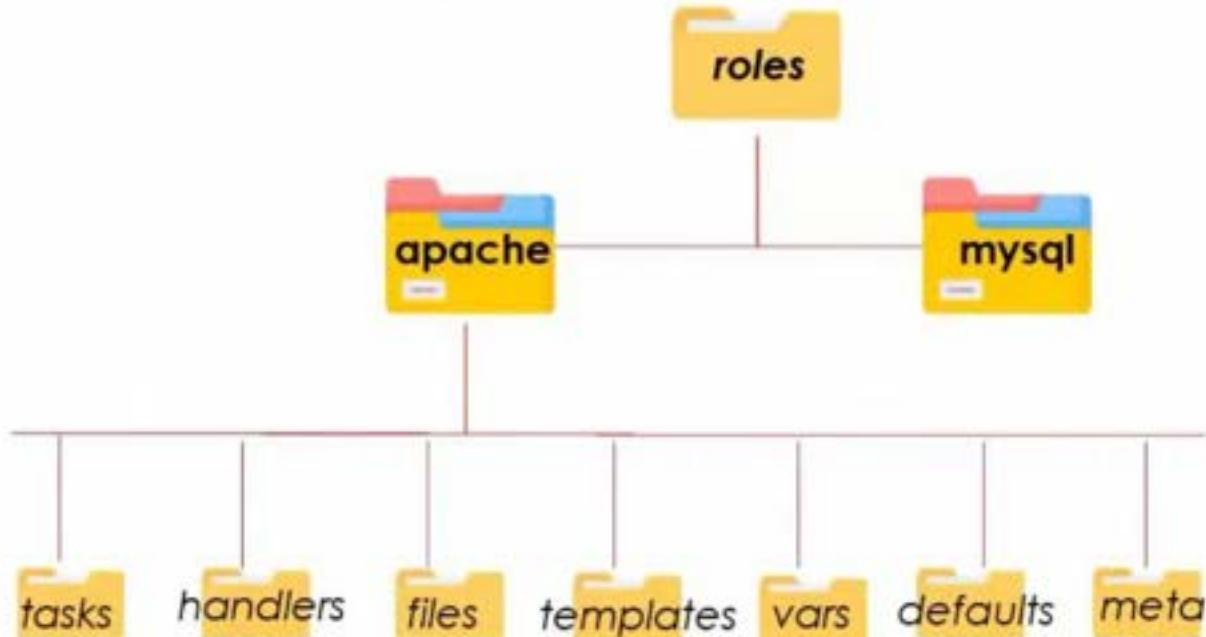
```
Running MYSQL Docker Container Using Ansible Playbook | Ansible How to create simple ansible playbook to set up mysql db [root@localhost ~]# docker ps
```

CONTAINER ID	IMAGE	COMMAND	CREATED	STATUS	PORTS
77e40afedfdc	mysql:5.7	"docker-entrypoint.s..."	15 seconds ago	Up 12 seconds	33060/tcp, 0.0.0.0:33306->3306/tcp

```
[root@localhost ~]#
```

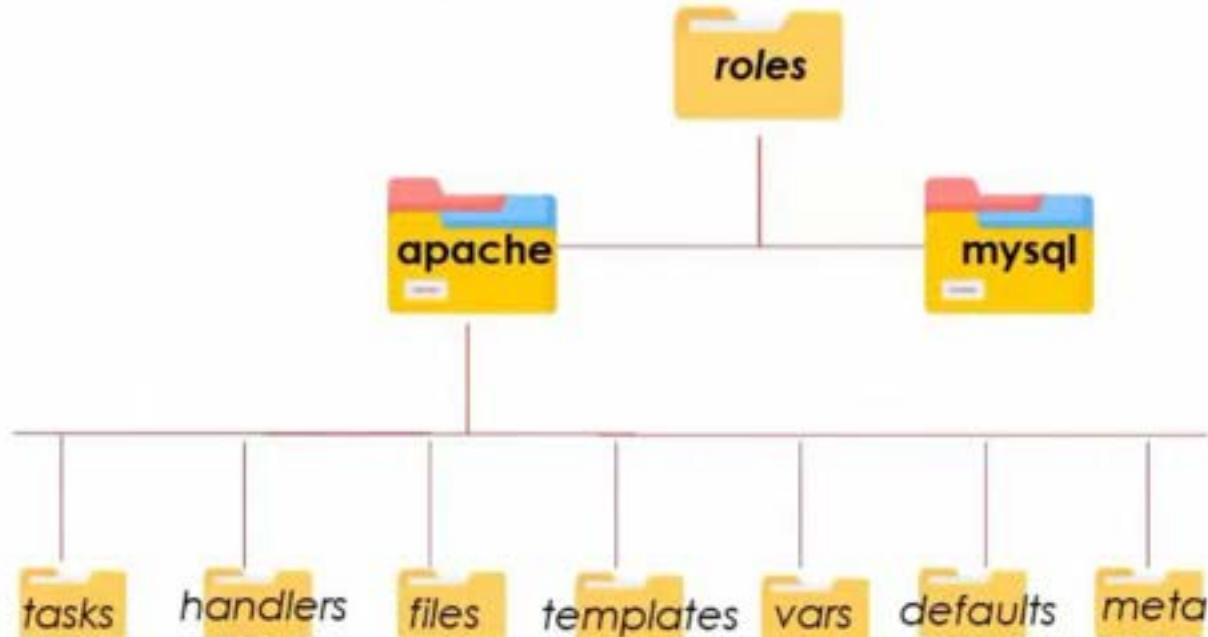
Role directory structure

```
roles/  
  apache/  
    tasks/  
    handlers/  
    library/  
    files/  
    templates/  
    vars/  
    defaults/  
    meta/  
  
  mysql/  
    tasks/  
    handlers/  
    library/  
    files/  
    templates/  
    vars/  
    defaults/  
    meta/
```



Role directory structure

```
roles/  
  apache/  
    tasks/  
    handlers/  
    library/  
    files/  
    templates/  
    vars/  
    defaults/  
    meta/  
  
  mysql/  
    tasks/  
    handlers/  
    library/  
    files/  
    templates/  
    vars/  
    defaults/  
    meta/
```



What is Ansible Roles? | Ansible Roles Explained | Role directory structure & creating a new Role.

I

Agenda

- **What is Ansible Roles.**
- **Role directory structure.**
- **Creating a new role.**
- **Using roles in playbook.**

What is Ansible?

- Ansible is an open-source software provisioning, configuration management, and application-deployment tool enabling infrastructure as code.
- Ansible is a simple, agentless IT automation tool that anyone can use to automate complex IT application environments.
- Since Ansible is called as agentless tool, it uses PUSH Approach and not PULL approach.

Ansible Playbook

- A playbook is a configuration file written in YAML (YAML stands for yet another markup language) that provides instructions for what needs to be done.
- Ansible playbooks are executed on a set, group, or classification of hosts, which together make up an Ansible inventory.
- One or more Ansible tasks can be combined to make a play. Two or more plays can be combined to create an Ansible playbook.

Ansible Playbook

- A playbook is a configuration file written in YAML (YAML stands for yet another markup language) that provides instructions for what needs to be done.
- Ansible playbooks are executed on a set, group, or classification of hosts, which together make up an Ansible inventory.
- One or more Ansible tasks can be combined to make a play. Two or more plays can be combined to create an Ansible playbook.

☰ What is Ansible Roles? | Ansible Roles Explained | Role directory structure & creating a new Role.



```
---  
- hosts: testservers  
  remote_user: root  
  
  tasks:  
  
    - name: Installing Apache2  
      yum:  
        name: httpd  
        state: installed  
  
    - name: Enabling and starting the apache service  
      service:  
        name: httpd  
        enabled: yes  
        state: started  
  
    - name: Installing MYSQL Community Server  
      yum:  
        name:  
          - mariadb-server  
          - mariadb  
        state: latest  
  
    - name: Starting the MySQL DB service  
      service:  
        name: mysqld  
        enabled: yes  
        state: started
```

☰ What is Ansible Roles? | Ansible Roles Explained | Role directory structure & creating a new Role.



```
---  
- hosts: testservers  
  remote_user: root  
  
  tasks:  
  
    - name: Installing Apache2  
      yum:  
        name: httpd  
        state: installed  
  
    - name: Enabling and starting the apache service  
      service:  
        name: httpd  
        enabled: yes  
        state: started  
  
    - name: Installing MYSQL Community Server  
      yum:  
        name:  
          - mariadb-server  
          - mariadb  
        state: latest  
  
    - name: Starting the MySQL DB service  
      service:  
        name: mysqld  
        enabled: yes  
        state: started
```

What is Ansible Roles

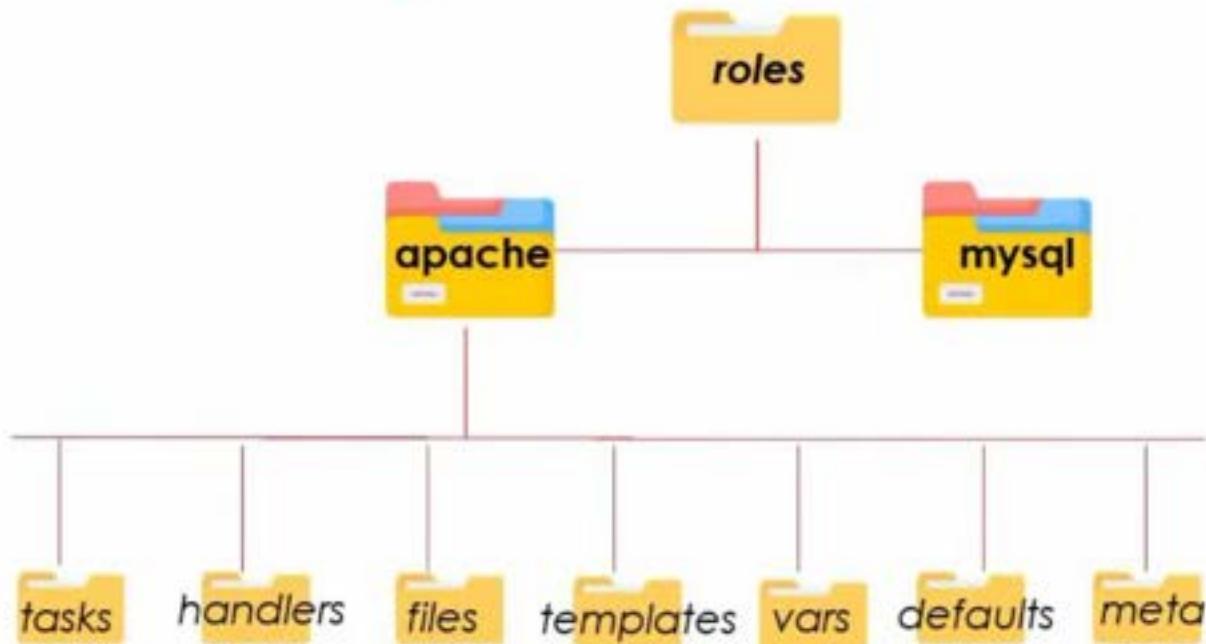
- Roles in ansible is a pre-defined directory structure that is utilized to put the individual pieces of playbook in to , to make it easier to re-use and share.
- In Ansible, the role is the primary mechanism for breaking a playbook into multiple files. This simplifies writing complex playbooks, and it makes them easier to reuse.
- Roles are not playbooks. Roles are small functionality which can be independently used but have to be used within playbooks.

```
---  
- hosts: testservers  
  remote_user: root  
  
  tasks:  
  
    - name: Installing Apache2  
      yum:  
        name: httpd  
        state: installed  
  
    - name: Enabling and starting the apache service  
      service:  
        name: httpd  
        enabled: yes  
        state: started  
  
    - name: Installing MYSQL Community Server  
      yum:  
        name:  
          - mariadb-server  
          - mariadb  
        state: latest  
  
    - name: Starting the MySQL DB service  
      service:  
        name: mysqld  
        enabled: yes  
        state: started
```



Role directory structure

```
roles/  
  apache/  
    tasks/  
    handlers/  
    library/  
    files/  
    templates/  
    vars/  
    defaults/  
    meta/  
  
  mysql/  
    tasks/  
    handlers/  
    library/  
    files/  
    templates/  
    vars/  
    defaults/  
    meta/
```



Role directory structure

- defaults: Contains default variables for the role. Variables in default have the lowest priority so they are easy to override..
- vars: Contains variables for the role. Variables in vars have higher priority than variables in defaults directory..
- tasks: Contains the main list of steps to be executed by the role.
- files: Contains files which we want to be copied to the remote host. We don't need to specify a path of resources stored in this directory.
- templates: Contains file template which supports modifications from the role.
- meta: Contains metadata of role like an author, support platforms, dependencies.
- handlers: Contains handlers which can be invoked by "notify" directives and are associated with service.

Creating a New Role

- ansible-galaxy
- ansible-galaxy --help or ansible-galaxy -h
- ansible-galaxy init newrole

What is Ansible Roles? | Ansible Roles Explained | Role directory structure & creating a new Role.

```
[root@ansible1 ansibleroles]# cd /etc/ansible/  
[root@ansible1 ansible]# ls  
ansible.cfg hosts roles  
[root@ansible1 ansible]# cd roles/  
[root@ansible1 roles]# ls  
[root@ansible1 roles]#
```

I



10:36 / 14:13 • Creating a New Role >

Scroll for details



≡ What is Ansible Roles? | Ansible Roles Explained | Role directory structure & creating a new Role.

```
[root@ansible1 ansibleroles]# cd /etc/ansible/
[root@ansible1 ansible]# ls
ansible.cfg hosts roles
[root@ansible1 ansible]# cd roles/
[root@ansible1 roles]# ls
[root@ansible1 roles]# cd ~
[root@ansible1 ~]# cd ansibleroles/
[root@ansible1 ansibleroles]#
```

I



10:51 / 14:13 • Creating a New Role > Scroll for details



≡ What is Ansible Roles? | Ansible Roles Explained | Role directory structure & creating a new Role.

```
[root@ansible1 ansibleroles]# cd /etc/ansible/
[root@ansible1 ansible]# ls
ansible.cfg hosts roles
[root@ansible1 ansible]# cd roles/
[root@ansible1 roles]# ls
[root@ansible1 roles]# cd ~
[root@ansible1 ~]# cd ansibleroles/
[root@ansible1 ansibleroles]# ls
[root@ansible1 ansibleroles]#
```

I

What is Ansible Roles? | Ansible Roles Explained | Role directory structure & creating a new Role.

```
[root@ansible1 ansibleroles]# ansible-galaxy init apache
- Role apache was created successfully
[root@ansible1 ansibleroles]# ls
apache
[root@ansible1 ansibleroles]# cd apache/
[root@ansible1 apache]# █
```

I



11:15 / 14:13 • Creating a New Role >

Scroll for details



What is Ansible Roles? | Ansible Roles Explained | Role directory structure & creating a new Role.

```
[root@ansible1 ansibleroles]# ansible-galaxy init apache
- Role apache was created successfully
[root@ansible1 ansibleroles]# ls
apache
[root@ansible1 ansibleroles]# cd apache/
[root@ansible1 apache]# ls
defaults  files  handlers  meta  README.md  tasks  templates  tests  vars
[root@ansible1 apache]#
```



11:19 / 14:13 • Creating a New Role >

Scroll for details



What is Ansible Roles? | Ansible Roles Explained | Role directory structure & creating a new Role.

```
[root@ansible1 ansibleroles]# ansible-galaxy init apache
- Role apache was created successfully
[root@ansible1 ansibleroles]# ls
apache
[root@ansible1 ansibleroles]# cd apache/
[root@ansible1 apache]# ls
defaults files handlers meta README.md tasks templates tests vars
[root@ansible1 apache]#
```



11:28 / 14:13 • Creating a New Role >

Scroll for details



What is Ansible Roles? | Ansible Roles Explained | Role directory structure & creating a new Role.

```
[root@ansible1 ansibleroles]# ansible-galaxy init apache
- Role apache was created successfully
[root@ansible1 ansibleroles]# ls
apache
[root@ansible1 ansibleroles]# cd apache/
[root@ansible1 apache]# ls
defaults  files  handlers  meta  README.md  tasks  templates  tests  vars
[root@ansible1 apache]# cd tasks/
[root@ansible1 tasks]# ls
main.yml
[root@ansible1 tasks]#
```



11:35 / 14:13 • Creating a New Role >

Scroll for details



What is Ansible Roles? | Ansible Roles Explained | Role directory structure & creating a new Role.

```
[root@ansible1 ansibleroles]# ansible-galaxy init apache
- Role apache was created successfully
[root@ansible1 ansibleroles]# ls
apache
[root@ansible1 ansibleroles]# cd apache/
[root@ansible1 apache]# ls
defaults  files  handlers  meta  README.md  tasks  templates  tests  vars
[root@ansible1 apache]# cd tasks/
[root@ansible1 tasks]# ls
main.yml
[root@ansible1 tasks]# cat main.yml
---
# tasks file for apache[root@ansible1 tasks]#
```



11:38 / 14:13 • Creating a New Role >

Scroll for details



What is Ansible Roles? | Ansible Roles Explained | Role directory structure & creating a new Role.

```
[root@ansible1 tasks]# ansible-galaxy --help
```

```
usage: ansible-galaxy [-h] [--version] [-v] TYPE ...
```

Perform various Role and Collection related operations.

positional arguments:

TYPE

```
collection  Manage an Ansible Galaxy collection.  
role        Manage an Ansible Galaxy role.
```

optional arguments:

```
--version      show program's version number, config file location,  
               configured module search path, module location, executable  
               location and exit  
-h, --help      show this help message and exit  
-v, --verbose   verbose mode (-vvv for more, -vvvv to enable connection  
               debugging)
```

```
[root@ansible1 tasks]#
```



12:04 / 14:13 • Creating a New Role > Scroll for details



Using roles in playbook

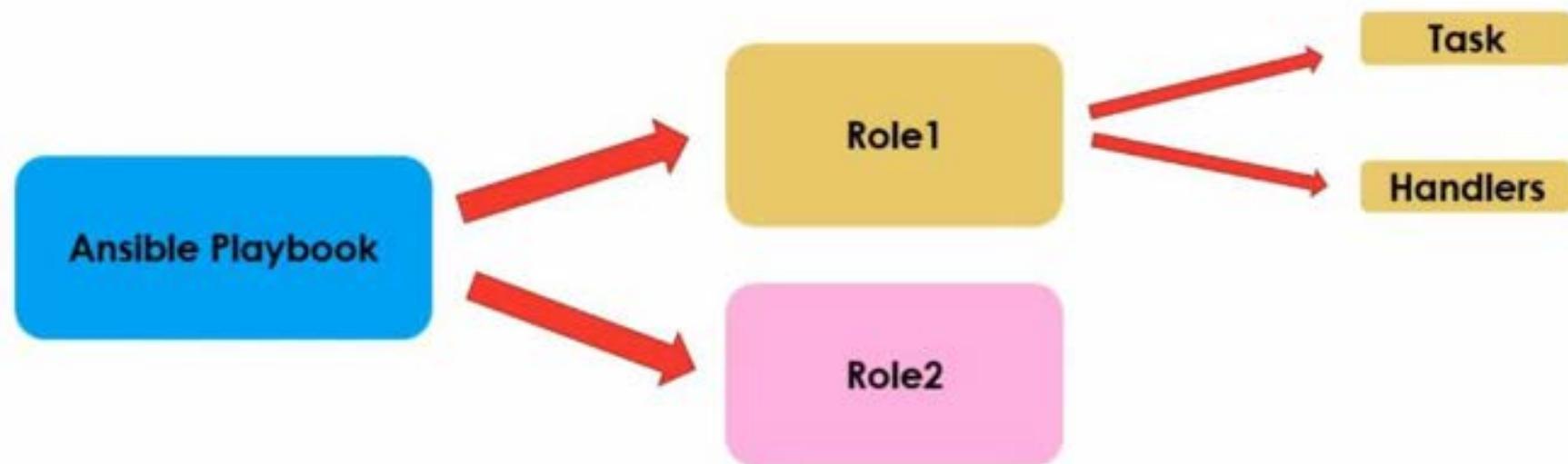
```
---  
hosts: testservers  
remote_user: root  
  
roles:  
- apache  
- mysql
```

Using roles in playbook

```
---  
hosts: testservers  
remote_user: root  
  
roles:  
- apache  
- mysql
```

Ansible Tutorial

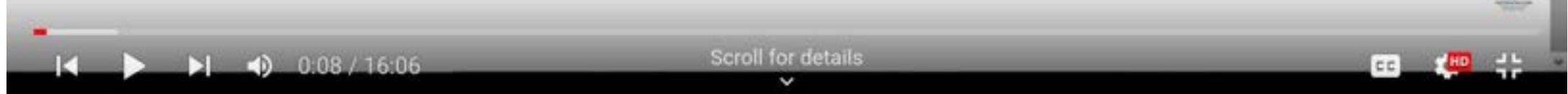
Ansible roles explained with example



Ahible Tutorial | How To Create Ansible Roles From Playbook | Ansible Playbook vs Role vs ...   

```
[root@ansible1 ansibleroles]# ls
[root@ansible1 ansibleroles]# ansible-galaxy init install
```

I



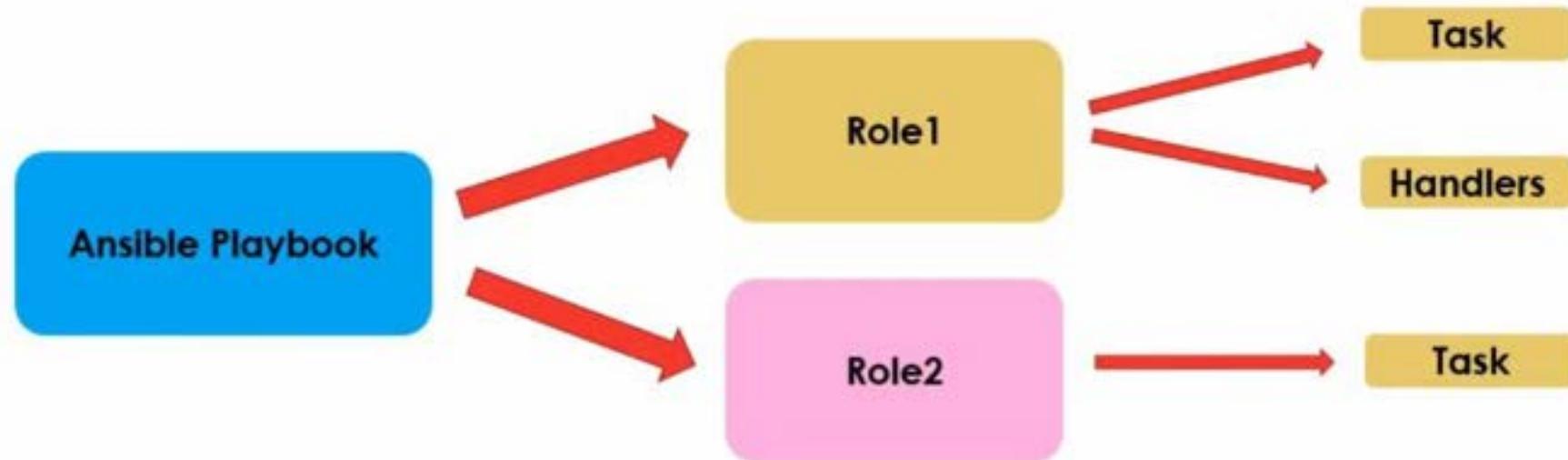
0:08 / 16:06

Scroll for details

CC HD

Ansible Tutorial

Ansible roles explained with example



Ansible Playbook vs Role vs Task



**If you haven't subscribed to my
channel, Kindly subscribe, like the
video, share and comment.**



Ansible Tutorial | How To Create Ansible Roles From Playbook | Ansible Playbook vs Role vs ...

```
File Edit Selection Find View Goto Tools Project Preferences Help
simplified lamp stack.yml x
1
2
3 - hosts: testservers
4   remote_user: root
5
6
7 tasks:
8   - name: Installing Apache, MYSQL DB Community edition & PHP
9     yum:
10       name:
11         - httpd
12         - mysql-community-server
13         - mysql-community-client
14         - php
15         - php-mysql
16       state: installed
17
18   - name: Enabling httpd service and start the service
19     service:
20       name: httpd
21       enabled: yes
22       state: started
23
24   - name: Enabling mysql service and start the service
25     service:
26       name: mysqld
27       enabled: yes
28       state: started
29
30
31   - name: Copying your php file in the /var/www/html
```

Ansible Tutorial | How To Create Ansible Roles From Playbook | Ansible Playbook vs Role vs ...

```
File Edit Selection Find View Goto Tools Project Preferences Help
simplified lamp stack.yml
10
11     name:
12         - httpd
13         - mysql-community-server
14         - mysql-community-client
15         - php
16         - php-mysql
17     state: installed
18
19     - name: Enabling httpd service and start the service
20       service:
21           name: httpd
22           enabled: yes
23           state: started
24
25     - name: Enabling mysql service and start the service
26       service:
27           name: mysqld
28           enabled: yes
29           state: started
30
31     - name: Copying your php file in the /var/www/html
32       copy:
33           src: /root/ansibleplaybook/index.php
34           dest: /var/www/html/index.php
35           force: yes
36
37     notify:
38         - restart httpd service
39
40 handlers:
```

Ansible Tutorial | How To Create Ansible Roles From Playbook | Ansible Playbook vs Role vs ...

```
File Edit Selection Find View Goto Tools Project Preferences Help
simplified lamp stack.yml x
16      state: installed
17
18      - name: Enabling httpd service and start the service
19        service:
20          name: httpd
21          enabled: yes
22          state: started
23
24      - name: Enabling mysql service and start the service
25        service:
26          name: mysqld
27          enabled: yes
28          state: started
29
30
31      - name: Copying your php file in the /var/www/html
32        copy:
33          src: /root/ansibleplaybook/index.php
34          dest: /var/www/html/index.php
35          force: yes
36
37      notify:
38          - restart httpd service
39
40  handlers:
41      - name: restart httpd service
42        service:
43          name: httpd
44          state: restarted
45
46
```

Ansible Tutorial | How To Create Ansible Roles From Playbook | Ansible Playbook vs Role vs ...

```
File Edit Selection Find View Goto Tools Project Preferences Help
simplified lamp stack.yml x
1
2
3 - hosts: testservers
4   remote_user: root
5
6
7 - tasks:
8   - name: Installing Apache, MYSQL DB Community edition & PHP
9     yum:
10    - name:
11      - httpd
12      - mysql-community-server
13      - mysql-community-client
14      - php
15      - php-mysql
16      state: installed
17
18   - name: Enabling httpd service and start the service
19     service:
20     - name: httpd
21     enabled: yes
22     state: started
23
24   - name: Enabling mysql service and start the service
25     service:
26     - name: mysqld
27     enabled: yes
28     state: started
29
30
31   - name: Copying your php file in the /var/www/html
```

Ansible Tutorial | How To Create Ansible Roles From Playbook | Ansible Playbook vs Role vs ...

```
File Edit Selection Find View Goto Tools Project Preferences Help
simplified lamp stack.yml *
25
26     service:
27         name: mysqld
28         enabled: yes
29         state: started
30
31     - name: Copying your php file in the /var/www/html
32       copy:
33         src: /root/ansibleplaybook/index.php
34         dest: /var/www/html/index.php
35         force: yes
36
37     - notify:
38         - restart httpd service
39
40     handlers:
41         - name: restart httpd service
42             service:
43                 name: httpd
44                 state: restarted
45
46
47
```

Ansible Tutorial | How To Create Ansible Roles From Playbook | Ansible Playbook vs Role vs ...

```
File Edit Selection Find View Goto Tools Project Preferences Help
simplified lamp stack.yml x
25
26   service:
27     name: mysqld
28     enabled: yes
29     state: started
30
31   - name: Copying your php file in the /var/www/html
32     copy:
33       src: /root/ansibleplaybook/index.php
34       dest: /var/www/html/index.php
35       force: yes
36
37     notify:
38       - restart httpd service
39
40   handlers:
41     - name: restart httpd service
42       service:
43         name: httpd
44         state: restarted
45
46
47
```

Ahible Tutorial | How To Create Ansible Roles From Playbook | Ansible Playbook vs Role vs ...

```
[root@ansible1 ansibleroles]# ls
[root@ansible1 ansibleroles]# ansible-galaxy init install
- Role install was created successfully
[root@ansible1 ansibleroles]#
```

I

Ahumble Tutorial | How To Create Ansible Roles From Playbook | Ansible Playbook vs Role vs ...



```
[root@ansible1 ansibleroles]# ls
[root@ansible1 ansibleroles]# ansible-galaxy init install
- Role install was created successfully
[root@ansible1 ansibleroles]# ls
install
[root@ansible1 ansibleroles]# ansible-galaxy init copy
- Role copy was created successfully
[root@ansible1 ansibleroles]# ls
copy install
[root@ansible1 ansibleroles]# cd install/
[root@ansible1 install]# ls
defaults files handlers meta README.md tasks templates tests vars
[root@ansible1 install]#
```

I

Ahible Tutorial | How To Create Ansible Roles From Playbook | Ansible Playbook vs Role vs ...

```
[root@ansible1 ansibleroles]# ls
[root@ansible1 ansibleroles]# ansible-galaxy init install
- Role install was created successfully
[root@ansible1 ansibleroles]# ls
install
[root@ansible1 ansibleroles]# ansible-galaxy init copy
- Role copy was created successfully
[root@ansible1 ansibleroles]# ls
copy install
[root@ansible1 ansibleroles]# cd install/
[root@ansible1 install]# ls
defaults files handlers meta README.md tasks templates tests vars
[root@ansible1 install]# cd tasks/
[root@ansible1 tasks]# ls
main.yml
[root@ansible1 tasks]#
```

Ahumble Tutorial | How To Create Ansible Roles From Playbook | Ansible Playbook vs Role vs ...



```
[root@ansible1 ansibleroles]# ls
[root@ansible1 ansibleroles]# ansible-galaxy init install
- Role install was created successfully
[root@ansible1 ansibleroles]# ls
install
[root@ansible1 ansibleroles]# ansible-galaxy init copy
- Role copy was created successfully
[root@ansible1 ansibleroles]# ls
copy install
[root@ansible1 ansibleroles]# cd install/
[root@ansible1 install]# ls
defaults files handlers meta README.md tasks templates tests vars
[root@ansible1 install]# cd tasks/
[root@ansible1 tasks]# ls
main.yml
[root@ansible1 tasks]# cat main.yml
---
# tasks file for install[root@ansible1 tasks]# █
```

Ansible Tutorial | How To Create Ansible Roles From Playbook | Ansible Playbook vs Role vs ...

```
File Edit Selection Find View Goto Tools Project Preferences Help
simplified lamp stack.yml x
25
26   service:
27     name: mysqld
28     enabled: yes
29     state: started
30
31   - name: Copying your php file in the /var/www/html
32     copy:
33       src: /root/ansibleplaybook/index.php
34       dest: /var/www/html/index.php
35       force: yes
36
37     notify:
38       - restart httpd service
39
40   handlers:
41     - name: restart httpd service
42       service:
43         name: httpd
44         state: restarted
45
46
47
```

Ansible Tutorial | How To Create Ansible Roles From Playbook | Ansible Playbook vs Role vs ...

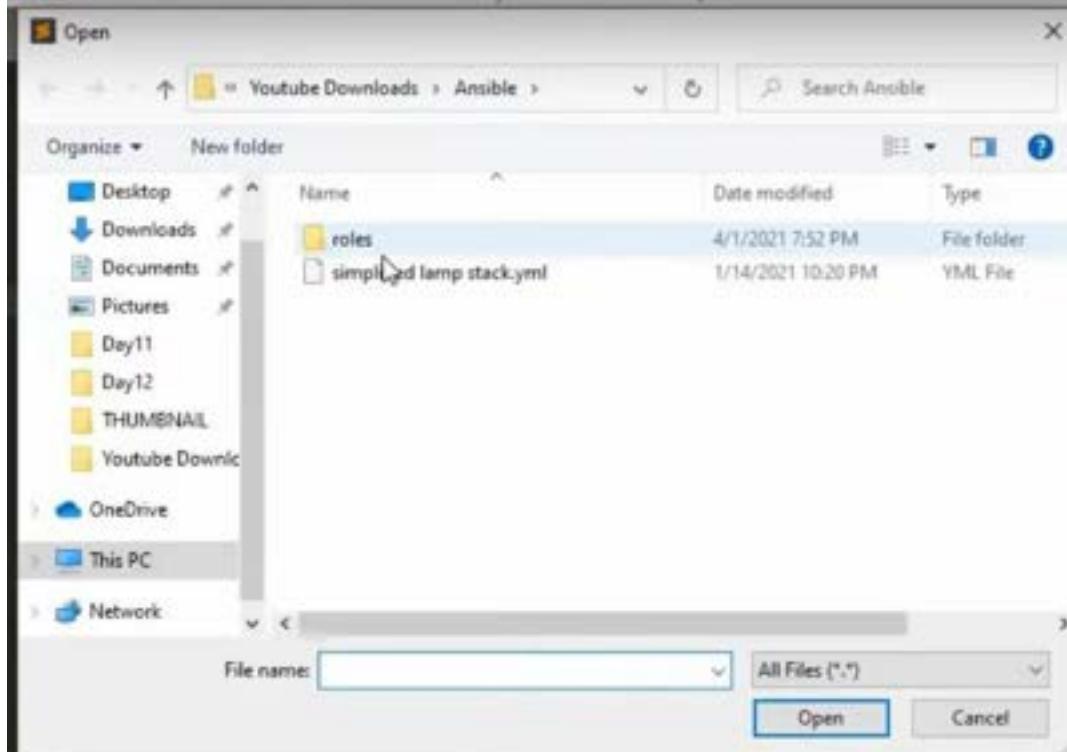
The screenshot shows a video player interface with a dark theme. The main content area displays a portion of an Ansible playbook. The visible code is:

```
 41:     -> php file in the /var/www/html
 42:     roleplaybook/index.php
 43:     html/index.php
 44:     httpd service
 45:
 46:     state: restarted
 47:
```

The video player has a progress bar at the bottom left, showing 5:00 / 16:06. At the bottom right, there are icons for closed captions (CC), HD resolution, and volume control.

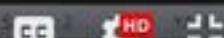
A context menu is open over the first line of the playbook, listing various file operations like 'New File', 'Open File...', 'Save', etc., along with their keyboard shortcuts.

Ansible Tutorial | How To Create Ansible Roles From Playbook | Ansible Playbook vs Role vs ...

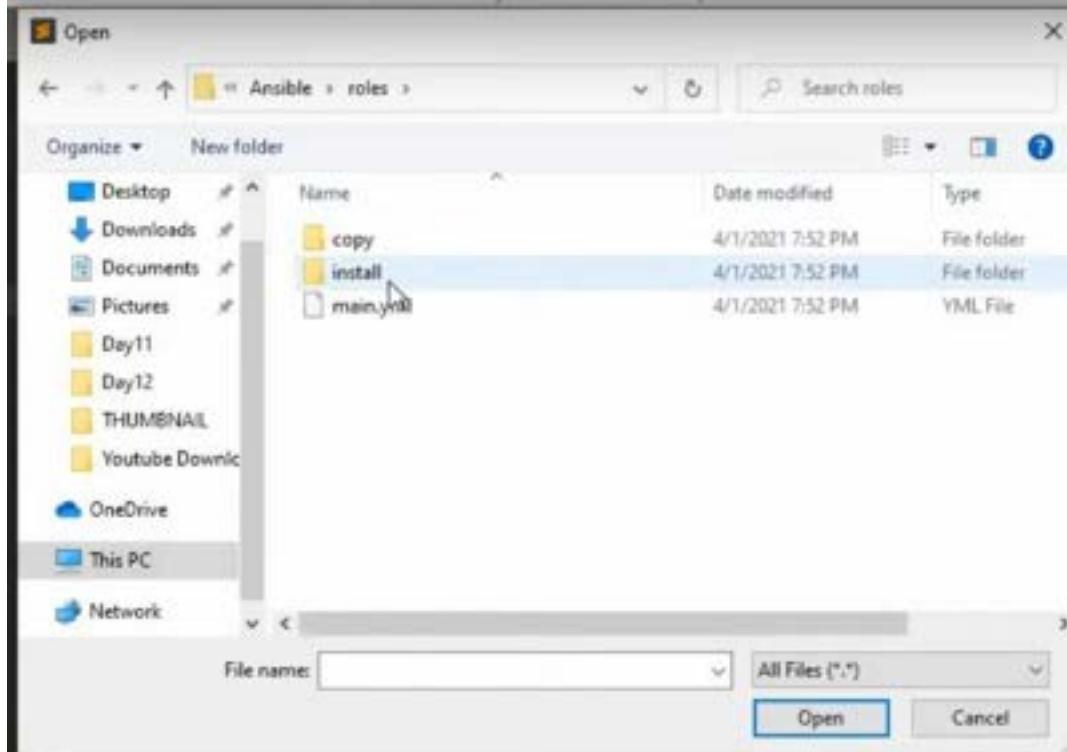


47

Scroll for details



Ansible Tutorial | How To Create Ansible Roles From Playbook | Ansible Playbook vs Role vs ...

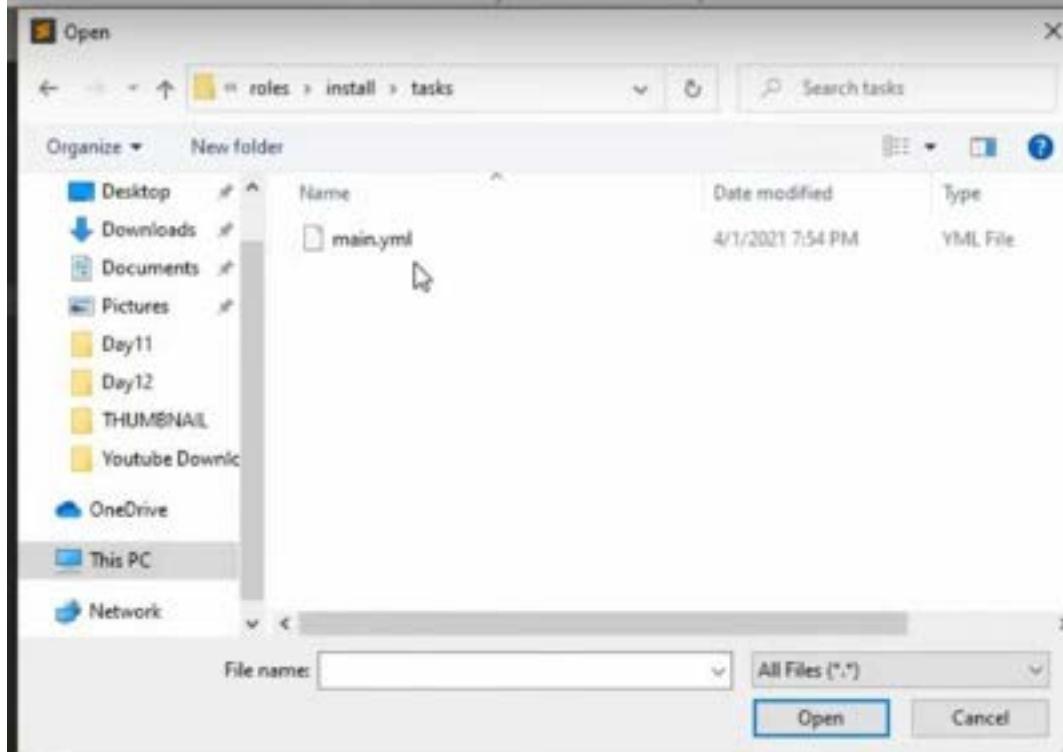


47

Scroll for details



Ansible Tutorial | How To Create Ansible Roles From Playbook | Ansible Playbook vs Role vs ...



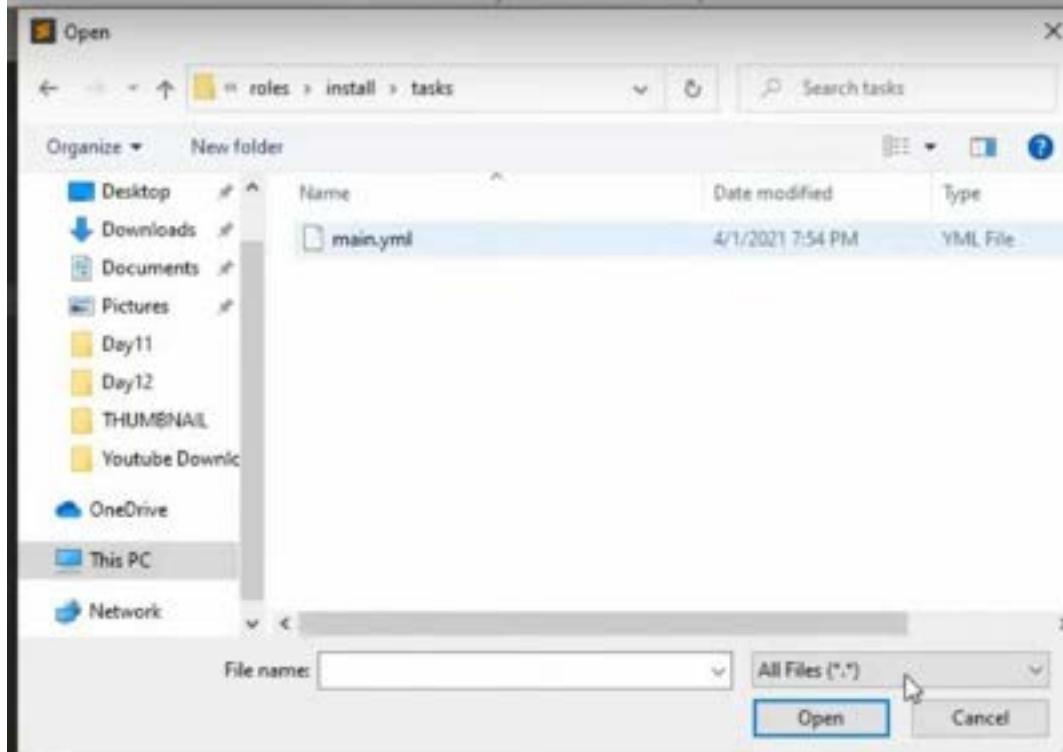
47

◀ ▶ 🔍 5:07 / 16:06

Scroll for details



Ansible Tutorial | How To Create Ansible Roles From Playbook | Ansible Playbook vs Role vs ...



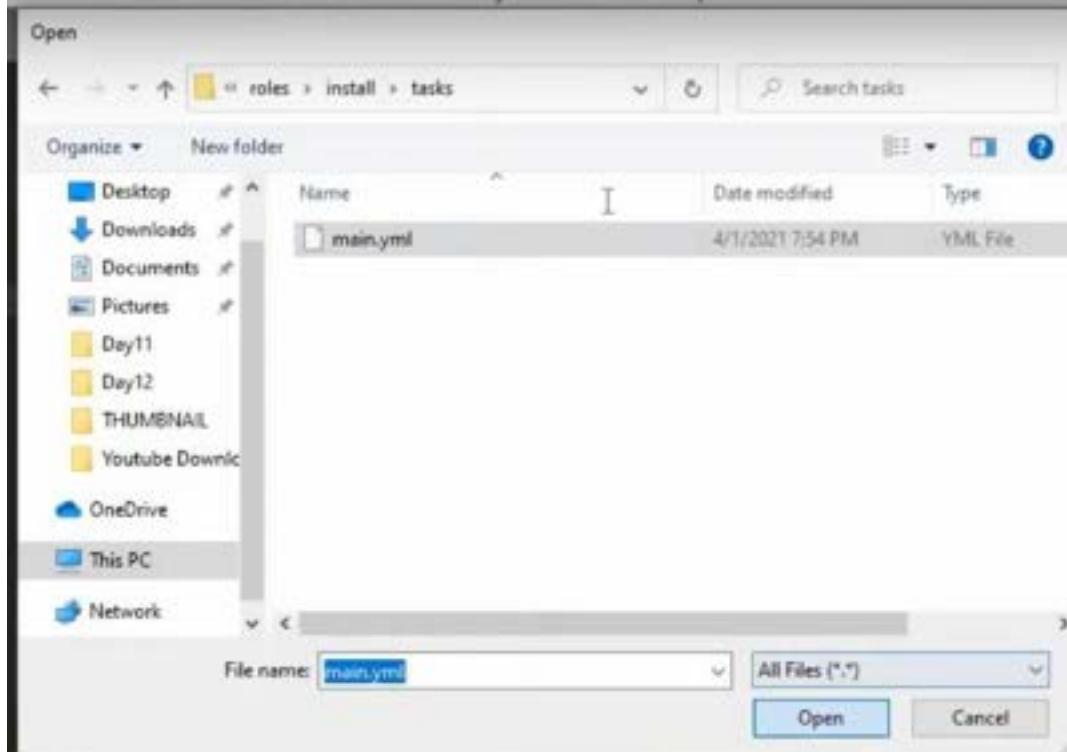
47

◀ ▶ ⏪ ⏩ 5:08 / 16:06

Scroll for details

CC HD

Ansible Tutorial | How To Create Ansible Roles From Playbook | Ansible Playbook vs Role vs ...



47

Scroll for details

5:10 / 16:06

CC HD

Ansible Tutorial | How To Create Ansible Roles From Playbook | Ansible Playbook vs Role vs ...

```
1
2
3 - hosts: testservers
4   remote_user: root
5
6
7 tasks:
8   - name: Installing Apache, MYSQL DB Community edition & PHP
9     yum:
10       name:
11         - httpd
12         - mysql-community-server
13         - mysql-community-client
14         - php
15         - php-mysql
16       state: installed
17
18   - name: Enabling httpd service and start the service
19     service:
20       name: httpd
21       enabled: yes
22       state: started
23
24   - name: Enabling mysql service and start the service
25     service:
26       name: mysqld
27       enabled: yes
28       state: started
29
30
31   - name: Copying your php file in the /var/www/html
```

Ansible Tutorial | How To Create Ansible Roles From Playbook | Ansible Playbook vs Role vs ...

```
1
2
3 - hosts: testservers
4   remote_user: root
5
6
7 tasks:
8   - name: Installing Apache, MYSQL DB Community edition & PHP
9     yum:
10    name:
11      - httpd
12      - mysql-community-server
13      - mysql-community-client
14      - php
15      - php-mysql
16    state: installed
17
18   - name: Enabling httpd service and start the service
19     service:
20       name: httpd
21       enabled: yes
22       state: started
23
24   - name: Enabling mysql service and start the service
25     service:
26       name: mysqld
27       enabled: yes
28       state: started
29
30
31   - name: Copying your php file in the /var/www/html
```

Ansible Tutorial | How To Create Ansible Roles From Playbook | Ansible Playbook vs Role vs ...

```
1
2
3 - hosts: testservers
4   remote_user: root
5
6
7 tasks:
8   - name: Installing Apache, MYSQL DB Community edition & PHP
9     yum:
10    name:
11      - httpd
12      - mysql-community-server
13      - mysql-community-client
14      - php
15      - php-mysql
16    state: installed
17
18   - name: Enabling httpd service and start the service
19     service:
20       name: httpd
21       enabled: yes
22       state: started
23
24   - name: Enabling mysql service and start the service
25     service:
26       name: mysqld
27       enabled: yes
28       state: started
29
30   - name: Copying your php file in the /var/www/html
```

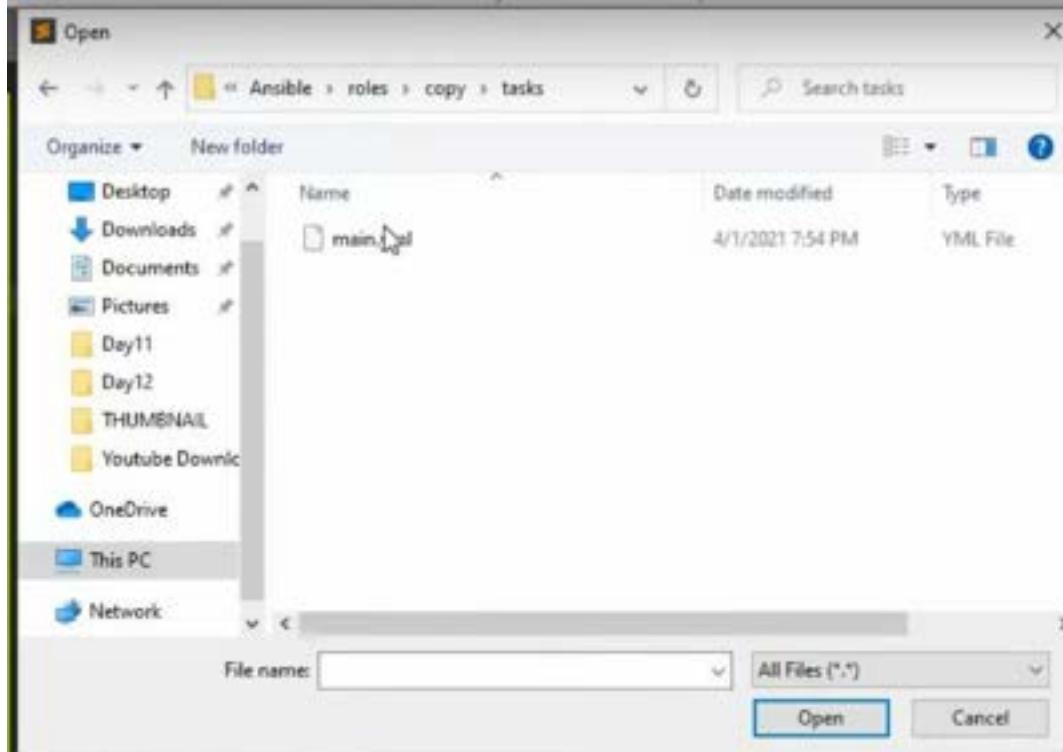
Ansible Tutorial | How To Create Ansible Roles From Playbook | Ansible Playbook vs Role vs ...

```
File Edit Selection Find View Goto Tools Project Preferences Help
simplified_lamp_stack.yml main.yml
1 ---| main.yml
2
3 - name: Installing Apache, MySQL DB Community edition & PHP
4   yum:
5     name:
6       - httpd
7       - mysql-community-server
8       - mysql-community-client
9       - php
10      - php-mysql
11      state: installed
12
13 - name: Enabling httpd service and start the service
14   service:
15     name: httpd
16     enabled: yes
17     state: started
18
19 - name: Enabling mysql service and start the service
20   service:
21     name: mysqld
22     enabled: yes
23     state: started
```

Ansible Tutorial | How To Create Ansible Roles From Playbook | Ansible Playbook vs Role vs ...

```
1 ---
2   - name: Installing Apache, MYSQL DB Community edition & PHP
3     yum:
4       name:
5         - httpd
6         - mysql-community-server
7         - mysql-community-client
8         - php
9         - php-mysql
10      state: installed
11
12   - name: Enabling httpd service and start the service
13     service:
14       name: httpd
15       enabled: yes
16       state: started
17
18   - name: Enabling mysql service and start the service
19     service:
20       name: mysqld
21       enabled: yes
22       state: started
23
```

Ansible Tutorial | How To Create Ansible Roles From Playbook | Ansible Playbook vs Role vs ...



23 state: started

Scroll for details

◀ ▶ ⏪ ⏩ 🔍 7:00 / 16:06

CC HD

Ansible Tutorial | How To Create Ansible Roles From Playbook | Ansible Playbook vs Role vs ...

```
22      state: started
23
24      - name: Enabling mysql service and start the service
25        service:
26          name: mysqld
27          enabled: yes
28          state: started
29
30
31      - name: Copying your php file in the /var/www/html
32        copy:
33          src: /root/ansibleplaybook/index.php
34          dest: /var/www/html/index.php
35          force: yes
36
37      - notify:
38        - restart httpd service
39
40    handlers:
41      - name: restart httpd service
42        service:
43          name: httpd
44          state: restarted
45
46
47
```

Ansible Tutorial | How To Create Ansible Roles From Playbook | Ansible Playbook vs Role vs ...

```
1 ---
2   - name: Copying your php file in the /var/www/html
3     copy:
4       src: /root/ansibleplaybook/index.php
5       dest: /var/www/html/index.php
6       force: yes
7
8   notify:
9     - restart httpd service
```

Ansible Tutorial | How To Create Ansible Roles From Playbook | Ansible Playbook vs Role vs ...

```
1 ---  
2 - name: Copying your php file in the /var/www/html  
3 copy:  
4   src: /root/ansibleplaybook/index.php  
5   dest: /var/www/html/index.php  
6   force: yes  
7 notify:  
8   - restart httpd service
```

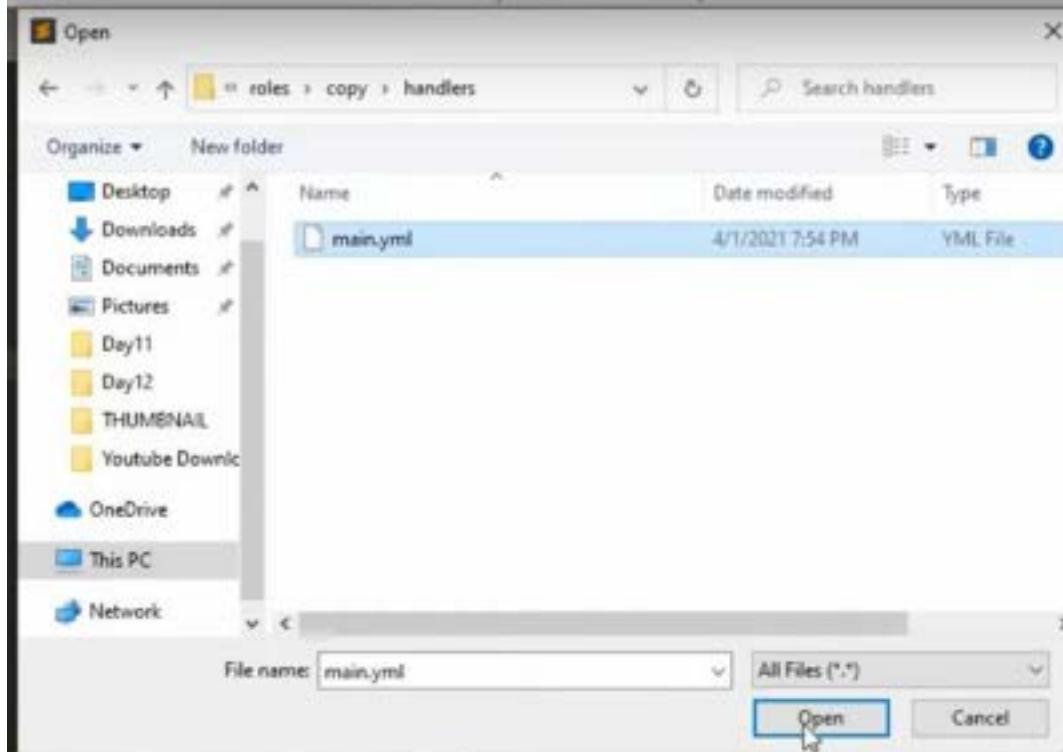
Ansible Tutorial | How To Create Ansible Roles From Playbook | Ansible Playbook vs Role vs ...

```
1 ---
2
3 - name: Installing Apache, MYSQL DB Community edition & PHP
4   yum:
5     name:
6       - httpd
7       - mysql-community-server
8       - mysql-community-client
9       - php
10      - php-mysql
11    state: installed
12
13 - name: Enabling httpd service and start the service
14   service:
15     name: httpd
16     enabled: yes
17     state: started
18
19 - name: Enabling mysql service and start the service
20   service:
21     name: mysqld
22     enabled: yes
23     state: started
```

Ansible Tutorial | How To Create Ansible Roles From Playbook | Ansible Playbook vs Role vs ...

```
File Edit Selection Find View Goto Tools Project Preferences Help
simplified lamp stack.yml main.yml -- instances copyTasks
10
11     name:
12         - httpd
13         - mysql-community-server
14         - mysql-community-client
15         - php
16         - php-mysql
17     state: installed
18
19     - name: Enabling httpd service and start the service
20       service:
21           name: httpd
22           enabled: yes
23           state: started
24
25     - name: Enabling mysql service and start the service
26       service:
27           name: mysqld
28           enabled: yes
29           state: started
30
31     - name: Copying your php file in the /var/www/html
32       copy:
33           src: /root/ansibleplaybook/index.php
34           dest: /var/www/html/index.php
35           force: yes
36
37       notify:
38           - restart httpd service
39
40   handlers:
```

Ansible Tutorial | How To Create Ansible Roles From Playbook | Ansible Playbook vs Role vs ...



```
44      state: restarted
45
46
47
```

8:23 / 16:06

Scroll for details



Ansible Tutorial | How To Create Ansible Roles From Playbook | Ansible Playbook vs Role vs ...

```
File Edit Selection Find View Goto Tools Project Preferences Help
simplified lamp stack.yml main.yml -- copy handlers main.yml -- install tasks main.yml -- copy tasks
22     state: started
23
24 - name: Enabling mysql service and start the service
25   service:
26     name: mysqld
27     enabled: yes
28     state: started
29
30
31 - name: Copying your php file in the /var/www/html
32   copy:
33     src: /root/ansibleplaybook/index.php
34     dest: /var/www/html/index.php
35     force: yes
36
37   notify:
38     - restart httpd service
39
40 handlers:
41   - name: restart httpd service
42     service:
43       name: httpd
44       state: restarted
45
46
47
```

Ansible Tutorial | How To Create Ansible Roles From Playbook | Ansible Playbook vs Role vs ...

```
1 ---
2
3 - name: restart httpd service
4   service:
5     name: httpd
6     state: restarted
```

Ansible Tutorial | How To Create Ansible Roles From Playbook | Ansible Playbook vs Role vs ...

```
File Edit Selection Find View Goto Tools Project Preferences Help
simplified lamp stack.yml X main.yml -- copy handlers X main.yml -- install tasks X main.yml -- copy tasks X
4
5
6
7 tasks:
8   - name: Installing Apache, MYSQL DB Community edition & PHP
9     yum:
10       name:
11         - httpd
12         - mysql-community-server
13         - mysql-community-client
14         - php
15         - php-mysql
16       state: installed
17
18   - name: Enabling httpd service and start the service
19     service:
20       name: httpd
21       enabled: yes
22       state: started
23
24   - name: Enabling mysql service and start the service
25     service:
26       name: mysqld
27       enabled: yes
28       state: started
29
30
31   - name: Copying your php file in the /var/www/html
32     copy:
33       src: /root/ansibleplaybook/index.php
34       dest: /var/www/html/index.php
```

Ansible Tutorial | How To Create Ansible Roles From Playbook | Ansible Playbook vs Role vs ...

```
File Edit Selection Find View Goto Tools Project Preferences Help
simplified lamp stack.yml main.yml -- copy files main.yml -- install tasks main.yml -- copy tasks
1
2
3 - hosts: testservers
4   remote_user: root
5
6
7 tasks:
8
9
10
11
```

Ansible Tutorial | How To Create Ansible Roles From Playbook | Ansible Playbook vs Role vs ...

```
File Edit Selection Find View Goto Tools Project Preferences Help
simplified lamp stack.yml * main.yml -- copy handlers * main.yml -- install tasks * main.yml -- copy tasks *
1
2
3 - hosts: testservers
4   remote_user: root
5
6
7   roles:
8     - install
9     - copy
10
11
12
13
```

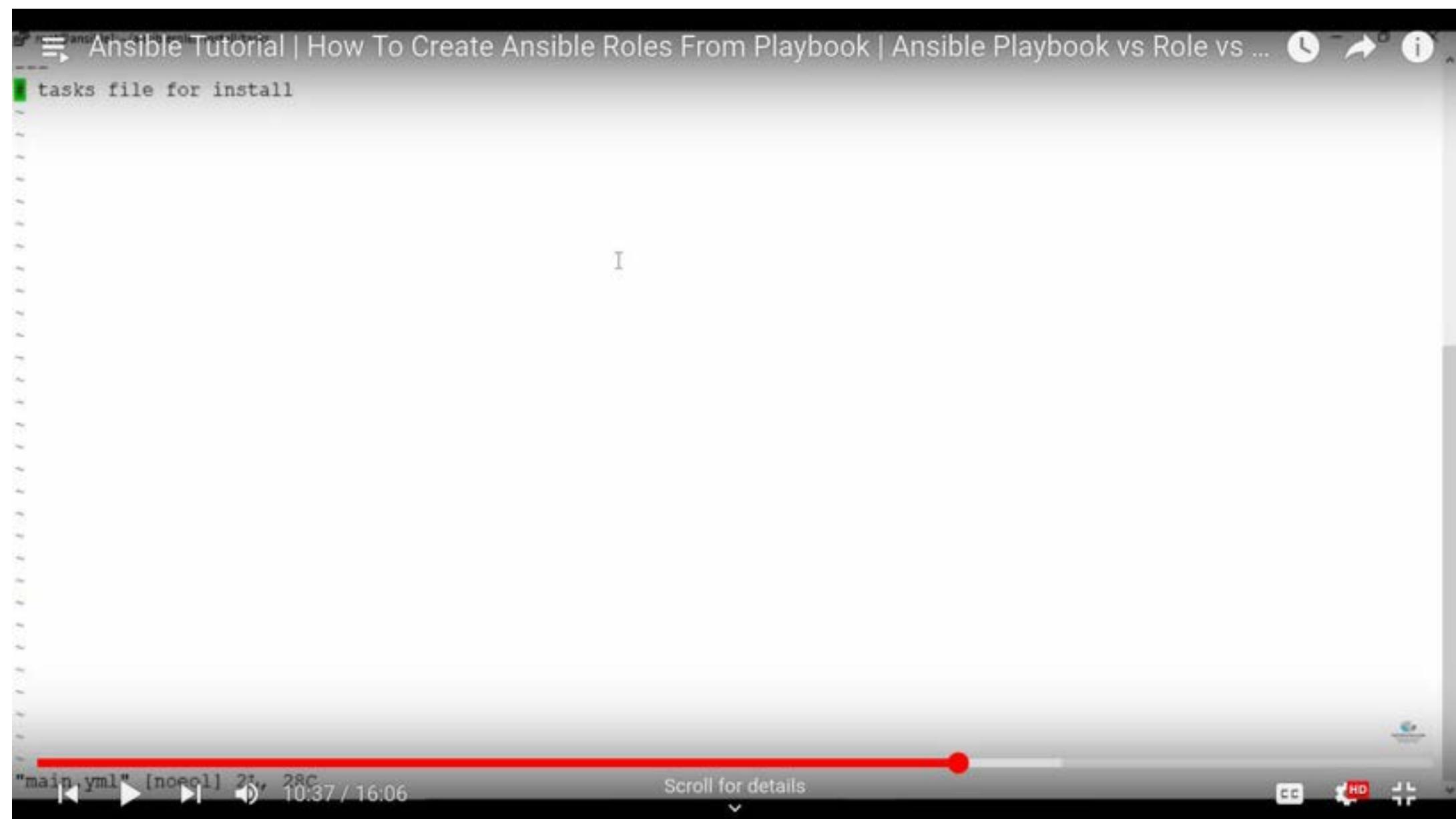
Ahible Tutorial | How To Create Ansible Roles From Playbook | Ansible Playbook vs Role vs ...

[root@ansible1 tasks]# ls

main.yml

[root@ansible1 tasks]# vi main.yml

I



Ansible Tutorial | How To Create Ansible Roles From Playbook | Ansible Playbook vs Role vs ...

```
---  
I  
- name: Installing Apache, MYSQL DB Community edition & PHP  
  yum:  
    name:  
      - httpd  
      - mysql-community-server  
      - mysql-community-client  
      - php  
      - php-mysql  
    state: installed  
  
- name: Enabling httpd service and start the service  
  service:  
    name: httpd  
    enabled: yes  
    state: started  
  
- name: Enabling mysql service and start the service  
  service:  
    name: mysqld  
    enabled: yes  
    state: started
```

10:46 / 16:06

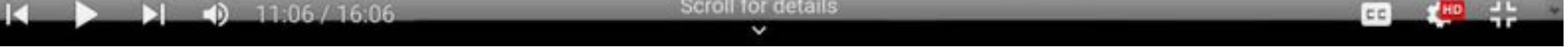
Scroll for details

Ahnsible Tutorial | How To Create Ansible Roles From Playbook | Ansible Playbook vs Role vs ...   

```
[root@ansible1 tasks]# ls
main.yml
[root@ansible1 tasks]# vi main.yml
[root@ansible1 tasks]# cd ..
[root@ansible1 install]# ls
defaults files handlers meta README.md tasks templates tests vars
[root@ansible1 install]#
```

Ahible Tutorial | How To Create Ansible Roles From Playbook | Ansible Playbook vs Role vs ...   

```
[root@ansible1 tasks]# ls
main.yml
[root@ansible1 tasks]# vi main.yml
[root@ansible1 tasks]# cd ..
[root@ansible1 install]# ls
defaults files handlers meta README.md tasks templates tests vars
[root@ansible1 install]# cd ..
[root@ansible1 ansibleroles]# ls
copy install
[root@ansible1 ansibleroles]# c
```



11:06 / 16:06

Scroll for details

Ahumble Tutorial | How To Create Ansible Roles From Playbook | Ansible Playbook vs Role vs ...   

```
[root@ansible1 tasks]# ls
main.yml
[root@ansible1 tasks]# vi main.yml
[root@ansible1 tasks]# cd ..
[root@ansible1 install]# ls
defaults files handlers meta README.md tasks templates tests vars
[root@ansible1 install]# cd ..
[root@ansible1 ansibleroles]# ls
copy install
[root@ansible1 ansibleroles]# cd copy/
[root@ansible1 copy]# ls
defaults files handlers meta README.md tasks templates tests vars
[root@ansible1 copy]# cd ..
```

AhumbleTutorial | How To Create Ansible Roles From Playbook | Ansible Playbook vs Role vs ...   

```
[root@ansible1 tasks]# ls
main.yml
[root@ansible1 tasks]# vi main.yml
[root@ansible1 tasks]# cd ..
[root@ansible1 install]# ls
defaults files handlers meta README.md tasks templates tests vars
[root@ansible1 install]# cd ..
[root@ansible1 ansibleroles]# ls
copy install
[root@ansible1 ansibleroles]# cd copy/
[root@ansible1 copy]# ls
defaults files handlers meta README.md tasks templates tests vars
[root@ansible1 copy]# cd tasks
[root@ansible1 tasks]# vi main.yml
```

Ansible Tutorial | How To Create Ansible Roles From Playbook | Ansible Playbook vs Role vs ...

```
# tasks file for copy
```

11:17 / 16:06

Scroll for details

CC HD

Ahsible Tutorial | How To Create Ansible Roles From Playbook | Ansible Playbook vs Role vs ...

```
# tasks file for copy
```

I

"main.yml" [noepl] 21, 25C 11:15 / 16:06

Scroll for details

CC HD

Ansible Tutorial | How To Create Ansible Roles From Playbook | Ansible Playbook vs Role vs ...



```
1 ---  
2   - name: Copying your php file in the /var/www/html  
3     copy:  
4       src: /root/ansibleplaybook/index.php  
5       dest: /var/www/html/index.php  
6       force: yes  
7     notify:  
8       - restart httpd service
```

Scroll for details

◀ ▶ ⏪ ⏩ 🔍 11:28 / 16:06

CC HD

Ahsible Tutorial | How To Create Ansible Roles From Playbook | Ansible Playbook vs Role vs ...



```
---
```

```
- name: Copying your php file in the /var/www/html
  copy:
    src: /root/ansibleplaybook/index.php
    dest: /var/www/html/index.php
    force: yes
  notify:
    - restart httpd service
```

Ahumble Tutorial | How To Create Ansible Roles From Playbook | Ansible Playbook vs Role vs ...   

```
[root@ansible1 tasks]# ls
main.yml
[root@ansible1 tasks]# vi main.yml
[root@ansible1 tasks]# cd ..
[root@ansible1 install]# ls
defaults files handlers meta README.md tasks templates tests vars
[root@ansible1 install]# cd ..
[root@ansible1 ansibleroles]# ls
copy install
[root@ansible1 ansibleroles]# cd copy/
[root@ansible1 copy]# ls
defaults files handlers meta README.md tasks templates tests vars
[root@ansible1 copy]# cd tasks
[root@ansible1 tasks]# vi main.yml
[root@ansible1 tasks]# ls
main.yml
[root@ansible1 tasks]# cd ..
[root@ansible1 copy]# 
```

Ahumble Tutorial | How To Create Ansible Roles From Playbook | Ansible Playbook vs Role vs ...   

```
[root@ansible1 tasks]# ls
main.yml
[root@ansible1 tasks]# vi main.yml
[root@ansible1 tasks]# cd ..
[root@ansible1 install]# ls
defaults files handlers meta README.md tasks templates tests vars
[root@ansible1 install]# cd ..
[root@ansible1 ansibleroles]# ls
copy install
[root@ansible1 ansibleroles]# cd copy/
[root@ansible1 copy]# ls
defaults files handlers meta README.md tasks templates tests vars
[root@ansible1 copy]# cd tasks
[root@ansible1 tasks]# vi main.yml
[root@ansible1 tasks]# ls
main.yml
[root@ansible1 tasks]# cd ..
[root@ansible1 copy]# cd handlers/
[root@ansible1 handlers]# ls
main.yml
[root@ansible1 handlers]#
```

Ahumble Tutorial | How To Create Ansible Roles From Playbook | Ansible Playbook vs Role vs ...   

```
[root@ansible1 tasks]# ls
main.yml
[root@ansible1 tasks]# vi main.yml
[root@ansible1 tasks]# cd ..
[root@ansible1 install]# ls
defaults files handlers meta README.md tasks templates tests vars
[root@ansible1 install]# cd ..
[root@ansible1 ansibleroles]# ls
copy install
[root@ansible1 ansibleroles]# cd copy/
[root@ansible1 copy]# ls
defaults files handlers meta README.md tasks templates tests vars
[root@ansible1 copy]# cd tasks
[root@ansible1 tasks]# vi main.yml
[root@ansible1 tasks]# ls
main.yml
[root@ansible1 tasks]# cd ..
[root@ansible1 copy]# cd handlers/
[root@ansible1 handlers]# ls
main.yml
[root@ansible1 handlers]# vi main.yml
```

Ahsible Tutorial | How To Create Ansible Roles From Playbook | Ansible Playbook vs Role vs ...

```
# handlers file for copy
```

"main.yml" [noepl] 21, 28C 11:48 / 16:06

Scroll for details

CC HD

Ansible Tutorial | How To Create Ansible Roles From Playbook | Ansible Playbook vs Role vs ...



```
1 ...
2
3 - name: restart httpd service
4   service:
5     name: httpd
6     state: restarted
```

◀ ▶ ⏪ ⏩ 🔍 11:59 / 16:06

Scroll for details

CC HD

Ansible Tutorial | How To Create Ansible Roles From Playbook | Ansible Playbook vs Role vs ...

```
- name: restart httpd service
  service:
    name: httpd
    state: restarted
```

12:01 / 16:06

Scroll for details

AhumbleTutorial | How To Create Ansible Roles From Playbook | Ansible Playbook vs Role vs ...   

```
[root@ansible1 tasks]# ls
main.yml
[root@ansible1 tasks]# vi main.yml
[root@ansible1 tasks]# cd ..
[root@ansible1 install]# ls
defaults files handlers meta README.md tasks templates tests vars
[root@ansible1 install]# cd ..
[root@ansible1 ansibleroles]# ls
copy install
[root@ansible1 ansibleroles]# cd copy/
[root@ansible1 copy]# ls
defaults files handlers meta README.md tasks templates tests vars
[root@ansible1 copy]# cd tasks
[root@ansible1 tasks]# vi main.yml
[root@ansible1 tasks]# ls
main.yml
[root@ansible1 tasks]# cd ..
[root@ansible1 copy]# cd handlers/
[root@ansible1 handlers]# ls
main.yml
[root@ansible1 handlers]# vi main.yml
[root@ansible1 handlers]# cd ..
[root@ansible1 copy]# ls
defaults files handlers meta README.md tasks templates tests vars
[root@ansible1 copy]# cd ..
```

Ahible Tutorial | How To Create Ansible Roles From Playbook | Ansible Playbook vs Role vs ...   

```
[root@ansible1 tasks]# ls
main.yml
[root@ansible1 tasks]# vi main.yml
[root@ansible1 tasks]# cd ..
[root@ansible1 install]# ls
defaults files handlers meta README.md tasks templates tests vars
[root@ansible1 install]# cd ..
[root@ansible1 ansibleroles]# ls
copy install
[root@ansible1 ansibleroles]# cd copy/
[root@ansible1 copy]# ls
defaults files handlers meta README.md tasks templates tests vars
[root@ansible1 copy]# cd tasks
[root@ansible1 tasks]# vi main.yml
[root@ansible1 tasks]# ls
main.yml
[root@ansible1 tasks]# cd ..
[root@ansible1 copy]# cd handlers/
[root@ansible1 handlers]# ls
main.yml
[root@ansible1 handlers]# vi main.yml
[root@ansible1 handlers]# cd ..
[root@ansible1 copy]# ls
defaults files handlers meta README.md tasks templates tests vars
[root@ansible1 copy]# cd ..
[root@ansible1 ansibleroles]# ls
copy install
[root@ansible1 ansibleroles]#
```

AhumbleTutorial | How To Create Ansible Roles From Playbook | Ansible Playbook vs Role vs ...   

```
[root@ansible1 tasks]# ls
main.yml
[root@ansible1 tasks]# vi main.yml
[root@ansible1 tasks]# cd ..
[root@ansible1 install]# ls
defaults files handlers meta README.md tasks templates tests vars
[root@ansible1 install]# cd ..
[root@ansible1 ansibleroles]# ls
copy install
[root@ansible1 ansibleroles]# cd copy/
[root@ansible1 copy]# ls
defaults files handlers meta README.md tasks templates tests vars
[root@ansible1 copy]# cd tasks
[root@ansible1 tasks]# vi main.yml
[root@ansible1 tasks]# ls
main.yml
[root@ansible1 tasks]# cd ..
[root@ansible1 copy]# cd handlers/
[root@ansible1 handlers]# ls
main.yml
[root@ansible1 handlers]# vi main.yml
[root@ansible1 handlers]# cd ..
[root@ansible1 copy]# ls
defaults files handlers meta README.md tasks templates tests vars
[root@ansible1 copy]# cd ..
[root@ansible1 ansibleroles]# ls
copy install
[root@ansible1 ansibleroles]# vi lampstack.yml
```

Ansible Tutorial | How To Create Ansible Roles From Playbook | Ansible Playbook vs Role vs ...

```
1
2
3 hosts: testservers
4 remote_user: root
5
6
7 roles:
8   - install
9   - copy
10
11
12
13
```

Ahsible Tutorial | How To Create Ansible Roles From Playbook | Ansible Playbook vs Role vs ...

```
---  
- hosts: testservers  
  remote_user: root  
  
  roles:  
    - install  
    - copy
```

12:47 / 16:06

Scroll for details

Ahsible Tutorial | How To Create Ansible Roles From Playbook | Ansible Playbook vs Role vs ...

```
[root@ansible1 tasks]# ls
main.yml
[root@ansible1 tasks]# vi main.yml
[root@ansible1 tasks]# cd ..
[root@ansible1 install]# ls
defaults files handlers meta README.md tasks templates tests vars
[root@ansible1 install]# cd ..
[root@ansible1 ansibleroles]# ls
copy install
[root@ansible1 ansibleroles]# cd copy/
[root@ansible1 copy]# ls
defaults files handlers meta README.md tasks templates tests vars
[root@ansible1 copy]# cd tasks
[root@ansible1 tasks]# vi main.yml
[root@ansible1 tasks]# ls
main.yml
[root@ansible1 tasks]# cd ..
[root@ansible1 copy]# cd handlers/
[root@ansible1 handlers]# ls
main.yml
[root@ansible1 handlers]# vi main.yml
[root@ansible1 handlers]# cd ..
[root@ansible1 copy]# ls
defaults files handlers meta README.md tasks templates tests vars
[root@ansible1 copy]# cd ..
[root@ansible1 ansibleroles]# ls
copy install
[root@ansible1 ansibleroles]# vi lampstack.yml
[root@ansible1 ansibleroles]# ls
copy install lampstack.yml
[root@ansible1 ansibleroles]#
```

Ansible Tutorial | How To Create Ansible Roles From Playbook | Ansible Playbook vs Role vs ...

[root@ansible1 ansibleroles]# ansible-playbook lampstack.yml



Ahsible Tutorial | How To Create Ansible Roles From Playbook | Ansible Playbook vs Role vs ...

```
[root@ansible1 ansibleroles]# ansible-playbook lampstack.yml

PLAY [testservers] *****

TASK [Gathering Facts] *****
ok: [192.168.145.146]

TASK [install : Installing Apache, MYSQL DB Community edition & PHP] *****
changed: [192.168.145.146]

TASK [install : Enabling httpd service and start the service] *****
changed: [192.168.145.146]

TASK [install : Enabling mysql service and start the service] *****
changed: [192.168.145.146]

TASK [copy : Copying your php file in the /var/www/html] *****
changed: [192.168.145.146]

RUNNING HANDLER [copy : restart httpd service] *****
changed: [192.168.145.146]

PLAY RECAP *****
192.168.145.146      : ok=6    changed=5    unreachable=0    failed=0    skipped=0    rescued=0    ignored=0

[root@ansible1 ansibleroles]#
```

Ahnsible Tutorial | How To Create Ansible Roles From Playbook | Ansible Playbook vs Role vs ...

```
[root@ansible1 ansibleroles]# ansible-playbook lampstack.yml

PLAY [testservers] *****

TASK [Gathering Facts] *****
ok: [192.168.145.146]

TASK [install : Installing Apache, MYSQL DB Community edition & PHP] *****
changed: [192.168.145.146]

TASK [install : Enabling httpd service and start the service] *****
changed: [192.168.145.146]

TASK [install : Enabling mysql service and start the service] *****
changed: [192.168.145.146]

TASK [copy : Copying your php file in the /var/www/html] *****
changed: [192.168.145.146]

RUNNING HANDLER [copy : restart httpd service] *****
changed: [192.168.145.146] I

PLAY RECAP *****
192.168.145.146      : ok=6    changed=5    unreachable=0    failed=0    skipped=0    rescued=0    ignored=0

[root@ansible1 ansibleroles]# ansible-playbook lampstack.yml

PLAY [testservers] *****

TASK [Gathering Facts] *****
```

14:22 / 16:06

CC HD

Ahsible Tutorial | How To Create Ansible Roles From Playbook | Ansible Playbook vs Role vs ...

```
TASK [copy : Copying your php file in the /var/www/html] ****
changed: [192.168.145.146]

RUNNING HANDLER [copy : restart httpd service] ****
changed: [192.168.145.146]

PLAY RECAP ****
192.168.145.146      : ok=6    changed=5    unreachable=0    failed=0    skipped=0    rescued=0    ignored=0

[root@ansible1 ansibleroles]# ansible-playbook lampstack.yml

PLAY [testservers] ****

TASK [Gathering Facts] ****
ok: [192.168.145.146]

TASK [install : Installing Apache, MYSQL DB Community edition & PHP] ****
ok: [192.168.145.146]

TASK [install : Enabling httpd service and start the service] ****
ok: [192.168.145.146]

TASK [install : Enabling mysql service and start the service] ****
ok: [192.168.145.146]

TASK [copy : Copying your php file in the /var/www/html] ****
ok: [192.168.145.146]

PLAY RECAP ****
192.168.145.146      : ok=5    changed=0    unreachable=0    failed=0    skipped=0    rescued=0    ignored=0
```

[root@ansible1 ansibleroles]#

14:44 / 16:06

Scroll for details

Ansible Tutorial | How To Create Ansible Roles From Playbook | Ansible Playbook vs Role vs ...

```
1
2
3 - hosts: testservers
4   remote_user: root
5
6
7 roles:
8   - install
9   - copy
10
11
12
13
```

Ansible Tutorial | How To Create Ansible Roles From Playbook | Ansible Playbook vs Role vs ...

```
1 ---
2
3 - name: Installing Apache, MYSQL DB Community edition & PHP
4   yum:
5     name:
6       - httpd
7       - mysql-community-server
8       - mysql-community-client
9       - php
10      - php-mysql
11    state: installed
12
13 - name: Enabling httpd service and start the service
14   service:
15     name: httpd
16     enabled: yes
17     state: started
18
19 - name: Enabling mysql service and start the service
20   service:
21     name: mysqld
22     enabled: yes
23     state: started
```

Ansible Tutorial | How To Create Ansible Roles From Playbook | Ansible Playbook vs Role vs ...

```
1 ...
2 ...
3 - name: restart httpd service
4   service:
5     name: httpd
6     state: restarted
```

Ansible Tutorial For Beginners | Using Variables and Facts



```
1 ---  
2 - hosts: testservers  
3 vars:  
4   Color: Green  
5   Car: Ferrari  
6   Place: US  
7 remote_user: root  
8 tasks:  
9   - name: Variable tutorial  
10    shell: echo " My Car is {{ Car }} and the color of the car is {{ Color }} " > /root/ansiblevariables.txt
```

Ansible Tutorial For Beginners | Using Variables and Facts
[root@ansible1 ansiblevariables]# ansible-playbook test.yaml

```
PLAY [testservers] *****  
  
TASK [Gathering Facts] *****  
ok: [192.168.145.146]  
  
TASK [Variable tutorial] *****  
changed: [192.168.145.146]  
  
PLAY RECAP *****  
192.168.145.146 : ok=2    changed=1    unreachable=0    failed=0    skipped=0    rescued=0    ignored=0  
  
[root@ansible1 ansiblevariables]# cat test.yaml
```

```
---  
- hosts: testservers  
  vars_files:  
  - variable.txt  
  remote_user: root  
  tasks:  
  - name: Variable tutorial  
    shell: echo " My Car is {{ Car }} and the color of the car is {{ Color }} " > /root/ansiblevariable.txt
```

[root@ansible1 ansiblevariables]#

Ahsible Tutorial For Beginners | Using Variables and Facts

```
"cap_lease",
"cap_audit_write",
"cap_audit_control",
"cap_setfcap",
"cap_mac_override",
"cap_mac_admin",
"cap_syslog",
"35",
"36+ep"
],
"ansible_system_capabilities_enforced": "True",
"ansible_system_vendor": "VMware, Inc.",
"ansible_uptime_seconds": 1573,
"ansible_user_dir": "/root",
"ansible_user_gecos": "root",
"ansible_user_gid": 0,
"ansible_user_id": "root",
"ansible_user_shell": "/bin/bash",
"ansible_user_uid": 0,
"ansible_userspace_architecture": "x86_64",
"ansible_userspace_bits": "64",
"ansible_virtualization_role": "guest",
"ansible_virtualization_type": "VMware",
"discovered_interpreter_python": "/usr/bin/python",
"gather_subset": [
    "all"
],
"module_setup": true
},
"changed": false
]
[root@ansible1 ansiblevariables]# ansible testservers -m setup -a " filter=*family*"
```

Scroll for details



Ansible Tutorial for Beginners



Ansible Tutorial for Beginners



- Ansible Variable
- Ansible Facts

Ansible Variables



- A variable name can only include letters, numbers, and underscores
- Variable names can begin with an underscore
- A variable name cannot begin with a number.

Valid Variables	In-Valid Variables
foo	*foo
foo_port	foo-port, foo port, foo.port
foo5, _foo	5foo, 12

Ansible Tutorial For Beginners | Using Variables and Facts

```
1 ---  
2 - hosts: testservers  
3 vars:  
4   Color: Green  
5   Car: Ferrari  
6   Place: US  
7 remote_user: root  
8 tasks:  
9   - name: Variable tutorial  
10    shell: echo " My Car is {{ Car }} and the color of the car is {{ Color }} " > /root/ansiblevariable.txt  
11
```

Ansible Tutorial For Beginners | Using Variables and Facts

A screenshot of a video player interface displaying an Ansible configuration file. The file is named `ansible_variable_usecases.yaml` and contains the following YAML code:

```
1
2 - hosts: testservers
3   vars:
4     Color: Green
5     Car: Ferrari
6     Place: US
7   remote_user: root
8   tasks:
9     - name: Variable tutorial
10       shell: echo " My Car is {{ Car }} and the color of the car is {{ Color }} " > /root/ansiblevariable.txt
```

The code defines a host group `testservers` with variables `Color`, `Car`, and `Place`. It specifies a `remote_user` of `root` and a task to run a shell command that outputs the value of `Car` and `Color` to a file.

A context menu is open over the line containing the shell command, with the "Copy" option highlighted. The menu options include:

- Show Diff Hunk
- Show Unsaved Changes...
- Cut
- Copy (highlighted)
- Paste
- Select All
- Open Containing Folder...
- Copy File Path
- Reveal in Side Bar

The video player interface at the bottom shows a progress bar at 6:47 / 25:09 and a "Scroll for details" button.

Ahhsible Tutorial For Beginners | Using Variables and Facts

```
[root@ansible1 ~]# cd ansiblevariables/  
[root@ansible1 ansiblevariables]# ls  
[root@ansible1 ansiblevariables]# vi test.yaml
```

I

Ahible Tutorial For Beginners | Using Variables and Facts



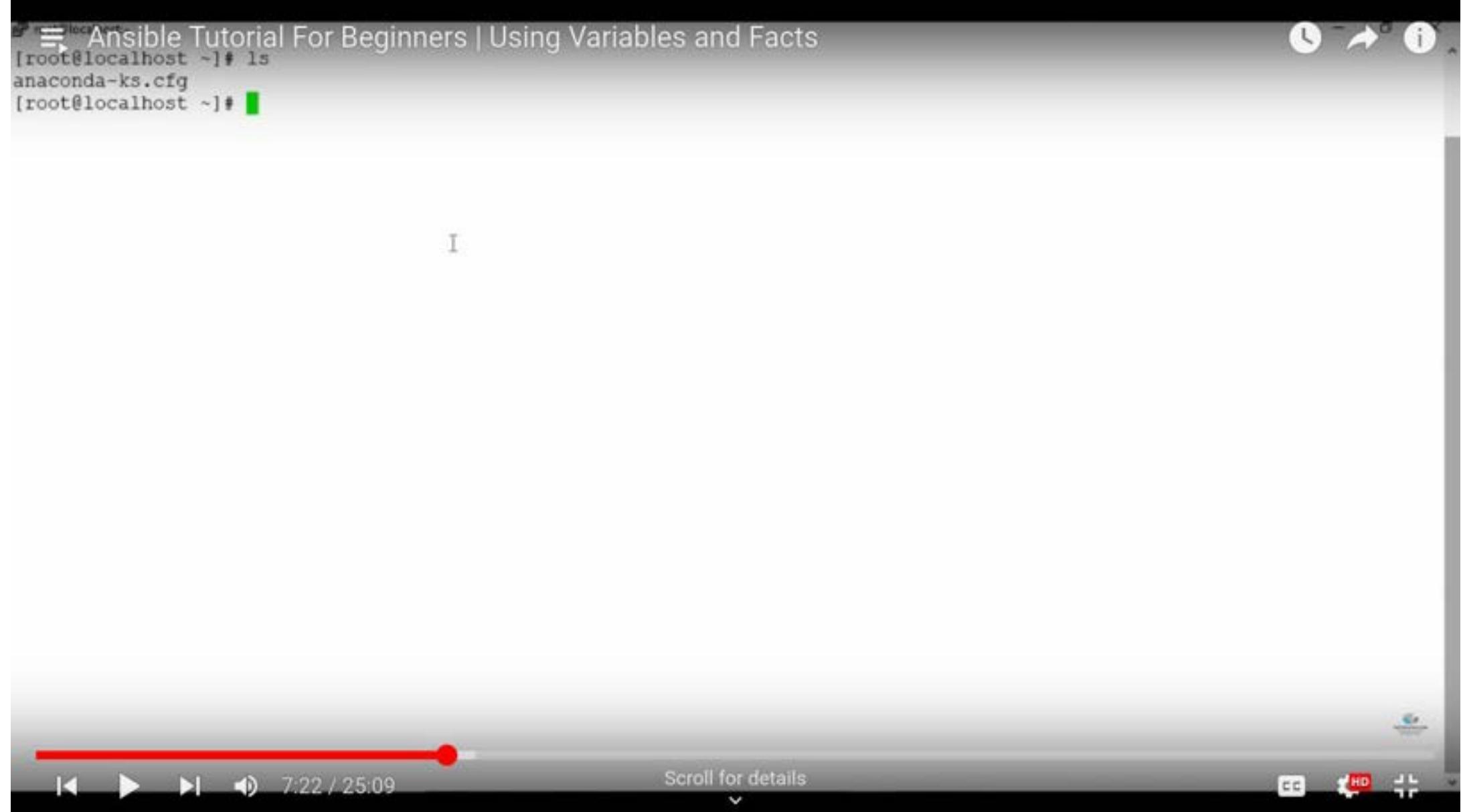
```
[root@ansible1 ~]# cd ansiblevariables/  
[root@ansible1 ansiblevariables]# ls  
[root@ansible1 ansiblevariables]# vi test.yaml  
[root@ansible1 ansiblevariables]#  
  
Car: Ferrari  
Place: US  
remote_user: root  
tasks:  
- name: Variable tutorial  
  shell: echo " My Car is {{ Car }} and the color of the car is {{ Color }} " > /root/ansiblevariable.txt
```

Ahsible Tutorial For Beginners | Using Variables and Facts



```
---  
- hosts: testservers  
  vars:  
    Color: Green  
    Car: Ferrari  
    Place: US  
  remote_user: root  
  tasks:  
    - name: Variable tutorial  
      shell: echo " My Car is {{ Car }} and the color of the car is {{ Color }} " > /root/ansiblevariable.txt
```





Ansible Tutorial For Beginners | Using Variables and Facts

```
[root@localhost ~]# ls  
anaconda-ks.cfg  
[root@localhost ~]# pwd  
/root  
[root@localhost ~]# █
```

◀ ▶ ⏪ ⏩ 🔍 7:26 / 25:09

Scroll for details

CC HD

Ahible Tutorial For Beginners | Using Variables and Facts

```
[root@ansible1 ~]# cd ansiblevariables/  
[root@ansible1 ansiblevariables]# ls  
[root@ansible1 ansiblevariables]# vi test.yaml  
[root@ansible1 ansiblevariables]# ansible-playbook test.yaml
```

I

◀ ▶ ⏪ 🔍 7:36 / 25:09

Scroll for details

CC HD

```
Ansible Tutorial For Beginners | Using Variables and Facts
[root@ansible ~]# cd ansiblevariables/
[root@ansible ansiblevariables]# ls
[root@ansible ansiblevariables]# vi test.yaml
[root@ansible ansiblevariables]# ansible-playbook test.yaml

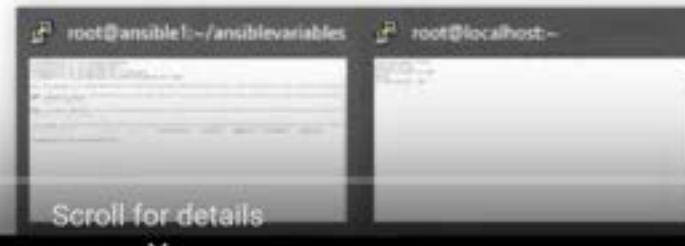
PLAY [testservers] *****

TASK [Gathering Facts] *****
ok: [192.168.145.146]

TASK [Variable tutorial] *****
changed: [192.168.145.146]

PLAY RECAP *****
192.168.145.146 : ok=2    changed=1    unreachable=0    failed=0    skipped=0    rescued=0    ignored=0

[root@ansible ansiblevariables]#
```



◀ ▶ ⏪ ⏩ 7:44 / 25:09

CC HD

Ansible Tutorial For Beginners | Using Variables and Facts

```
[root@localhost ~]# ls  
anaconda-ks.cfg  
[root@localhost ~]# pwd  
/root  
[root@localhost ~]# ls  
anaconda-ks.cfg  ansiblevariable.txt  
[root@localhost ~]#
```

Ansible Tutorial For Beginners | Using Variables and Facts



```
[root@localhost ~]# ls  
anaconda-ks.cfg  
[root@localhost ~]# pwd  
/root  
[root@localhost ~]# ls  
anaconda-ks.cfg  ansiblevariable.txt  
[root@localhost ~]# cat ansiblevariable.txt
```

Ansible Tutorial For Beginners | Using Variables and Facts

```
[root@localhost ~]# ls
anaconda-ks.cfg
[root@localhost ~]# pwd
/root
[root@localhost ~]# ls
anaconda-ks.cfg  ansiblevariable.txt
[root@localhost ~]# cat ansiblevariable.txt
My Car is Ferrari and the color of the car is Green
[root@localhost ~]# █
```

Ansible Tutorial For Beginners | Using Variables and Facts

```
1 ---  
2 - hosts: testservers  
3   vars:  
4     Color: Green  
5     Car: Ferrari  
6     Place: US  
7   remote_user: root  
8   tasks:  
9     - name: Variable tutorial  
10       shell: echo " My Car is {{ Car }} and the color of the car is {{ Color }} " > /root/ansiblevariable.txt  
11
```

Ahhsible Tutorial For Beginners | Using Variables and Facts

```
[root@ansible1 ~]# cd ansiblevariables/
[root@ansible1 ansiblevariables]# ls
[root@ansible1 ansiblevariables]# vi test.yaml
[root@ansible1 ansiblevariables]# ansible-playbook test.yaml

PLAY [testservers] *****

TASK [Gathering Facts] *****
ok: [192.168.145.146]

TASK [Variable tutorial] *****
changed: [192.168.145.146]

PLAY RECAP *****
192.168.145.146      : ok=2    changed=1    unreachable=0    failed=0    skipped=0    rescued=0    ignored=0

[root@ansible1 ansiblevariables]#
```

Ahumble Tutorial For Beginners | Using Variables and Facts

```
[root@ansible1 ~]# cd ansiblevariables/
[root@ansible1 ansiblevariables]# ls
[root@ansible1 ansiblevariables]# vi test.yaml
[root@ansible1 ansiblevariables]# ansible-playbook test.yaml

PLAY [testservers] *****

TASK [Gathering Facts] *****
ok: [192.168.145.146]

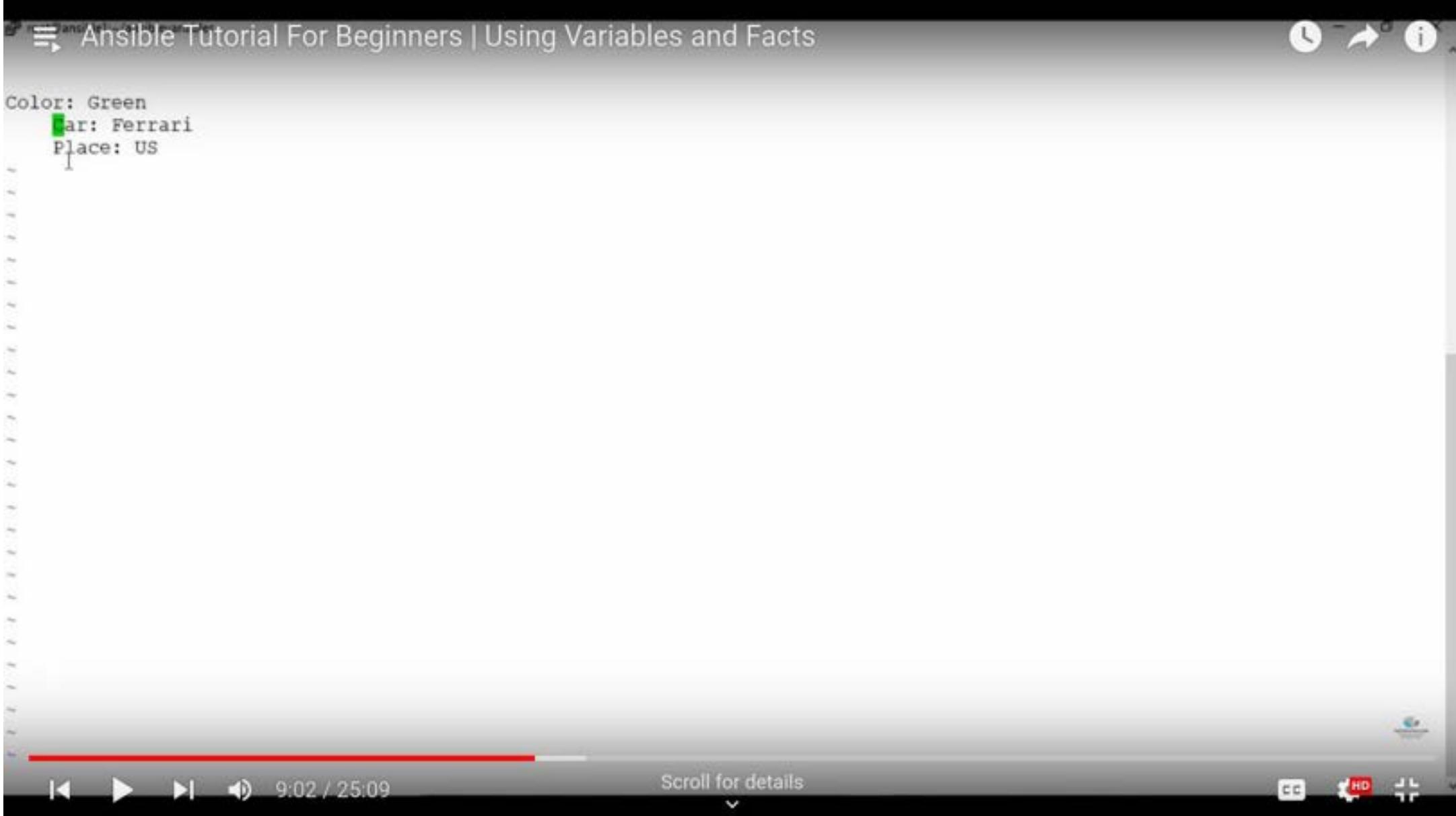
TASK [Variable tutorial] *****
changed: [192.168.145.146]

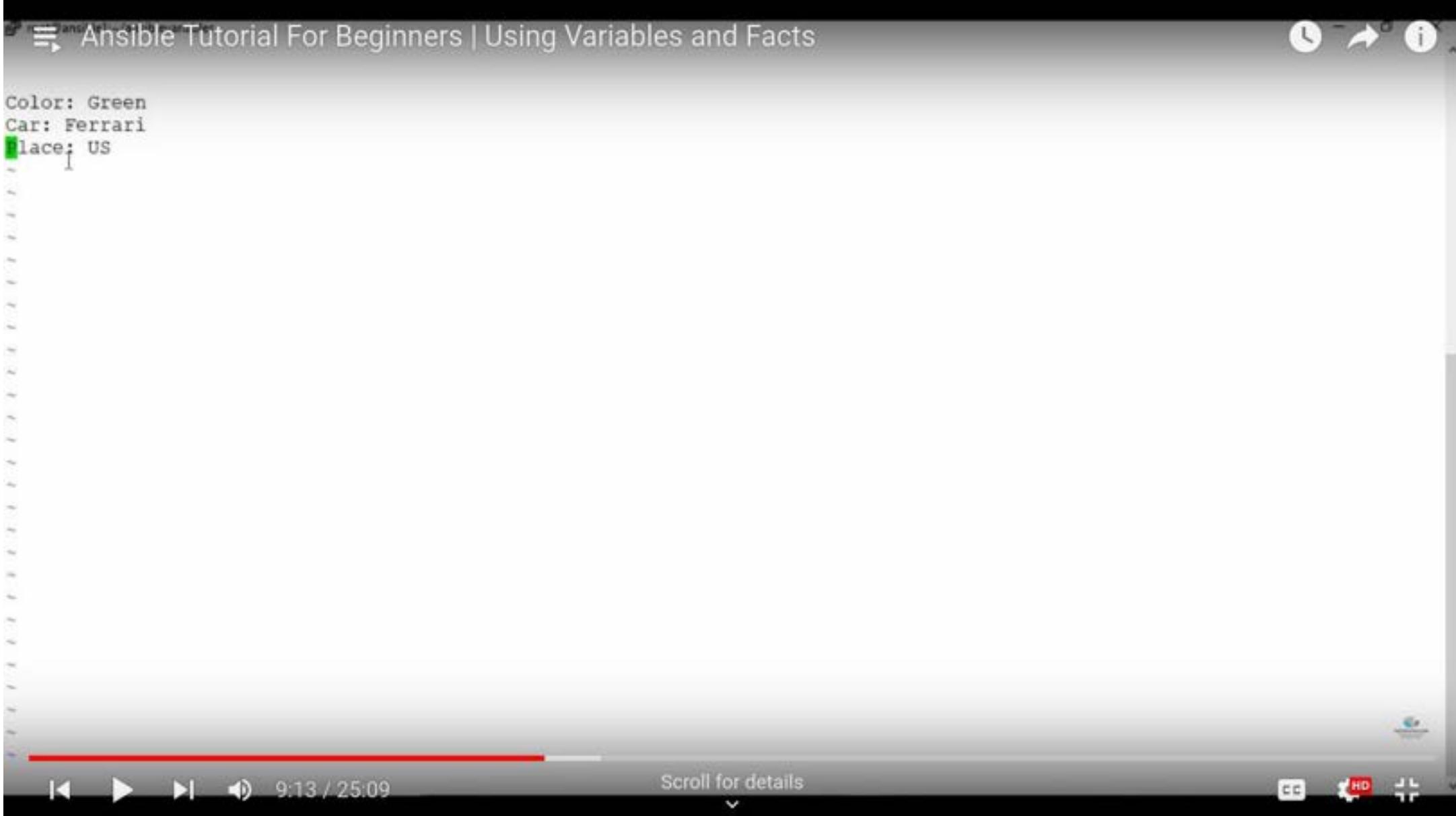
PLAY RECAP *****
192.168.145.146      : ok=2    changed=1    unreachable=0    failed=0    skipped=0    rescued=0    ignored=0

[root@ansible1 ansiblevariables]# ls
test.yaml
[root@ansible1 ansiblevariables]# vi variable.txt
```

Ansible Tutorial For Beginners | Using Variables and Facts

```
1 ---  
2 - hosts: testservers  
3 vars:  
4   Color: Green  
5   Car: Ferrari  
6   Place: US  
7 remote_user: root  
8 tasks:  
9   - name: Variable tutorial  
10    shell: echo " My Car is {{ Car }} and the color of the car is {{ Color }} " > /root/ansiblevariable.txt  
11
```





Ahible Tutorial For Beginners | Using Variables and Facts



```
[root@ansible1 ~]# cd ansiblevariables/  
[root@ansible1 ansiblevariables]# ls  
[root@ansible1 ansiblevariables]# vi test.yaml  
[root@ansible1 ansiblevariables]# ansible-playbook test.yaml  
  
PLAY [testservers] *****  
  
TASK [Gathering Facts] *****  
ok: [192.168.145.146]  
  
TASK [Variable tutorial] *****  
changed: [192.168.145.146]  
  
PLAY RECAP *****  
192.168.145.146 : ok=2    changed=1    unreachable=0    failed=0    skipped=0    rescued=0    ignored=0  
  
[root@ansible1 ansiblevariables]# ls  
test.yaml  
[root@ansible1 ansiblevariables]# vi variable.txt  
[root@ansible1 ansiblevariables]# ls  
test.yaml  variable.txt  
[root@ansible1 ansiblevariables]# █
```

I

Ahible Tutorial For Beginners | Using Variables and Facts



```
[root@ansible1 ~]# cd ansiblevariables/  
[root@ansible1 ansiblevariables]# ls  
[root@ansible1 ansiblevariables]# vi test.yaml  
[root@ansible1 ansiblevariables]# ansible-playbook test.yaml  
  
PLAY [testservers] *****  
  
TASK [Gathering Facts] *****  
ok: [192.168.145.146]  
  
TASK [Variable tutorial] *****  
changed: [192.168.145.146]  
  
PLAY RECAP *****  
192.168.145.146 : ok=2    changed=1    unreachable=0    failed=0    skipped=0    rescued=0    ignored=0  
  
[root@ansible1 ansiblevariables]# ls  
test.yaml  
[root@ansible1 ansiblevariables]# vi variable.txt  
[root@ansible1 ansiblevariables]# ls  
test.yaml  variable.txt  
[root@ansible1 ansiblevariables]# vi test.yaml
```

Ahsible Tutorial For Beginners | Using Variables and Facts



```
---
```

```
- hosts: testservers
  vars:
    Color: Green
    Car: Ferrari
    Place: US
  remote_user: root
  tasks:
    - name: Variable tutorial
      shell: echo " My Car is {{ Car }} and the color of the car is {{ Color }} " > /root/ansiblevariable.txt
```

I

Ahsible Tutorial For Beginners | Using Variables and Facts

```
---  
- hosts: testservers  
  vars_files:  
    - variables.txt  
  remote_user: root  
  tasks:  
    - name: Variable tutorial  
      shell: echo " My Car is {{ Car }} and the color of the car is {{ Color }} " > /root/ansiblevariable.txt
```

I

10:15 / 25:09

Scroll for details

Ahnsible Tutorial For Beginners | Using Variables and Facts



```
[root@ansible1 ~]# cd ansiblevariables/  
[root@ansible1 ansiblevariables]# ls  
[root@ansible1 ansiblevariables]# vi test.yaml  
[root@ansible1 ansiblevariables]# ansible-playbook test.yaml  
  
PLAY [testservers] *****  
  
TASK [Gathering Facts] *****  
ok: [192.168.145.146]  
  
TASK [Variable tutorial] *****  
changed: [192.168.145.146]  
  
PLAY RECAP *****  
192.168.145.146 : ok=2    changed=1    unreachable=0    failed=0    skipped=0    rescued=0    ignored=0  
  
[root@ansible1 ansiblevariables]# ls  
test.yaml  
[root@ansible1 ansiblevariables]# vi variable.txt  
[root@ansible1 ansiblevariables]# ls  
test.yaml variable.txt  
[root@ansible1 ansiblevariables]# vi test.yaml  
[root@ansible1 ansiblevariables]# ls  
test.yaml variable.txt  
[root@ansible1 ansiblevariables]# █
```

Ahumble Tutorial For Beginners | Using Variables and Facts

```
[root@ansible1 ~]# cd ansiblevariables/
[root@ansible1 ansiblevariables]# ls
[root@ansible1 ansiblevariables]# vi test.yaml
[root@ansible1 ansiblevariables]# ansible-playbook test.yaml

PLAY [testservers] *****

TASK [Gathering Facts] *****
ok: [192.168.145.146]

TASK [Variable tutorial] *****
changed: [192.168.145.146]

PLAY RECAP *****
192.168.145.146 : ok=2    changed=1    unreachable=0    failed=0    skipped=0    rescued=0    ignored=0

[root@ansible1 ansiblevariables]# ls
test.yaml
[root@ansible1 ansiblevariables]# vi variable.txt
[root@ansible1 ansiblevariables]# ls
test.yaml  variable.txt
[root@ansible1 ansiblevariables]# vi test.yaml
[root@ansible1 ansiblevariables]# ls
test.yaml  variable.txt
[root@ansible1 ansiblevariables]# vi test.yaml
```

Ahsible Tutorial For Beginners | Using Variables and Facts

```
---  
- hosts: testservers  
  vars_files:  
    - variables.txt  
  remote_user: root  
  tasks:  
    - name: Variable tutorial  
      shell: echo " My Car is {{ Car }} and the color of the car is {{ Color }} " > /root/ansiblevariable.txt
```

"test.yaml" 11L 226C 10:28 / 25:09

Scroll for details

Ahsible Tutorial For Beginners | Using Variables and Facts

```
---  
- hosts: testservers  
  vars_files:  
    - variable.txt  
  remote_user: root  
  tasks:  
    - name: Variable tutorial  
      shell: echo " My Car is {{ Car }} and the color of the car is {{ Color }} " > /root/ansiblevariable.txt
```

"test.yaml" 11L 226C 10:31 / 25:09

Scroll for details

Ansible Tutorial For Beginners | Using Variables and Facts

```
[root@localhost ~]# ls  
anaconda-ks.cfg  
[root@localhost ~]# pwd  
/root  
[root@localhost ~]# ls  
anaconda-ks.cfg  ansiblevariable.txt  
[root@localhost ~]# cat ansiblevariable.txt  
My Car is Ferrari and the color of the car is Green  
[root@localhost ~]# ls  
anaconda-ks.cfg  ansiblevariable.txt  
[root@localhost ~]# █
```

Ansible Tutorial For Beginners | Using Variables and Facts

```
[root@localhost ~]# ls
anaconda-ks.cfg
[root@localhost ~]# pwd
/root
[root@localhost ~]# ls
anaconda-ks.cfg  ansiblevariable.txt
[root@localhost ~]# cat ansiblevariable.txt
My Car is Ferrari and the color of the car is Green
[root@localhost ~]# ls
anaconda-ks.cfg  ansiblevariable.txt
[root@localhost ~]# rm -f ansiblevariable.txt
[root@localhost ~]#
```

Ansible Tutorial For Beginners | Using Variables and Facts

```
[root@localhost ~]# ls  
anaconda-ks.cfg  
[root@localhost ~]#
```

The image shows a video player interface with a terminal window. The terminal window title is "Ansible Tutorial For Beginners | Using Variables and Facts". Inside the terminal, the command "ls" is run, showing the file "anaconda-ks.cfg". The video player has a progress bar at the bottom left, a timestamp of "10:49 / 25:09" in the center, and various control icons like play, pause, and volume on the right.

Ahnsible Tutorial For Beginners | Using Variables and Facts

```
[root@ansible1 ansiblevariables]# ansible-playbook test.yaml

PLAY [testservers] *****

TASK [Gathering Facts] *****
ok: [192.168.145.146]

TASK [Variable tutorial] *****
changed: [192.168.145.146]

PLAY RECAP *****
192.168.145.146      : ok=2    changed=1    unreachable=0    failed=0    skipped=0    rescued=0    ignored=0

[root@ansible1 ansiblevariables]#
```

Ansible Tutorial For Beginners | Using Variables and Facts

```
[root@localhost ~]# ls  
anaconda-ks.cfg  
[root@localhost ~]# ls  
anaconda-ks.cfg  ansiblevariable.txt  
[root@localhost ~]# cat ansiblevariable.txt  
My Car is Ferrari and the color of the car is Green  
[root@localhost ~]#
```

I

Ahnsible Tutorial For Beginners | Using Variables and Facts

```
[root@ansible1 ansiblevariables]# ansible-playbook test.yaml

PLAY [testservers] *****

TASK [Gathering Facts] *****
ok: [192.168.145.146]

TASK [Variable tutorial] *****
changed: [192.168.145.146]

PLAY RECAP *****
192.168.145.146 : ok=2    changed=1    unreachable=0    failed=0    skipped=0    rescued=0    ignored=0

[root@ansible1 ansiblevariables]# cat test.yaml
```

```
---
- hosts: testservers
  vars_files:
    - variable.txt
  remote_user: root
  tasks:
    - name: Variable tutorial
      shell: echo " My Car is {{ Car }} and the color of the car is {{ Color }} " > /root/ansiblevariable.txt
```

```
[root@ansible1 ansiblevariables]#
```

Ansible Tutorial For Beginners | Using Variables and Facts
[root@ansible1 ansiblevariables]# vi test.yaml



I

◀ ▶ ⏪ ⏩ 🔍 11:53 / 25:09

Scroll for details

CC HD

Ansible Tutorial For Beginners | Using Variables and Facts
[root@ansible1 ansiblevariables]# vi test.yaml



I

◀ ▶ ⏪ 🔍 11:53 / 25:09

Scroll for details

CC HD

Ahnsible Tutorial For Beginners | Using Variables and Facts

```
[root@ansible1 ansiblevariables]# ansible-playbook test.yaml

PLAY [testservers] *****

TASK [Gathering Facts] *****
ok: [192.168.145.146]

TASK [Variable tutorial] *****
changed: [192.168.145.146]

PLAY RECAP *****
192.168.145.146 : ok=2    changed=1    unreachable=0    failed=0    skipped=0    rescued=0    ignored=0

[root@ansible1 ansiblevariables]# cat test.yaml

---
- hosts: testservers
  vars_files:
    - variable.txt
  remote_user: root
  tasks:
    - name: Variable tutorial
      shell: echo " My Car is {{ Car }} and the color of the car is {{ Color }} " > /root/ansiblevariable.txt

[root@ansible1 ansiblevariables]#
```

Ansible Tutorial For Beginners | Using Variables and Facts
[root@ansible1 ansiblevariables]# vi test.yaml



I

◀ ▶ ⏪ ⏩ 🔍 11:53 / 25:09

Scroll for details

CC HD

Ahsible Tutorial For Beginners | Using Variables and Facts

```
---  
- hosts: testservers  
  vars_files:  
    - variable.txt  
  remote_user: root  
  tasks:  
    - name: Variable tutorial  
      shell: echo " My Car is {{ Car }} and the color of the car is {{ Color }} " > /root/ansiblevariable.txt
```

I

"test.yaml" 11L 225C 11:54 / 25:09

Scroll for details

Ahsible Tutorial For Beginners | Using Variables and Facts

```
---  
- hosts: testservers  
  remote_user: root  
  tasks:  
    - include_vars: variable.txt  
    - name: Variable tutorial  
      shell: echo " My Car is {{ Car }} and the color of the car is {{ Color }} " > /root/ansiblevariable.txt
```

Ansible Tutorial For Beginners | Using Variables and Facts

```
[root@localhost ~]# ls
anaconda-ks.cfg
[root@localhost ~]# ls
anaconda-ks.cfg  ansiblevariable.txt
[root@localhost ~]# cat ansiblevariable.txt
My Car is Ferrari and the color of the car is Green
[root@localhost ~]# ls
anaconda-ks.cfg  ansiblevariable.txt
[root@localhost ~]# c
```

Ansible Tutorial For Beginners | Using Variables and Facts

```
[root@localhost ~]# ls  
anaconda-ks.cfg  
[root@localhost ~]# ls  
anaconda-ks.cfg  ansiblevariable.txt  
[root@localhost ~]# cat ansiblevariable.txt  
My Car is Ferrari and the color of the car is Green  
[root@localhost ~]# ls  
anaconda-ks.cfg  ansiblevariable.txt  
[root@localhost ~]# rm -f ansiblevariable.txt  
[root@localhost ~]#
```

Ansible Tutorial For Beginners | Using Variables and Facts

```
[root@localhost ~]# ls  
anaconda-ks.cfg  
[root@localhost ~]# ls  
anaconda-ks.cfg  ansiblevariable.txt  
[root@localhost ~]# cat ansiblevariable.txt  
My Car is Ferrari and the color of the car is Green  
[root@localhost ~]# ls  
anaconda-ks.cfg  ansiblevariable.txt  
[root@localhost ~]# rm -f ansiblevariable.txt  
[root@localhost ~]# ls  
anaconda-ks.cfg  
[root@localhost ~]#
```

Ahnsible Tutorial For Beginners | Using Variables and Facts

```
[root@ansible1 ansiblevariables]# vi test.yaml
[root@ansible1 ansiblevariables]# ansible-playbook test.yaml

PLAY [testservers] *****

TASK [Gathering Facts] *****
ok: [192.168.145.146]

TASK [include_vars] *****
ok: [192.168.145.146]

TASK [Variable tutorial] *****
changed: [192.168.145.146]

PLAY RECAP *****
192.168.145.146      : ok=3    changed=1    unreachable=0    failed=0    skipped=0    rescued=0    ignored=0

[root@ansible1 ansiblevariables]#
```

Ansible Tutorial For Beginners | Using Variables and Facts

```
[root@localhost ~]# ls  
anaconda-ks.cfg  ansiblevariable.txt  
[root@localhost ~]# cat ansiblevariable.txt  
My Car is Ferrari and the color of the car is Green  
[root@localhost ~]#
```

I

13:01 / 25:09

Scroll for details

CC HD

Ansible Tutorial For Beginners | Using Variables and Facts

```
[root@localhost ~]# ls  
anaconda-ks.cfg  ansiblevariable.txt  
[root@localhost ~]# cat ansiblevariable.txt  
My Car is Ferrari and the color of the car is Green  
[root@localhost ~]#
```

I

13:01 / 25:09

Scroll for details

CC HD

Ansible Tutorial For Beginners | Using Variables and Facts

```
[root@ansible1 ansiblevariables]# vi test.yaml
[root@ansible1 ansiblevariables]# ansible-playbook test.yaml

PLAY [testservers] *****

TASK [Gathering Facts] *****
ok: [192.168.145.146]

TASK [include_vars] *****
ok: [192.168.145.146]

TASK [Variable tutorial] *****
changed: [192.168.145.146]

PLAY RECAP *****
192.168.145.146 : ok=3    changed=1    unreachable=0    failed=0    skipped=0    rescued=0    ignored=0

[root@ansible1 ansiblevariables]# cat test.yaml

---
- hosts: testservers
  remote_user: root
  tasks:
    - include_vars: variable.txt
    - name: Variable tutorial
      shell: echo " My Car is {{ Car }} and the color of the car is {{ Color }} " > /root/ansiblevariable.txt

[root@ansible1 ansiblevariables]#
```

Ahnsible Tutorial For Beginners | Using Variables and Facts

```
[root@ansible1 ansiblevariables]# vi test.yaml
[root@ansible1 ansiblevariables]# ansible-playbook test.yaml

PLAY [testservers] *****

TASK [Gathering Facts] *****
ok: [192.168.145.146]

TASK [include_vars] *****
ok: [192.168.145.146]

TASK [Variable tutorial] *****
changed: [192.168.145.146]

PLAY RECAP *****
192.168.145.146 : ok=3    changed=1    unreachable=0    failed=0    skipped=0    rescued=0    ignored=0

[root@ansible1 ansiblevariables]# cat test.yaml

---
- hosts: testservers
  remote_user: root
  tasks:
    - include_vars: variable.txt
    - name: Variable tutorial
      shell: echo " My Car is {{ Car }} and the color of the car is {{ Color }} " > /root/ansiblevariable.txt

[root@ansible1 ansiblevariables]#
```

Ansible Tutorial For Beginners | Using Variables and Facts

```
1  ---
2  - hosts: testservers
3  vars:
4    Color:
5      - Green
6
7
8  remote_user: root
9  tasks:
10 - name: Variable tutorial
11   shell: echo " My Car is {{ Car }} and the color of the car is {{ Color }} " > /root/ansiblevariable.txt
12
```

Ansible Tutorial For Beginners | Using Variables and Facts

```
1  ---
2  - hosts: testservers
3  vars:
4    Color:
5      - Green
6      - White
7      - Red
8
9
10 remote_user: root
11 tasks:
12 - name: Variable tutorial
13   shell: echo " My Car is {{ Car }} and the color of the car is {{ Color }} " > /root/ansiblevariable.txt
14
```

Ansible Tutorial For Beginners | Using Variables and Facts



```
1  ---
2  - hosts: testservers
3  vars:
4    Color:
5      - Green
6      - White
7      - Red
8
9
10 remote_user: root
11 tasks:
12   - name: Variable tutorial
13     shell: echo " My Favourite Color is {{ Color[2] }} and the color i dont like is {{ Color[0] }} " > /root/ansiblevariable.
14
```

Ansible Tutorial For Beginners | Using Variables and Facts



```
1  ---
2  - hosts: testservers
3  vars:
4    Color:
5      - Green
6      - White
7      - Red
8
9
10 remote_user: root
11 tasks:
12   - name: Variable tutorial
13     shell: echo " My Favourite Color is {{ Color[2] }} and the color i dont like is {{ Color[0] }} " > /root/ansiblevariable.
14
```

Ansible Tutorial For Beginners | Using Variables and Facts

```
[root@ansible1 ansiblevariables]# vi test.yaml
[root@ansible1 ansiblevariables]# ansible-playbook test.yaml

PLAY [testservers] *****

TASK [Gathering Facts] *****
ok: [192.168.145.146]

TASK [include_vars] *****
ok: [192.168.145.146]

TASK [Variable tutorial] *****
changed: [192.168.145.146]

PLAY RECAP *****
192.168.145.146 : ok=3    changed=1    unreachable=0    failed=0    skipped=0    rescued=0    ignored=0

[root@ansible1 ansiblevariables]# cat test.yaml

---
- hosts: testservers
  remote_user: root
  tasks:
    - include_vars: variable.txt
    - name: Variable tutorial
      shell: echo " My Car is {{ Car }} and the color of the car is {{ Color }} " > /root/ansiblevariable.txt

[root@ansible1 ansiblevariables]# ls
test.yaml  variable.txt
[root@ansible1 ansiblevariables]#
```

Ansible Tutorial For Beginners | Using Variables and Facts

```
[root@ansible1 ansiblevariables]# vi test.yaml
[root@ansible1 ansiblevariables]# ansible-playbook test.yaml

PLAY [testservers] *****

TASK [Gathering Facts] *****
ok: [192.168.145.146]

TASK [include_vars] *****
ok: [192.168.145.146]

TASK [Variable tutorial] *****
changed: [192.168.145.146]

PLAY RECAP *****
192.168.145.146 : ok=3    changed=1    unreachable=0    failed=0    skipped=0    rescued=0    ignored=0

[root@ansible1 ansiblevariables]# cat test.yaml

---
- hosts: testservers
  remote_user: root
  tasks:
    - include_vars: variable.txt
    - name: Variable tutorial
      shell: echo " My Car is {{ Car }} and the color of the car is {{ Color }} " > /root/ansiblevariable.txt

[root@ansible1 ansiblevariables]# ls
test.yaml  variable.txt
[root@ansible1 ansiblevariables]# vi testnew.yaml
```

Ahnsible Tutorial For Beginners | Using Variables and Facts

```
[root@ansible1 ansiblevariables]# vi test.yaml
[root@ansible1 ansiblevariables]# ansible-playbook test.yaml

PLAY [testservers] ****
TASK [Gathering Facts] ****
ok: [192.168.145.146]

TASK [include_vars] ****
ok: [192.168.145.146]

TASK [Variable tutorial] ****
changed: [192.168.145.146]

PLAY RECAP ****
192.168.145.146      : ok=3    changed=1    unreachable=0    failed=0    skipped=0    rescued=0    ignored=0

[root@ansible1 ansiblevariables]# cat test.yaml

---
- hosts: testservers
  remote_user: root
  tasks:
    - include_vars: variable.txt
    - name: Variable tutorial
      shell: echo " My Car is {{ Car }} and the color of the car is {{ Color }} " > /root/ansiblevariable.txt

[root@ansible1 ansiblevariables]# ls
test.yaml  variable.txt
[root@ansible1 ansiblevariables]# vi testnew.yam
```

Settings

15:38 / 25:09

Scroll for details

Ahsible Tutorial For Beginners | Using Variables and Facts

```
---  
- hosts: testservers  
  vars:  
    Color:  
      - Green  
      - White  
      - Red  
  
  remote_user: root  
  tasks:  
    - name: Variable tutorial  
      shell: echo " My Favourite Color is {{ Color[2] }} and the color i dont like is {{ Color[0] }} " > /root/ansiblevariable.txt
```

15:45 / 25:09

Scroll for details

Ansible Tutorial For Beginners | Using Variables and Facts



```
[root@localhost ~]# ls  
anaconda-ks.cfg  ansiblevariable.txt  
[root@localhost ~]# cat ansiblevariable.txt  
My Car is Ferrari and the color of the car is Green  
[root@localhost ~]# ls  
anaconda-ks.cfg  ansiblevariable.txt  
[root@localhost ~]# rm -f ansiblevariable.txt
```

Ansible Tutorial For Beginners | Using Variables and Facts
[root@localhost ~]# ls
anaconda-ks.cfg
[root@localhost ~]#



◀ ▶ ⏪ ⏩ 16:02 / 25:09

Scroll for details

CC HD

Ahible Tutorial For Beginners | Using Variables and Facts

```
[root@ansible1 ansiblevariables]# ansible-p  
ansible-playbook      ansible-playbook-2.7  ansible-pull-2  
ansible-playbook-2    ansible-pull        ansible-pull-2.7  
[root@ansible1 ansiblevariables]# ansible-p
```

I

Ahsible Tutorial For Beginners | Using Variables and Facts



```
[root@ansible1 ansiblevariables]# ansible-p
ansible-playbook      ansible-playbook-2.7  ansible-pull-2
ansible-playbook-2    ansible-pull          ansible-pull-2.7
[root@ansible1 ansiblevariables]# ansible-playbook testnew.yaml

PLAY [testservers] *****

TASK [Gathering Facts] *****
ok: [192.168.145.146]

TASK [Variable tutorial] *****
changed: [192.168.145.146]

PLAY RECAP *****
192.168.145.146      : ok=2    changed=1    unreachable=0    failed=0    skipped=0    rescued=0    ignored=0

[root@ansible1 ansiblevariables]#
```

Ansible Tutorial For Beginners | Using Variables and Facts

```
[root@localhost ~]# ls  
anaconda-ks.cfg  
[root@localhost ~]# ls  
anaconda-ks.cfg  ansiblevariable.txt  
[root@localhost ~]#
```

I

16:24 / 25:09

Scroll for details

CC HD

Ansible Tutorial For Beginners | Using Variables and Facts

```
[root@localhost ~]# ls  
anaconda-ks.cfg  
[root@localhost ~]# ls  
anaconda-ks.cfg  ansiblevariable.txt  
[root@localhost ~]# cat ansiblevariable.txt  
My Favourite Color is Red and the color i dont like is Green  
[root@localhost ~]#
```

Ansible Tutorial For Beginners | Using Variables and Facts



```
1  ---
2  - hosts: testservers
3  vars:
4    Color:
5      - Green
6      - White
7      - Red
8
9
10 remote_user: root
11 tasks:
12   - name: Variable tutorial
13     shell: echo " My Favourite Color is {{ Color[2] }} and the color i dont like is {{ Color[0] }} " > /root/ansiblevariable.
14
```

Ansible Tutorial For Beginners | Using Variables and Facts

```
1 ---  
2 - hosts: testservers  
3 vars:  
4   Color: [ Green, White, Red ]  
5     - Green  
6     - White  
7     - Red  
8  
9  
10 remote_user: root  
11 tasks:  
12   - name: Variable tutorial  
13     shell: echo " My Favourite Color is {{ Color[2] }} and the color i dont like is {{ Color[0] }} " > /root/ansiblevariable.  
14
```

Ahnsible Tutorial For Beginners | Using Variables and Facts



```
[root@ansible1 ansiblevariables]# ansible-p
ansible-playbook      ansible-playbook-2.7  ansible-pull-2
ansible-playbook-2    ansible-pull          ansible-pull-2.7
[root@ansible1 ansiblevariables]# ansible-playbook testnew.yaml

PLAY [testservers] *****

TASK [Gathering Facts] *****
ok: [192.168.145.146]

TASK [Variable tutorial] *****
changed: [192.168.145.146]

PLAY RECAP *****
192.168.145.146      : ok=2    changed=1    unreachable=0    failed=0    skipped=0    rescued=0    ignored=0

[root@ansible1 ansiblevariables]#
```



Ahsible Tutorial For Beginners | Using Variables and Facts

```
    "uids": [
        "3b568120-4646-46ff-aaf2-a65016183b9f"
    ],
    "sectors": "2097152",
    "sectorsize": 512,
    "size": "1.00 GB",
    "start": "2048",
    "uuid": "3b568120-4646-46ff-aaf2-a65016183b9f"
},
"sda2": {
    "holders": [
        "centos-root",
        "centos-swap"
    ],
    "links": {
        "ids": [
            "lvm-pv-uuid-oml5Ln-uoVX-3RYm-QXUE-lUrs-w3Yn-91NCH4"
        ],
        "labels": [],
        "masters": [
            "dm-0",
            "dm-1"
        ],
        "uids": []
    },
    "sectors": "39843840",
    "sectorsize": 512,
    "size": "19.00 GB",
    "start": "2099200",
    "uuid": null
}
```

Ansible Tutorial For Beginners | Using Variables and Facts

```
    "sas_address": null,
    "sas_device_handle": null,
    "scheduler_mode": "deadline",
    "sectors": "1880064",
    "sectorsize": "2048",
    "size": "918.00 MB",
    "support_discard": "0",
    "vendor": "NECVMWar",
    "virtual": 1
  },
  "ansible_distribution": "CentOS",
  "ansible_distribution_file_parsed": true,
  "ansible_distribution_file_path": "/etc/redhat-release",
  "ansible_distribution_file_variety": "RedHat",
  "ansible_distribution_major_version": "?",
  "ansible_distribution_release": "Core",
  "ansible_distribution_version": "7.6",
  "ansible_dns": {
    "nameservers": [
      "192.168.145.2"
    ],
    "search": [
      "localdomain"
    ]
  },
  "ansible_domain": "localdomain",
  "ansible_effective_group_id": 0,
  "ansible_effective_user_id": 0,
  "ansible_ens33": {
    "active": true,
    "device": "ens33",
    "features": [
      "large_sendoff"
    ],
    "mac": "00:0c:29:4d:4f:00",
    "mtu": 1500,
    "name": "ens33",
    "parent": null,
    "queueingdiscipline": "bfq"
  }
}
```

Ansible Tutorial For Beginners | Using Variables and Facts

```
    "sas_address": null,
    "sas_device_handle": null,
    "scheduler_mode": "deadline",
    "sectors": "1880064",
    "sectorsize": "2048",
    "size": "918.00 MB",
    "support_discard": "0",
    "vendor": "NECVMWar",
    "virtual": 1
  },
  "ansible_distribution": "CentOS",
  "ansible_distribution_file_parsed": true,
  "ansible_distribution_file_path": "/etc/redhat-release",
  "ansible_distribution_file_variety": "RedHat",
  "ansible_distribution_major_version": "?",
  "ansible_distribution_release": "Core",
  "ansible_distribution_version": "7.6",
  "ansible_dns": {
    "nameservers": [
      "192.168.145.2"
    ],
    "search": [
      "localdomain"
    ]
  },
  "ansible_domain": "localdomain",
  "ansible_effective_group_id": 0,
  "ansible_effective_user_id": 0,
  "ansible_ens33": {
    "active": true,
    "device": "ens33",
    "features": 15248593
  }
},
```



```
        "sas_address": null,
        "sas_device_handle": null,
        "scheduler_mode": "deadline",
        "sectors": "1880064",
        "sectorsize": "2048",
        "size": "918.00 MB",
        "support_discard": "0",
        "vendor": "NECVMWar",
        "virtual": 1
    },
    "ansible_distribution": "CentOS",
    "ansible_distribution_file_parsed": true,
    "ansible_distribution_file_path": "/etc/redhat-release",
    "ansible_distribution_file_variety": "RedHat",
    "ansible_distribution_major_version": "7",
    "ansible_distribution_release": "Core",
    "ansible_distribution_version": "7.6",
    "ansible_dns": {
        "nameservers": [
            "192.168.145.2"
        ],
        "search": [
            "localdomain"
        ]
    },
    "ansible_domain": "localdomain",
    "ansible_effective_group_id": 0,
    "ansible_effective_user_id": 0,
    "ansible_ens33": {
        "active": true,
        "device": "ens33",
        "features": [
            "large_sendoff"
        ],
        "mac": "00:0c:29:4d:3e:00",
        "mtu": 1500,
        "name": "ens33",
        "parent": null,
        "queueingdiscipline": "bfq"
    }
}
```



```
        "sas_address": null,
        "sas_device_handle": null,
        "scheduler_mode": "deadline",
        "sectors": "1880064",
        "sectorsize": "2048",
        "size": "918.00 MB",
        "support_discard": "0",
        "vendor": "NECVMWar",
        "virtual": 1
    },
    "ansible_distribution": "CentOS",
    "ansible_distribution_file_parsed": true,
    "ansible_distribution_file_path": "/etc/redhat-release",
    "ansible_distribution_file_variety": "RedHat",
    "ansible_distribution_major_version": "?",
    "ansible_distribution_release": "Core",
    "ansible_distribution_version": "7.6",
    "ansible_dns": {
        "nameservers": [
            "192.168.145.2"
        ],
        "search": [
            "localdomain"
        ]
    },
    "ansible_domain": "localdomain",
    "ansible_effective_group_id": 0,
    "ansible_effective_user_id": 0,
    "ansible_ens33": {
        "active": true,
        "device": "ens33",
        "features": [
            "large_sendoff"
        ],
        "mac": "00:0c:29:4d:4f:00",
        "mtu": 1500,
        "name": "ens33",
        "parent": null,
        "queueingdiscipline": "bfq"
    }
}
```

Ahsible Tutorial For Beginners | Using Variables and Facts



```
"sas_address": null,  
"sas_device_handle": null,  
"scheduler_mode": "deadline",  
"sectors": "1880064",  
"sectorsize": "2048",  
"size": "918.00 MB",  
"support_discard": "0",  
"vendor": "NECVMWar",  
"virtual": 1  
},  
"ansible_distribution": "CentOS",  
"ansible_distribution_file_parsed": true,  
"ansible_distribution_file_path": "/etc/redhat-release",  
"ansible_distribution_file_variety": "RedHat",  
"ansible_distribution_major_version": "?",  
"ansible_distribution_release": "Core",  
"ansible_distribution_version": "7.6",  
"ansible_dns": {  
    "nameservers": [  
        "192.168.145.2"  
    ],  
    "search": [  
        "localdomain"  
    ]  
},  
"ansible_domain": "localdomain",  
"ansible_effective_group_id": 0,  
"ansible_effective_user_id": 0,  
"ansible_ens33": {  
    "active": true,  
    "device": "ens33",  
    "features": {  
        "feature": "large_sendoff",  
        "value": 1  
    },  
    "mac": "00:0c:29:4d:3f:00",  
    "mtu": 1500,  
    "name": "ens33",  
    "parent": null,  
    "queueingdiscipline": "noop",  
    "state": "up",  
    "txqueuelen": 1000  
},  
"ansible_facts": {  
    "arch": "x86_64",  
    "blkdevs": [{}],  
    "bootloader": "GRUB2",  
    "chassis": "PC",  
    "cpu": {  
        "cores": 4,  
        "family": 65, "model": 4, "stepping": 0,  
        "topology": {  
            "cores": 1, "sockets": 1, "threads": 1  
        }  
    },  
    "distribution": "CentOS",  
    "distribution_file": "/etc/redhat-release",  
    "distribution_file_variety": "RedHat",  
    "distribution_major_version": "?",  
    "distribution_release": "Core",  
    "distribution_version": "7.6",  
    "dns": {  
        "nameservers": [  
            "192.168.145.2"  
        ],  
        "search": [  
            "localdomain"  
        ]  
    },  
    "domain": "localdomain",  
    "effective_group_id": 0,  
    "effective_user_id": 0, "file_descriptors": {  
        "limits": {  
            "hard": 4096, "soft": 1024  
        },  
        "open_max": 2048, "open_files": 1024  
    },  
    "font": "Ubuntu",  
    "group": {  
        "id": 0, "name": "root", "users": ["root"]  
    },  
    "hostname": "centos",  
    "hw": {  
        "mem": 8192, "model": "QEMU PC (AMD Phenom II X4 945) (Q35)",  
        "processors": [{"model": "AMD Phenom II X4 945", "speed": 2400, "type": "CPU"}]  
    },  
    "ip": {  
        "loopback": "127.0.0.1",  
        "eth0": "192.168.145.2",  
        "ens33": "192.168.145.2",  
        "ens34": "192.168.145.1",  
        "ens35": "192.168.145.3",  
        "ens36": "192.168.145.4",  
        "ens37": "192.168.145.5",  
        "ens38": "192.168.145.6",  
        "ens39": "192.168.145.7",  
        "ens40": "192.168.145.8",  
        "ens41": "192.168.145.9",  
        "ens42": "192.168.145.10",  
        "ens43": "192.168.145.11",  
        "ens44": "192.168.145.12",  
        "ens45": "192.168.145.13",  
        "ens46": "192.168.145.14",  
        "ens47": "192.168.145.15",  
        "ens48": "192.168.145.16",  
        "ens49": "192.168.145.17",  
        "ens50": "192.168.145.18",  
        "ens51": "192.168.145.19",  
        "ens52": "192.168.145.20",  
        "ens53": "192.168.145.21",  
        "ens54": "192.168.145.22",  
        "ens55": "192.168.145.23",  
        "ens56": "192.168.145.24",  
        "ens57": "192.168.145.25",  
        "ens58": "192.168.145.26",  
        "ens59": "192.168.145.27",  
        "ens60": "192.168.145.28",  
        "ens61": "192.168.145.29",  
        "ens62": "192.168.145.30",  
        "ens63": "192.168.145.31",  
        "ens64": "192.168.145.32",  
        "ens65": "192.168.145.33",  
        "ens66": "192.168.145.34",  
        "ens67": "192.168.145.35",  
        "ens68": "192.168.145.36",  
        "ens69": "192.168.145.37",  
        "ens70": "192.168.145.38",  
        "ens71": "192.168.145.39",  
        "ens72": "192.168.145.40",  
        "ens73": "192.168.145.41",  
        "ens74": "192.168.145.42",  
        "ens75": "192.168.145.43",  
        "ens76": "192.168.145.44",  
        "ens77": "192.168.145.45",  
        "ens78": "192.168.145.46",  
        "ens79": "192.168.145.47",  
        "ens80": "192.168.145.48",  
        "ens81": "192.168.145.49",  
        "ens82": "192.168.145.50",  
        "ens83": "192.168.145.51",  
        "ens84": "192.168.145.52",  
        "ens85": "192.168.145.53",  
        "ens86": "192.168.145.54",  
        "ens87": "192.168.145.55",  
        "ens88": "192.168.145.56",  
        "ens89": "192.168.145.57",  
        "ens90": "192.168.145.58",  
        "ens91": "192.168.145.59",  
        "ens92": "192.168.145.60",  
        "ens93": "192.168.145.61",  
        "ens94": "192.168.145.62",  
        "ens95": "192.168.145.63",  
        "ens96": "192.168.145.64",  
        "ens97": "192.168.145.65",  
        "ens98": "192.168.145.66",  
        "ens99": "192.168.145.67",  
        "ens100": "192.168.145.68",  
        "ens101": "192.168.145.69",  
        "ens102": "192.168.145.70",  
        "ens103": "192.168.145.71",  
        "ens104": "192.168.145.72",  
        "ens105": "192.168.145.73",  
        "ens106": "192.168.145.74",  
        "ens107": "192.168.145.75",  
        "ens108": "192.168.145.76",  
        "ens109": "192.168.145.77",  
        "ens110": "192.168.145.78",  
        "ens111": "192.168.145.79",  
        "ens112": "192.168.145.80",  
        "ens113": "192.168.145.81",  
        "ens114": "192.168.145.82",  
        "ens115": "192.168.145.83",  
        "ens116": "192.168.145.84",  
        "ens117": "192.168.145.85",  
        "ens118": "192.168.145.86",  
        "ens119": "192.168.145.87",  
        "ens120": "192.168.145.88",  
        "ens121": "192.168.145.89",  
        "ens122": "192.168.145.90",  
        "ens123": "192.168.145.91",  
        "ens124": "192.168.145.92",  
        "ens125": "192.168.145.93",  
        "ens126": "192.168.145.94",  
        "ens127": "192.168.145.95",  
        "ens128": "192.168.145.96",  
        "ens129": "192.168.145.97",  
        "ens130": "192.168.145.98",  
        "ens131": "192.168.145.99",  
        "ens132": "192.168.145.100",  
        "ens133": "192.168.145.101",  
        "ens134": "192.168.145.102",  
        "ens135": "192.168.145.103",  
        "ens136": "192.168.145.104",  
        "ens137": "192.168.145.105",  
        "ens138": "192.168.145.106",  
        "ens139": "192.168.145.107",  
        "ens140": "192.168.145.108",  
        "ens141": "192.168.145.109",  
        "ens142": "192.168.145.110",  
        "ens143": "192.168.145.111",  
        "ens144": "192.168.145.112",  
        "ens145": "192.168.145.113",  
        "ens146": "192.168.145.114",  
        "ens147": "192.168.145.115",  
        "ens148": "192.168.145.116",  
        "ens149": "192.168.145.117",  
        "ens150": "192.168.145.118",  
        "ens151": "192.168.145.119",  
        "ens152": "192.168.145.120",  
        "ens153": "192.168.145.121",  
        "ens154": "192.168.145.122",  
        "ens155": "192.168.145.123",  
        "ens156": "192.168.145.124",  
        "ens157": "192.168.145.125",  
        "ens158": "192.168.145.126",  
        "ens159": "192.168.145.127",  
        "ens160": "192.168.145.128",  
        "ens161": "192.168.145.129",  
        "ens162": "192.168.145.130",  
        "ens163": "192.168.145.131",  
        "ens164": "192.168.145.132",  
        "ens165": "192.168.145.133",  
        "ens166": "192.168.145.134",  
        "ens167": "192.168.145.135",  
        "ens168": "192.168.145.136",  
        "ens169": "192.168.145.137",  
        "ens170": "192.168.145.138",  
        "ens171": "192.168.145.139",  
        "ens172": "192.168.145.140",  
        "ens173": "192.168.145.141",  
        "ens174": "192.168.145.142",  
        "ens175": "192.168.145.143",  
        "ens176": "192.168.145.144",  
        "ens177": "192.168.145.145",  
        "ens178": "192.168.145.146",  
        "ens179": "192.168.145.147",  
        "ens180": "192.168.145.148",  
        "ens181": "192.168.145.149",  
        "ens182": "192.168.145.150",  
        "ens183": "192.168.145.151",  
        "ens184": "192.168.145.152",  
        "ens185": "192.168.145.153",  
        "ens186": "192.168.145.154",  
        "ens187": "192.168.145.155",  
        "ens188": "192.168.145.156",  
        "ens189": "192.168.145.157",  
        "ens190": "192.168.145.158",  
        "ens191": "192.168.145.159",  
        "ens192": "192.168.145.160",  
        "ens193": "192.168.145.161",  
        "ens194": "192.168.145.162",  
        "ens195": "192.168.145.163",  
        "ens196": "192.168.145.164",  
        "ens197": "192.168.145.165",  
        "ens198": "192.168.145.166",  
        "ens199": "192.168.145.167",  
        "ens200": "192.168.145.168",  
        "ens201": "192.168.145.169",  
        "ens202": "192.168.145.170",  
        "ens203": "192.168.145.171",  
        "ens204": "192.168.145.172",  
        "ens205": "192.168.145.173",  
        "ens206": "192.168.145.174",  
        "ens207": "192.168.145.175",  
        "ens208": "192.168.145.176",  
        "ens209": "192.168.145.177",  
        "ens210": "192.168.145.178",  
        "ens211": "192.168.145.179",  
        "ens212": "192.168.145.180",  
        "ens213": "192.168.145.181",  
        "ens214": "192.168.145.182",  
        "ens215": "192.168.145.183",  
        "ens216": "192.168.145.184",  
        "ens217": "192.168.145.185",  
        "ens218": "192.168.145.186",  
        "ens219": "192.168.145.187",  
        "ens220": "192.168.145.188",  
        "ens221": "192.168.145.189",  
        "ens222": "192.168.145.190",  
        "ens223": "192.168.145.191",  
        "ens224": "192.168.145.192",  
        "ens225": "192.168.145.193",  
        "ens226": "192.168.145.194",  
        "ens227": "192.168.145.195",  
        "ens228": "192.168.145.196",  
        "ens229": "192.168.145.197",  
        "ens230": "192.168.145.198",  
        "ens231": "192.168.145.199",  
        "ens232": "192.168.145.200",  
        "ens233": "192.168.145.201",  
        "ens234": "192.168.145.202",  
        "ens235": "192.168.145.203",  
        "ens236": "192.168.145.204",  
        "ens237": "192.168.145.205",  
        "ens238": "192.168.145.206",  
        "ens239": "192.168.145.207",  
        "ens240": "192.168.145.208",  
        "ens241": "192.168.145.209",  
        "ens242": "192.168.145.210",  
        "ens243": "192.168.145.211",  
        "ens244": "192.168.145.212",  
        "ens245": "192.168.145.213",  
        "ens246": "192.168.145.214",  
        "ens247": "192.168.145.215",  
        "ens248": "192.168.145.216",  
        "ens249": "192.168.145.217",  
        "ens250": "192.168.145.218",  
        "ens251": "192.168.145.219",  
        "ens252": "192.168.145.220",  
        "ens253": "192.168.145.221",  
        "ens254": "192.168.145.222",  
        "ens255": "192.168.145.223",  
        "ens256": "192.168.145.224",  
        "ens257": "192.168.145.225",  
        "ens258": "192.168.145.226",  
        "ens259": "192.168.145.227",  
        "ens260": "192.168.145.228",  
        "ens261": "192.168.145.229",  
        "ens262": "192.168.145.230",  
        "ens263": "192.168.145.231",  
        "ens264": "192.168.145.232",  
        "ens265": "192.168.145.233",  
        "ens266": "192.168.145.234",  
        "ens267": "192.168.145.235",  
        "ens268": "192.168.145.236",  
        "ens269": "192.168.145.237",  
        "ens270": "192.168.145.238",  
        "ens271": "192.168.145.239",  
        "ens272": "192.168.145.240",  
        "ens273": "192.168.145.241",  
        "ens274": "192.168.145.242",  
        "ens275": "192.168.145.243",  
        "ens276": "192.168.145.244",  
        "ens277": "192.168.145.245",  
        "ens278": "192.168.145.246",  
        "ens279": "192.168.145.247",  
        "ens280": "192.168.145.248",  
        "ens281": "192.168.145.249",  
        "ens282": "192.168.145.250",  
        "ens283": "192.168.145.251",  
        "ens284": "192.168.145.252",  
        "ens285": "192.168.145.253",  
        "ens286": "192.168.145.254",  
        "ens287": "192.168.145.255",  
        "ens288": "192.168.145.256",  
        "ens289": "192.168.145.257",  
        "ens290": "192.168.145.258",  
        "ens291": "192.168.145.259",  
        "ens292": "192.168.145.260",  
        "ens293": "192.168.145.261",  
        "ens294": "192.168.145.262",  
        "ens295": "192.168.145.263",  
        "ens296": "192.168.145.264",  
        "ens297": "192.168.145.265",  
        "ens298": "192.168.145.266",  
        "ens299": "192.168.145.267",  
        "ens300": "192.168.145.268",  
        "ens301": "192.168.145.269",  
        "ens302": "192.168.145.270",  
        "ens303": "192.168.145.271",  
        "ens304": "192.168.145.272",  
        "ens305": "192.168.145.273",  
        "ens306": "192.168.145.274",  
        "ens307": "192.168.145.275",  
        "ens308": "192.168.145.276",  
        "ens309": "192.168.145.277",  
        "ens310": "192.168.145.278",  
        "ens311": "192.168.145.279",  
        "ens312": "192.168.145.280",  
        "ens313": "192.168.145.281",  
        "ens314": "192.168.145.282",  
        "ens315": "192.168.145.283",  
        "ens316": "192.168.145.284",  
        "ens317": "192.168.145.285",  
        "ens318": "192.168.145.286",  
        "ens319": "192.168.145.287",  
        "ens320": "192.168.145.288",  
        "ens321": "192.168.145.289",  
        "ens322": "192.168.145.290",  
        "ens323": "192.168.145.291",  
        "ens324": "192.168.145.292",  
        "ens325": "192.168.145.293",  
        "ens326": "192.168.145.294",  
        "ens327": "192.168.145.295",  
        "ens328": "192.168.145.296",  
        "ens329": "192.168.145.297",  
        "ens330": "192.168.145.298",  
        "ens331": "192.168.145.299",  
        "ens332": "192.168.145.300",  
        "ens333": "192.168.145.301",  
        "ens334": "192.168.145.302",  
        "ens335": "192.168.145.303",  
        "ens336": "192.168.145.304",  
        "ens337": "192.168.145.305",  
        "ens338": "192.168.145.306",  
        "ens339": "192.168.145.307",  
        "ens340": "192.168.145.308",  
        "ens341": "192.168.145.309",  
        "ens342": "192.168.145.310",  
        "ens343": "192.168.145.311",  
        "ens344": "192.168.145.312",  
        "ens345": "192.168.145.313",  
        "ens346": "192.168.145.314",  
        "ens347": "192.168.145.315",  
        "ens348": "192.168.145.316",  
        "ens349": "192.168.145.317",  
        "ens350": "192.168.145.318",  
        "ens351": "192.168.145.319",  
        "ens352": "192.168.145.320",  
        "ens353": "192.168.145.321",  
        "ens354": "192.168.145.322",  
        "ens355": "192.168.145.323",  
        "ens356": "192.168.145.324",  
        "ens357": "192.168.145.325",  
        "ens358": "192.168.145.326",  
        "ens359": "192.168.145.327",  
        "ens360": "192.168.145.328",  
        "ens361": "192.168.145.329",  
        "ens362": "192.168.145.330",  
        "ens363": "192.168.145.331",  
        "ens364": "192.168.145.332",  
        "ens365": "192.168.145.333",  
        "ens366": "192.168.145.334",  
        "ens367": "192.168.145.335",  
        "ens368": "192.168.145.336",  
        "ens369": "192.168.145.337",  
        "ens370": "192.168.145.338",  
        "ens371": "192.168.145.339",  
        "ens372": "192.168.145.340",  
        "ens373": "192.168.145.341",  
        "ens374": "192.168.145.342",  
        "ens375": "192.168.145.343",  
        "ens376": "192.168.145.344",  
        "ens377": "192.168.145.345",  
        "ens378": "192.168.145.346",  
        "ens379": "192.168.145.347",  
        "ens380": "192.168.145.348",  
        "ens381": "192.168.145.349",  
        "ens382": "192.168.145.350",  
        "ens383": "192.168.145.351",  
        "ens384": "192.168.145.352",  
        "ens385": "192.168.145.353",  
        "ens386": "192.168.145.354",  
        "ens387": "192.168.145.355",  
        "ens388": "192.168.145.356",  
        "ens389": "192.168.145.357",  
        "ens390": "192.168.145.358",  
        "ens391": "192.168.145.359",  
        "ens392": "192.168.145.360",  
        "ens393": "192.168.145.361",  
        "ens394": "192.168.145.362",  
        "ens395": "192.168.145.363",  
        "ens396": "192.168.145.364",  
        "ens397": "192.168.145.365",  
        "ens398": "192.168.145.366",  
        "ens399": "192.168.145.367",  
        "ens400": "192.168.145.368",  
        "ens401": "192.168.145.369",  
        "ens402": "192.168.145.370",  
        "ens403": "192.168.145.371",  
        "ens404": "192.168.145.372",  
        "ens405": "192.168.145.373",  
        "ens406": "192.168.145.374",  
        "ens407": "192.168.145.375",  
        "ens408": "192.168.145.376",  
        "ens409": "192.168.145.377",  
        "ens410": "192.168.145.378",  
        "ens411": "192.168.145
```

Ansible Tutorial For Beginners | Using Variables and Facts

```
"ansible_distribution_release": "Core",
"ansible_distribution_version": "7.6",
"ansible_dns": [
    "nameservers": [
        "192.168.145.2"
    ],
    "search": [
        "localdomain"
    ]
},
"ansible_domain": "localdomain",
"ansible_effective_group_id": 0,
"ansible_effective_user_id": 0,
"ansible_ens33": [
    "active": true,
    "device": "ens33",
    "features": [
        "busy_poll": "off [fixed]",
        "fcoe_mtu": "off [fixed]",
        "generic_receive_offload": "on",
        "generic_segmentation_offload": "on",
        "highdma": "off [fixed]",
        "hw_tc_offload": "off [fixed]",
        "l2_fwd_offload": "off [fixed]",
        "large_receive_offload": "off [fixed]",
        "loopback": "off [fixed]",
        "netns_local": "off [fixed]",
        "ntuple_filters": "off [fixed]",
        "receive_hashing": "off [fixed]",
        "rx_all": "off",
        "rx_checksumming": "off",
        "rx_fcs": "off"
    ],
    "rx_gro_hw": "off [fixed]",
    "tx_gro_hw": "off [fixed]"
]
```

Scroll for details

◀ ▶ ⏪ ⏩ 🔍 18:56 / 25:09

CC HD

Ahsible Tutorial For Beginners | Using Variables and Facts



```
"tx_vlan_offload": "on [fixed]",
"tx_vlan_stag_hw_insert": "off [fixed]",
"udp_fragmentation_offload": "off [fixed]",
"vlan_challenged": "off [fixed]"
},
"hw_timestamp_filters": [],
"ipv4": [
    "address": "192.168.145.146",
    "broadcast": "192.168.145.255",
    "netmask": "255.255.255.0",
    "network": "192.168.145.0"
},
"ipv6": [
    {
        "address": "fe80::1245:15f8:cae4:d2e",
        "prefix": "64",
        "scope": "link"
    }
],
"macaddress": "00:0c:29:ee:1c:fe",
"module": "e1000",
"mtu": 1500,
"pciid": "0000:02:01.0",
"promisc": false,
"speed": 1000,
"timestamping": [
    "tx_software",
    "rx_software",
    "software"
],
"type": "ether"
```

[<] [>] [D] 19:01 / 25:09

Scroll for details



Ansible Tutorial For Beginners | Using Variables and Facts

```
:*.mp3=01:36:*.mpc=01:36:*.ogg=01:36:*.ra=01:36:*.wav=01:36:*.axa=01:36:*.oga=01:36:*.spx=01:36:*.xspf=01:36:*,  
    "MAIL": "/var/mail/root",  
    "PATH": "/usr/local/sbin:/usr/local/bin:/usr/sbin:/usr/bin",  
    "PWD": "/root",  
    "SELINUX_LEVEL_REQUESTED": "",  
    "SELINUX_ROLE_REQUESTED": "",  
    "SELINUX_USE_CURRENT_RANGE": "",  
    "SHELL": "/bin/bash",  
    "SHLVL": "2",  
    "SSH_CLIENT": "192.168.145.144 40758 22",  
    "SSH_CONNECTION": "192.168.145.144 40758 192.168.145.146 22",  
    "SSH_TTY": "/dev/pts/1",  
    "TERM": "xterm",  
    "USER": "root",  
    "XDG_RUNTIME_DIR": "/run/user/0",  
    "XDG_SESSION_ID": "7",  
    "_": "/usr/bin/python"  
},  
"ansible_fibre_channel_wwn": [],  
"ansible_fips": false,  
"ansible_form_factor": "Other",  
"ansible_fqdn": "localhost.localdomain",  
"ansible_hostname": "localhost",  
"ansible_hostqn": "",  
"ansible_interfaces": [  
    "lo",  
    "ens33"  
],  
"ansible_is_chroot": false,  
"ansible_iscsi_iqn": "",  
"ansible_kernel": "3.10.0-957.el7.x86_64",  
"ansible_kernel_version": "#1 SMP Thu Nov 9 23:29:32 UTC 2017",  
"ansible_lo": [
```

Scroll for details

◀ ▶ ⏪ ⏩ ⏴ 19:14 / 25:09

CC HD

Ahsible Tutorial For Beginners | Using Variables and Facts

```
        },
        "vgs": {
            "centos": {
                "free_g": "0",
                "num_lvs": "2",
                "num_pvs": "1",
                "size_g": "19.00"
            }
        },
        "ansible_machine": "x86_64",
        "ansible_machine_id": "c57c4dba3d3e452183adff9cb777246d",
        "ansible_memfree_mb": 197,
        "ansible_memory_mb": {
            "nocache": {
                "free": 403,
                "used": 569
            },
            "real": {
                "free": 197,
                "total": 972,
                "used": 775
            },
            "swap": {
                "cached": 0,
                "free": 2047,
                "total": 2047,
                "used": 0
            }
        },
        "ansible_mounttotal_mb": 972,
        "ansible_mounts": [
            {
                "device": "/dev/sda1",
                "fs_type": "ext4",
                "mount": "/",
                "size_gb": "19.00"
            }
        ]
    }
}
```

Scroll for details

◀ ▶ ⏪ ⏩ 🔍 19:23 / 25:09

CC HD

Ahsible Tutorial For Beginners | Using Variables and Facts

```
        },
        "vgs": {
            "centos": {
                "free_g": "0",
                "num_lvs": "2",
                "num_pvs": "1",
                "size_g": "19.00"
            }
        },
        "ansible_machine": "x86_64",
        "ansible_machine_id": "c57c4dba3d3e452183adff9cb777246d",
        "ansible_memfree_mb": 197,
        "ansible_memory_mb": {
            "nocache": {
                "free": 403,
                "used": 569
            },
            "real": {
                "free": 197,
                "total": 972,
                "used": 775
            },
            "swap": {
                "cached": 0,
                "free": 2047,
                "total": 2047,
                "used": 0
            }
        },
        "ansible_mounttotal_mb": 972,
        "ansible_mounts": [
            {
                "device": "/dev/sda1",
                "fs_type": "ext4",
                "mount": "/",
                "size": 19700
            }
        ]
    }
}
```

Scroll for details

◀ ▶ ⏪ ⏩ 🔍 19:23 / 25:09

CC HD

Ahsible Tutorial For Beginners | Using Variables and Facts

```
        },
        "vgs": {
            "centos": {
                "free_g": "0",
                "num_lvs": "2",
                "num_pvs": "1",
                "size_g": "19.00"
            }
        },
        "ansible_machine": "x86_64",
        "ansible_machine_id": "c57c4dba3d3e452183adff9cb777246d",
        "ansible_memfree_mb": 197,
        "ansible_memory_mb": [
            "nocache": [
                "free": 409,
                "used": 569
            ],
            "real": [
                "free": 197,
                "total": 972,
                "used": 775
            ],
            "swap": [
                "cached": 0,
                "free": 2047,
                "total": 2047,
                "used": 0
            ]
        ],
        "ansible_mounttotal_mb": 972,
        "ansible_mounts": [
            {
                "device": "/dev/sda1",
                "fs_type": "ext4",
                "mountpoint": "/",
                "size_gb": "19.00"
            }
        ]
    }
}
```

Scroll for details

◀ ▶ ⏪ ⏩ 19:26 / 25:09

CC HD

Ahsible Tutorial For Beginners | Using Variables and Facts

```
        },
        {
            "block_available": 3414947,
            "block_size": 4096,
            "block_total": 4452864,
            "block_used": 1037917,
            "device": "/dev/mapper/centos-root",
            "fstype": "xfs",
            "inode_available": 8880056,
            "inode_total": 8910848,
            "inode_used": 30792,
            "mount": "/",
            "options": "rw,seclabel,relatime,attr2,inode64,noquota",
            "size_available": 13987622912,
            "size_total": 18238930944,
            "uuid": "29afdddb6-e829-4d3b-a377-2f4901232fdf"
        }
    ],
    "ansible_nodename": "localhost.localdomain",
    "ansible_ps_family": "RedHat",
    "ansible_pkg_mgr": "yum",
    "ansible_proc_cmdline": [
        "BOOT_IMAGE": "/vmlinuz-3.10.0-957.el7.x86_64",
        "LANG": "en_US.UTF-8",
        "crashkernel": "auto",
        "quiet": true,
        "rd.lvm.lv": [
            "centos/root",
            "centos/swap"
        ],
        "rhgb": true,
        "root": true,
        "root": "/dev/mapper/centos-root"
    ]
}
```

Scroll for details

◀ ▶ ⏪ ⏩ 🔍 19:31 / 25:09

CC HD

Ahsible Tutorial For Beginners | Using Variables and Facts

```
        },
        "block_available": 3414947,
        "block_size": 4096,
        "block_total": 4452864,
        "block_used": 1037917,
        "device": "/dev/mapper/centos-root",
        "fstype": "xfs",
        "inode_available": 8880056,
        "inode_total": 8910848,
        "inode_used": 30792,
        "mount": "/",
        "options": "rw,seclabel,relatime,attr2,inode64,noquota",
        "size_available": 13987622912,
        "size_total": 18238930944,
        "uuid": "29afdd6-e829-4d3b-a377-2f4901232fdf"
    },
    "ansible_nodename": "localhost.localdomain",
    "ansible_os_family": "RedHat",
    "ansible_pkg_mgr": "yum",
    "ansible_proc_cmdline": {
        "BOOT_IMAGE": "/vmlinuz-3.10.0-957.el7.x86_64",
        "LANG": "en_US.UTF-8",
        "crashkernel": "auto",
        "quiet": true,
        "rd.lvm.lv": [
            "centos/root",
            "centos/swap"
        ],
        "rhgb": true,
        "root": true,
        "root": "/dev/mapper/centos-root"
    }
}
```

Scroll for details

◀ ▶ ⏪ ⏩ 19:33 / 25:09

CC HD

Ahsible Tutorial For Beginners | Using Variables and Facts

```
"cap_mknod",
"cap_lease",
"cap_audit_write",
"cap_audit_control",
"cap_setfcap",
"cap_mac_override",
"cap_mac_admin",
"cap_syslog",
"35",
"36+ep"
],
"ansible_system_capabilities_enforced": "True",
"ansible_system_vendor": "VMware, Inc.",
"ansible_uptime_seconds": 1573,
"ansible_user_dir": "/root",
"ansible_user_gecos": "root",
"ansible_user_gid": 0,
"ansible_user_id": "root",
"ansible_user_shell": "/bin/bash",
"ansible_user_uid": 0,
"ansible_userspace_architecture": "x86_64",
"ansible_userspace_bits": "64",
"ansible_virtualization_role": "guest",
"ansible_virtualization_type": "VMware",
"discovered_interpreter_python": "/usr/bin/python",
"gather_subset": [
    "all"
],
"module_setup": true
},
"changed": false
}
```

[root@ansible1 ansiblevariables]# ansible testservers -fScroll for detailsfilter='*family*' [

Ansible Tutorial For Beginners | Using Variables and Facts



```
  "35",
  "36+ep"
],
"ansible_system_capabilities_enforced": "True",
"ansible_system_vendor": "VMware, Inc.",
"ansible_uptime_seconds": 1573,
"ansible_user_dir": "/root",
"ansible_user_gecos": "root",
"ansible_user_gid": 0,
"ansible_user_id": "root",
"ansible_user_shell": "/bin/bash",
"ansible_user_uid": 0,
"ansible_userspace_architecture": "x86_64",
"ansible_userspace_bits": "64",
"ansible_virtualization_role": "guest",
"ansible_virtualization_type": "VMware",
"discovered_interpreter_python": "/usr/bin/python",
"gather_subset": [
  "all"
],
"module_setup": true
},
"changed": false
}
[root@ansible1 ansiblevariables]# ansible testserver -m setup -a " filter=*family*"
192.168.145.146 | SUCCESS => {
  "ansible_facts": [
    "ansible_os_family": "RedHat",
    "discovered_interpreter_python": "/usr/bin/python"
  ],
  "changed": false
}
```

[root@ansible1 ansiblevariables]#

Scroll for details



Ahsible Tutorial For Beginners | Using Variables and Facts



```
"35",
"36+ep"
],
"ansible_system_capabilities_enforced": "True",
"ansible_system_vendor": "VMware, Inc.",
"ansible_uptime_seconds": 1573,
"ansible_user_dir": "/root",
"ansible_user_gecos": "root",
"ansible_user_gid": 0,
"ansible_user_id": "root",
"ansible_user_shell": "/bin/bash",
"ansible_user_uid": 0,
"ansible_userspace_architecture": "x86_64",
"ansible_userspace_bits": "64",
"ansible_virtualization_role": "guest",
"ansible_virtualization_type": "VMware",
"discovered_interpreter_python": "/usr/bin/python",
"gather_subset": [
    "all"
],
"module_setup": true
},
"changed": false
}
[root@ansible1 ansiblevariables]# ansible testservers -m setup -a " filter=*family*"
192.168.145.146 | SUCCESS => {
    "ansible_facts": [
        "ansible os family": "RedHat",
        "discovered_interpreter_python": "/usr/bin/python"
    ],
    "changed": false
}
```

[root@ansible1 ansiblevariables]#

Scroll for details



Ahsible Tutorial For Beginners | Using Variables and Facts



```
  "35",
  "36+ep"
],
"ansible_system_capabilities_enforced": "True",
"ansible_system_vendor": "VMware, Inc.",
"ansible_uptime_seconds": 1573,
"ansible_user_dir": "/root",
"ansible_user_gecos": "root",
"ansible_user_gid": 0,
"ansible_user_id": "root",
"ansible_user_shell": "/bin/bash",
"ansible_user_uid": 0,
"ansible_userspace_architecture": "x86_64",
"ansible_userspace_bits": "64",
"ansible_virtualization_role": "guest",
"ansible_virtualization_type": "VMware",
"discovered_interpreter_python": "/usr/bin/python",
"gather_subset": [
  "all"
],
"module_setup": true
},
"changed": false
}
[root@ansible1 ansiblevariables]# ansible testservers -m setup -a " filter=*family*"
192.168.145.146 | SUCCESS => {
  "ansible_facts": [
    "ansible_os_family": "RedHat",
    "discovered_interpreter_python": "/usr/bin/python"
  ],
  "changed": false
}
```

[root@ansible1 ansiblevariables]#

Scroll for details



Ahible Tutorial For Beginners | Using Variables and Facts



```
[root@ansible1 ansiblevariables]# ansible testservers -m setup -a "filter=*family*"
192.168.145.146 | SUCCESS => {
    "ansible_facts": [
        "ansible_os_family": "RedHat",
        "discovered_interpreter_python": "/usr/bin/python"
    ],
    "changed": false
}
[root@ansible1 ansiblevariables]# ansible testservers -m setup -a "filter=*os*"
192.168.145.146 | SUCCESS => {
    "ansible_facts": [
        "ansible_bios_date": "07/29/2019",
        "ansible_bios_version": "6.00",
        "ansible_hostname": "localhost",
        "ansible_hostnqn": "",
        "ansible_os_family": "RedHat",
        "ansible_ssh_host_key_ecdsa_public": "AAAAE2VjZHNhLXNoYTItbmlzdHAyNTYAAAIBmlzdHAyNTYAAABBA6xSwhmzJDmxWMPr/6wpqiu
s+eMM+gBWEZW8hVqzo/J8Bg/yKY/C18kTHI9u/UCMZUR0K2wVmrvjZl7JsHcbr30=",
        "ansible_ssh_host_key_ed25519_public": "AAAAC3NzaC1lZDI1NTE5AAAAICchmExgq89FVfMRRPX8PBF6u9zh0zt35pIliqZcZNm4",
        "ansible_ssh_host_key_rsa_public": "AAAAB3NzaC1yc2EAAAQABAAQCo2mbXHTLqHR5ms1WhbeYcEqicpAolS9Tem02a9wadABES0S
pCzXbZ6EvVDbN37Kq/Rlj3yyVBvsFC2/T+ynhTqT6TTs1rdbLtRTDInpmdDI01ByQmFcNK0U89pMGGII3+Lhi+CAHm+S0l5j0YbBwWwmEmuaM4MyCfTu4DC6Z6
Z29UdmIFESS5wrTDlgXcfAnyoD79wvADkEaBLgiUGLfb3FI0b21QHoFlftZSF68zWHvX5ytNNNDyHy09hVPbfi8KRSkZiWTQ5DmZFswN44XWrCpaLGzMRxNe0i4
jpdlDr6FM/VX/uAIY0QQI4e8WyqYM2e6N1JJmHtql+lnp7wTLF",
        "ansible_user_gecos": "root",
        "discovered_interpreter_python": "/usr/bin/python"
    ],
    "changed": false
}
```

[root@ansible1 ansiblevariables]#

Scroll for details



Ahnsible Tutorial For Beginners | Using Variables and Facts

```
[root@ansible1 ansiblevariables]# ansible testservers -m setup -a "filter=*family*"
192.168.145.146 | SUCCESS => {
    "ansible_facts": {
        "ansible_os_family": "RedHat",
        "discovered_interpreter_python": "/usr/bin/python"
    },
    "changed": false
}
[root@ansible1 ansiblevariables]# ansible testservers -m setup -a "filter=*os*"
192.168.145.146 | SUCCESS => {
    "ansible_facts": {
        "ansible_bios_date": "07/29/2019",
        "ansible_bios_version": "6.00",
        "ansible_hostname": "localhost",
        "ansible_hostqn": "",
        "ansible_os_family": "RedHat",
        "ansible_ssh_host_key_ecdsa_public": "AAAAE2VjZHNhLXNoYTItbmlzdHAyNTYAAAIBmlzdHAyNTYAAABBA6xSwhmzJDmxWMPr/6wpqiu
s+eMM+gBWEW8hVqzo/J8Bg/yKY/C18kTHI9u/UCM2UR0K2wVmrrjZ17JsHcbr30=",
        "ansible_ssh_host_key_ed25519_public": "AAAAC3NzaC1lZDI1NTE5AAAAICchmExgg89FVfMRRPX8PBF6u9zh0zt35pIliqZcZNm4",
        "ansible_ssh_host_key_rsa_public": "AAAAB3NzaC1yc2EAAAQABAAQCo2mbXHTLqHR5ms1WhbeYcEqicpAols9Tem02a9wadABES05
pCzXbZ6EvVDbN37Kq/Rlj3yyVBvsFC2/T+ynhTqt6TTs1rdbLtRTDInpmddI01ByQmFcNK0U89pMGGII3+Lhi+CAHm+S0l5j0YbBwWwmEmuaM4MyCfTu4DC6Z6
Z29UdmIFES5wrTDlgXcfAnyoD79wvADkEaBLgIUGLfb3FI0b21QHoFlftZSF68zWHvX5ytNNNDyHy09hVPbfi8KRSkZiwtQ5DmZFswN44XWrCpaLGzMRxNe0i4
jpdlDr6FM/VX/uAIY0QQI4e8WyqYM2e6N1JJmHtql+lnp7wTLF",
        "ansible_user_gecos": "root",
        "discovered_interpreter_python": "/usr/bin/python"
    },
    "changed": false
}
```

20.42 / 25.09

Scroll for details

Ansible Tutorial For Beginners | Using Variables and Facts

```
1 ---  
2 - hosts: testservers  
3 vars:  
4   Color: [ Green, White, Red ]  
5     - Green  
6     - White  
7     - Red  
8  
9  
10 remote_user: root  
11 tasks:  
12   - name: Variable tutorial  
13     shell: echo " My Favourite Color is {{ Color[2] }} and the color i dont like is {{ Color[0] }} " > /root/ansiblevariable.  
14
```

Ansible Tutorial For Beginners | Using Variables and Facts



```
1 ---  
2 - hosts: testservers  
3   remote_user: root  
4   tasks:  
5     - name: Variable tutorial  
6       shell: echo " The hostname of this machine is {{ ansible_hostname }} and this machine belongs to OS family {{ ansible_os_
```

Ahible Tutorial For Beginners | Using Variables and Facts



```
[root@ansible1 ansiblevariables]# ls  
testnew.yaml test.yaml variable.txt  
[root@ansible1 ansiblevariables]# vi test1.yaml
```

I

Ahsible Tutorial For Beginners | Using Variables and Facts

```
---  
- hosts: testservers  
  remote_user: root  
  tasks:  
    - name: Variable tutorial  
      shell: echo " The hostname of this machine is {{ ansible_hostname }} and this machine belongs to OS family {{ ansible_os_family }} " > /root/ansiblevariable.txt
```

Ansible Tutorial For Beginners | Using Variables and Facts

```
[root@localhost ~]# ls
anaconda-ks.cfg
[root@localhost ~]# ls
anaconda-ks.cfg  ansiblevariable.txt
[root@localhost ~]# cat ansiblevariable.txt
My Favourite Color is Red and the color i dont like is Green
[root@localhost ~]# ls
anaconda-ks.cfg  ansiblevariable.txt
[root@localhost ~]# █
```

I

Ansible Tutorial For Beginners | Using Variables and Facts

```
[root@localhost ~]# ls
anaconda-ks.cfg
[root@localhost ~]# ls
anaconda-ks.cfg  ansiblevariable.txt
[root@localhost ~]# cat ansiblevariable.txt
My Favourite Color is Red and the color i dont like is Green
[root@localhost ~]# ls
anaconda-ks.cfg  ansiblevariable.txt
[root@localhost ~]# rm -f ansiblevariable.txt
[root@localhost ~]# clear
```



I

Ansible Tutorial For Beginners | Using Variables and Facts
[root@localhost ~]# ls
anaconda-ks.cfg
[root@localhost ~]#



I

◀ ▶ ▶| 🔍 22:34 / 25:09

Scroll for details

CC HD

Ahnsible Tutorial For Beginners | Using Variables and Facts



```
[root@ansible1 ansiblevariables]# ls  
testnew.yaml test.yaml variable.txt  
[root@ansible1 ansiblevariables]# vi test1.yaml  
[root@ansible1 ansiblevariables]# █
```

I

Ahible Tutorial For Beginners | Using Variables and Facts

```
[root@ansible1 ansiblevariables]# ls  
testnew.yaml test.yaml variable.txt  
[root@ansible1 ansiblevariables]# vi test1.yaml  
[root@ansible1 ansiblevariables]# ansible-playbook test1.yaml
```

Ahnsible Tutorial For Beginners | Using Variables and Facts



```
[root@ansible1 ansiblevariables]# ls
testnew.yaml test.yaml variable.txt
[root@ansible1 ansiblevariables]# vi test1.yaml
[root@ansible1 ansiblevariables]# ansible-playbook test1.yaml
ERROR! failed at splitting arguments, either an unbalanced jinja2 block or quotes: echo " The hostname of this machine is
{{ ansible_hostname }} and this machine belongs to OS family {{ ansible_os_family }} " > /root/ansiblevariable.txt
```

The error appears to be in '/root/ansiblevariables/test1.yaml': line 7, column 5, but may
be elsewhere in the file depending on the exact syntax problem.

The offending line appears to be:

```
tasks:
- name: Variable tutorial
  ^ here
[root@ansible1 ansiblevariables]#
```

Ahible Tutorial For Beginners | Using Variables and Facts



```
[root@ansible1 ansiblevariables]# ls  
testnew.yaml test.yaml variable.txt  
[root@ansible1 ansiblevariables]# vi test1.yaml  
[root@ansible1 ansiblevariables]# ansible-playbook test1.yaml  
ERROR! failed at splitting arguments, either an unbalanced jinja2 block or quotes: echo " The hostname of this machine is  
{{ ansible_hostname }} and this machine belongs to OS family {{ ansible_os_family }} " > /root/ansiblevariable.txt
```

The error appears to be in '/root/ansiblevariables/test1.yaml': line 7, column 5, but may
be elsewhere in the file depending on the exact syntax problem.

The offending line appears to be:

```
tasks:  
- name: Variable tutorial  
  ^ here  
[root@ansible1 ansiblevariables]# vi test1.yaml
```

I

Ahible Tutorial For Beginners | Using Variables and Facts



```
[root@ansible1 ansiblevariables]# ls  
testnew.yaml test.yaml variable.txt  
[root@ansible1 ansiblevariables]# vi test1.yaml  
[root@ansible1 ansiblevariables]# ansible-playbook test1.yaml  
ERROR! failed at splitting arguments, either an unbalanced jinja2 block or quotes: echo " The hostname of this machine is  
{{ ansible_hostname }} and this machine belongs to OS family {{ ansible_os_family }} " > /root/ansiblevariable.txt
```

The error appears to be in '/root/ansiblevariables/test1.yaml': line 7, column 5, but may
be elsewhere in the file depending on the exact syntax problem.

The offending line appears to be:

```
tasks:  
- name: Variable tutorial  
  ^ here  
[root@ansible1 ansiblevariables]# vi test1.yaml  
[root@ansible1 ansiblevariables]# ansible-playbook test1.yaml  
  
PLAY [testservers] *****  
  
TASK [Gathering Facts] *****  
ok: [192.168.145.146]  
  
TASK [Variable tutorial] *****  
changed: [192.168.145.146]  
  
PLAY RECAP *****  
192.168.145.146 : ok=2    changed=1    unreachable=0    failed=0    skipped=0    rescued=0    ignored=0  
  
[root@ansible1 ansiblevariables]#
```

Ansible Tutorial For Beginners | Using Variables and Facts



```
[root@localhost ~]# ls  
anaconda-ks.cfg  
[root@localhost ~]# ls  
anaconda-ks.cfg  ansiblevariable.txt  
[root@localhost ~]# cat ansiblevariable.txt  
The hostname of this machine is localhost and this machine belongs to OS family RedHat  
[root@localhost ~]#
```

I

Ansible Tutorial For Beginners | Using Variables and Facts



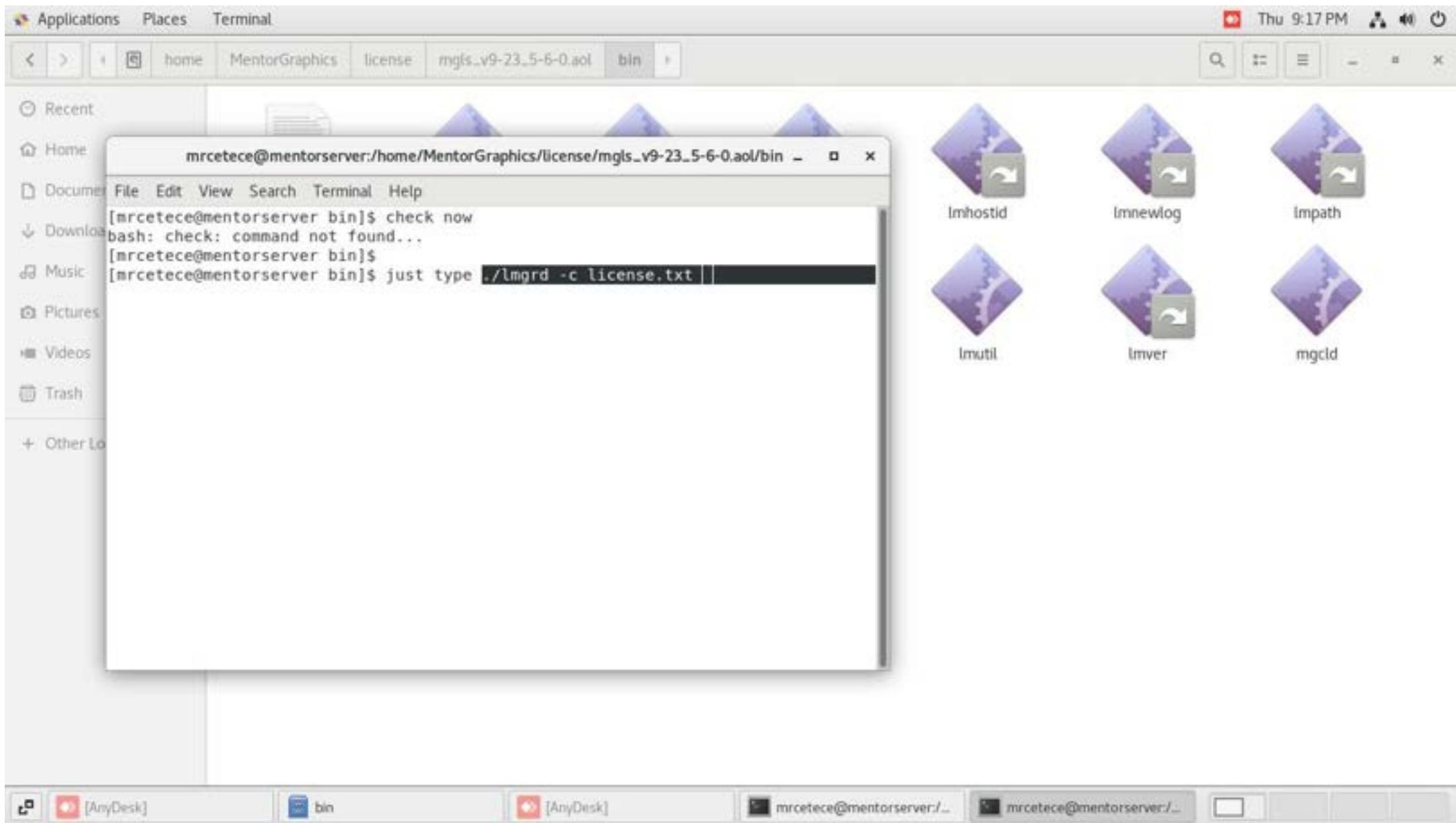
```
1 ---  
2 - hosts: testservers  
3   remote_user: root  
4   tasks:  
5     - name: Variable tutorial  
6       shell: echo " The hostname of this machine is {{ ansible_hostname }} and this machine belongs to OS family {{ ansible_os_
```

Ansible Tutorial For Beginners | Using Variables and Facts



```
[root@localhost ~]# ls  
anaconda-ks.cfg  
[root@localhost ~]# ls  
anaconda-ks.cfg  ansiblevariable.txt  
[root@localhost ~]# cat ansiblevariable.txt  
The hostname of this machine is localhost and this machine belongs to OS family RedHat  
[root@localhost ~]#
```

I





Home



Trash

```
mrcetece@mentorserver:/home/MentorGraphics/license/mgls_v9-23_5-6-0.aol/bin - □ ×  
File Edit View Search Terminal Help  
[mrcetece@mentorserver ~]$ cd /home/MentorGraphics/license/mgls_v9-23_5-6-0.aol/  
bin/  
[mrcetece@mentorserver bin]$ ./lmgrd -c license.txt  
21:21:57 (lmgrd) -----  
21:21:57 (lmgrd) Please Note:  
21:21:57 (lmgrd) This log is intended for debug purposes only.  
21:21:57 (lmgrd) In order to capture accurate license  
21:21:57 (lmgrd) usage data into an organized repository,  
21:21:57 (lmgrd) please enable report logging. Use Flexera's  
21:21:57 (lmgrd) software license administration solution,  
21:21:57 (lmgrd) FlexNet Manager, to readily gain visibility  
21:21:57 (lmgrd) into license usage data and to create  
21:21:57 (lmgrd) insightful reports on critical information like  
21:21:57 (lmgrd) license availability and usage. FlexNet Manager  
21:21:57 (lmgrd) can be fully automated to run these reports on  
21:21:57 (lmgrd) schedule and can be used to track license  
21:21:57 (lmgrd) servers and usage across a heterogeneous  
21:21:57 (lmgrd) network of servers including Windows NT, Linux  
21:21:57 (lmgrd)  
21:21:57 (lmgrd)
```

7

CENTOS



Home



Trash

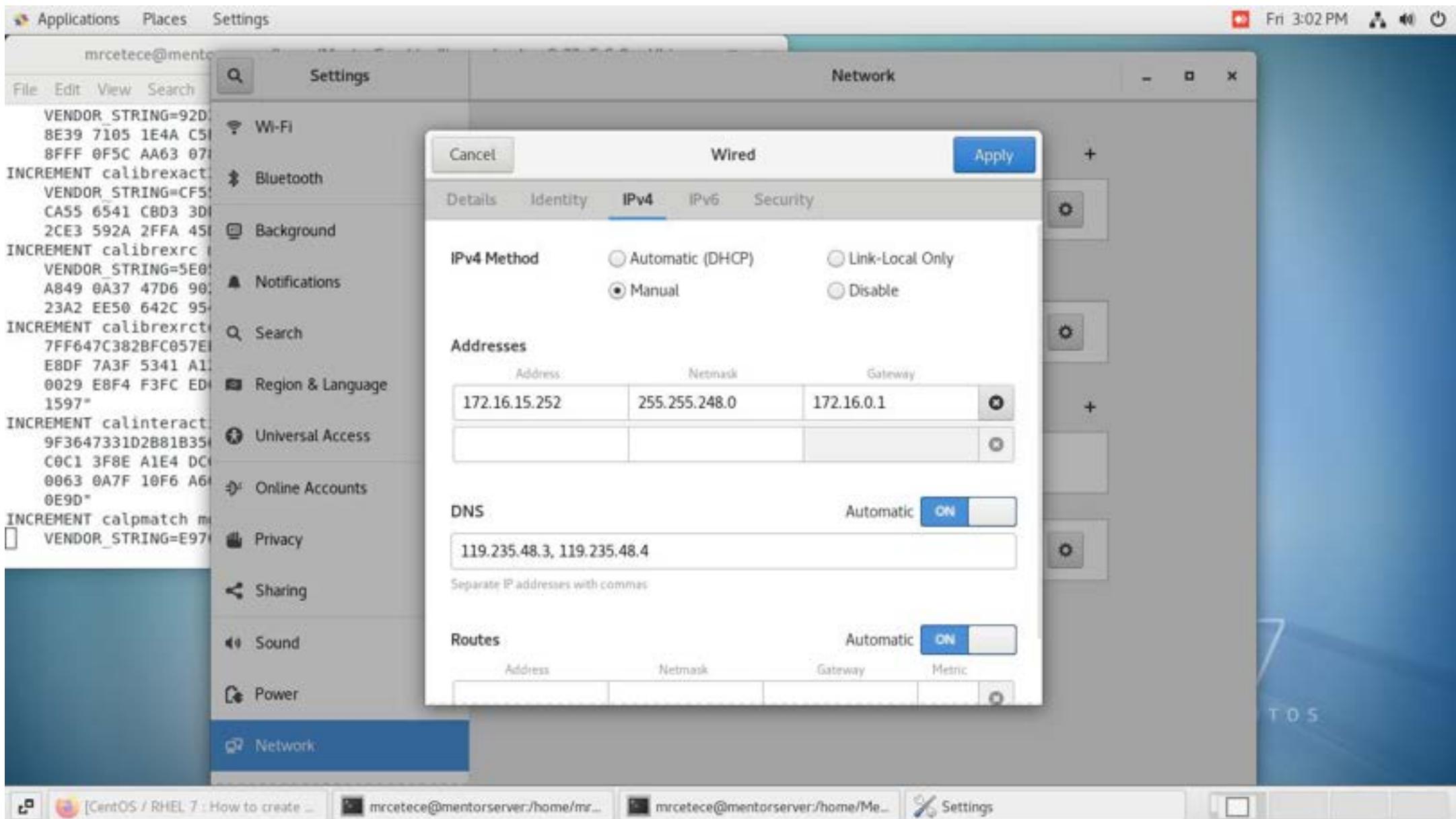


15 GB Volume

mrcetece@mentorserver:/home/MentorGraphics/license/mgls_v9-23_5-6-0.aol/bin - □ ×

```
File Edit View Search Terminal Help
[root@mentorserver bin]# pwd
/home/MentorGraphics/license/mgls_v9-23_5-6-0.aol/bin
[root@mentorserver bin]# █
```

CENTOS



Ansible Debug Module

- Debug module prints statements during execution and can be useful for debugging variables or expressions without necessarily halting the playbook.
- It is very useful to use Debug module in lot of cases along with When:.
- Debug module is supported in Windows Target as well.
- Debug module can have parameter msg or var

Ansible Debug Module

- Debug module prints statements during execution and can be useful for debugging variables or expressions without necessarily halting the playbook.
- It is very useful to use Debug module in lot of cases along with When:.
- Debug module is supported in Windows Target as well.
- Debug module can have parameter msg or var

Ansible Tutorial for Beginners | Ansible Debug Module and Register Variables



```
1
2 - hosts: testservers
3   tasks:
4
5     - name: printing Ip address
6       debug:
7         msg: "my system ip adress is {{ ansible_default_ipv4.address }}"
```

Ansible Tutorial for Beginners | Ansible Debug Module and Register Variables



```
1 ---
2   - hosts: testservers
3     tasks:
4       - name: executing sample command
5         shell: uptime
6         register: result
7
8       - name: printing variable
9         debug:
10           msg: "{{result}}"
```

Ansible Tutorial for Beginners | Ansible Debug Module and Register Variables



```
[root@ansible1 ansible]# ansible-playbook ansibledebug.yml
```

```
PLAY [testservers] *****

TASK [Gathering Facts] *****
ok: [192.168.145.146]

TASK [executing sample command] *****
changed: [192.168.145.146]

TASK [printing variable] *****
ok: [192.168.145.146] => {
    "msg": {
        "changed": true,
        "cmd": "uptime",
        "delta": "0:00:00.037291",
        "end": "2021-06-27 13:21:46.448060",
        "failed": false,
        "rc": 0,
        "start": "2021-06-27 13:21:46.410769",
        "stderr": "",
        "stderr_lines": [],
        "stdout": " 13:21:46 up 15 min,  1 user,  load average: 0.00, 0.04, 0.08",
        "stdout_lines": [
            " 13:21:46 up 15 min,  1 user,  load average: 0.00, 0.04, 0.08"
        ]
    }
}

PLAY RECAP *****
192.168.145.146 : ok=3    changed=1    unreachable=0    failed=0    skipped=0    rescued=0    ignored=0
```

```
[root@ansible1 ansible]#
```

Scroll for details



Ansible Tutorial for Beginners | Ansible Debug Module and Register Variables

```
[root@ansible1 ansible]# ansible-playbook ansibledebug.yml
```

```
PLAY [testservers] *****

TASK [Gathering Facts] *****
ok: [192.168.145.146]

TASK [executing sample command] *****
changed: [192.168.145.146]

TASK [printing variable] *****
ok: [192.168.145.146] => {
    "msg": {
        "changed": true,
        "cmd": "uptime",
        "delta": "0:00:00.037291",
        "end": "2021-06-27 13:21:46.448060",
        "failed": false,
        "rc": 0,
        "start": "2021-06-27 13:21:46.410769",
        "stderr": "",
        "stderr_lines": [],
        "stdout": " 13:21:46 up 15 min,  1 user,  load average: 0.00, 0.04, 0.08",
        "stdout_lines": [
            " 13:21:46 up 15 min,  1 user,  load average: 0.00, 0.04, 0.08"
        ]
    }
}

PLAY RECAP *****
192.168.145.146 : ok=3    changed=1    unreachable=0    failed=0    skipped=0    rescued=0    ignored=0
```

```
[root@ansible1 ansible]#
```

Scroll for details



Ansible Tutorial For Beginners

Ansible Debug Module

- Debug module prints statements during execution and can be useful for debugging variables or expressions without necessarily halting the playbook.
- It is very useful to use Debug module in lot of cases along with When:..
- Debug module is supported in Windows Target as well.
- Debug module can have parameter msg or var

Ansible Essentials : Registering Variables

- Register is used to create variables from the output of an Ansible task
- We can use registered variables in any later tasks in your play
- Registered variables are stored in memory. You cannot cache registered variables for use in future plays

Ansible Tutorial for Beginners | Ansible Debug Module and Register Variables



```
1  ---
2  - hosts: testservers
3  tasks:
4
5  - name: printing Ip address
6    debug:
7      msg: "my system ip adres is {{ ansible_default_ipv4.address }}"
```

Ansible Tutorial for Beginners | Ansible Debug Module and Register Variables



```
1 ...
2 hosts: testservers
3 tasks:
4
5   - name: printing ip address
6     debug:
7       msg: "my system ip address is {{ ansible_default_ipv4.address }}"
```

Ansible Tutorial for Beginners | Ansible Debug Module and Register Variables



```
1  ---
2  - hosts: testservers
3  tasks:
4
5  - name: printing Ip address
6    debug:
7      msg: "my system ip adres is {{ ansible_default_ipv4.address }}"
```

Ansible Tutorial for Beginners | Ansible Debug Module and Register Variables



```
1
2 - hosts: testservers
3 tasks:
4
5 - name: printing Ip address
6   debug:
7     msg: "my system ip adress is {{ ansible_default_ipv4.address }}"
```

Ansible Tutorial for Beginners | Ansible Debug Module

Suggested: Ansible Playbook Beginners Tutorial // Create Your Fir...



```
1
2 hosts: testservers
3 tasks:
4
5   - name: printing Ip address
6     debug:
7       msg: "My system ip address is {{ ansible_default_ipv4.address }}"
```

Ansible Tutorial for Beginners | Ansible Debug Module and Register Variables

[root@ansible1 ansible]# vi ansibledebug.yml



◀ ▶ ⏪ 🔍 5:38 / 16:29

Scroll for details



Ansible Tutorial for Beginners | Ansible Debug Module and Register Variables



```
---
- hosts: testservers
  tasks:
    - name: printing Ip address
      debug:
        msg: "my system ip adress is {{ ansible_default_ipv4.address }}"
```

```
[root@ansible ansible]# vi ansibledebug.yml
[root@ansible ansible]# ansible-playbook ansibledebug.yml

PLAY [testservers] *****

TASK [Gathering Facts] *****
ok: [192.168.145.146]

TASK [printing Ip address] *****
ok: [192.168.145.146] => {
    "msg": "my system ip adress is 192.168.145.146"
}

PLAY RECAP *****
192.168.145.146 : ok=2    changed=0    unreachable=0    failed=0    skipped=0    rescued=0    ignored=0

[root@ansible ansible]#
```

Ansible Tutorial for Beginners | Ansible Debug Module and Register Variables

```
[root@ansible1 ansible]# vi ansibledebug.yml
[root@ansible1 ansible]# ansible-playbook ansibledebug.yml

PLAY [testservers] *****

TASK [Gathering Facts] *****
ok: [192.168.145.146]

TASK [printing Ip address] *****
ok: [192.168.145.146] => {
    "msg": "my system ip address is 192.168.145.146"
}

PLAY RECAP *****
192.168.145.146 : ok=2    changed=0    unreachable=0    failed=0    skipped=0    rescued=0    ignored=0

[root@ansible1 ansible]#
```

Ansible Tutorial for Beginners | Ansible Debug Module and Register Variables



```
1
2 - hosts: testservers
3 tasks:
4
5 - name: printing Ip address
6   debug:
7     msg: "my system ip adres is {{ ansible_default_ipv4.address }}"
```

Ansible Tutorial for Beginners | Ansible Debug Module and Register Variables



```
1
2 - hosts: testservers
3 tasks:
4
5 - name: printing Ip address
6   debug:
7     msg: "My system ip adress is {{ ansible_default_ipv4.address }}"
```

Ansible Tutorial for Beginners | Ansible Debug Module and Register Variables



```
1  ---
2  - hosts: testservers
3  tasks:
4  - name: executing sample command
5  shell: uptime
6  register: result
7
8  - name: printing variable
9  debug:
10   msg: "{{result}}"
```

Ansible Tutorial for Beginners | Ansible Debug Module and Register Variables



```
1  ---
2  - hosts: testservers
3  tasks:
4  - name: executing sample command
5  shell: uptime
6  register: result
7
8  - name: printing variable
9  debug:
10   msg: "{{result}}"
```

Ansible Tutorial for Beginners | Ansible Debug Module and Register Variables



```
1 ---  
2   - hosts: testservers  
3     tasks:  
4       - name: executing sample command  
5         shell: uptime  
6         register: result  
7  
8       - name: printing variable  
9         debug:  
10        msg: "{{result}}"
```

Ansible Tutorial for Beginners | Ansible Debug Module and Register Variables



```
1 ---  
2 - hosts: testservers  
3 tasks:  
4   - name: executing sample command  
5     shell: uptime  
6     register: result  
7  
8   - name: printing variable  
9     debug:  
10    msg: "{{result}}"
```

Ansible Tutorial for Beginners | Ansible Debug Module and Register Variables
[root@ansible1 ansible]# rm -f ansibledebug.yml



1

◀ ▶ ▶| 🔍 8:16 / 16:29

Scroll for details

CC ⚙️ 🌐

Ansible Tutorial for Beginners | Ansible Debug Module and Register Variables

```
[root@ansible1 ansible]# rm -f ansibledebug.yml
[root@ansible1 ansible]# vi ansibledebug.yml
```



Ansible Tutorial for Beginners | Ansible Debug Module and Register Variables



```
---  
- hosts: testservers  
  tasks:  
    - name: executing sample command  
      shell: uptime  
      register: result  
  
    - name: printing variable  
      debug:  
        msg: "{{result}}"
```

Ansible Tutorial for Beginners | Ansible Debug Module and Register Variables



```
---  
- hosts: testservers  
  tasks:  
    - name: executing sample command  
      shell: uptime  
      register: result  
  
    - name: printing variable  
      debug:  
        msg: "{{result}}"
```

Ansible Tutorial for Beginners | Ansible Debug Module and Register Variables



```
[root@ansible ansible]# rm -f ansibledebug.yml  
[root@ansible ansible]# vi ansibledebug.yml  
[root@ansible ansible]# ansible-playbook ansibledebug.yml
```

Ansible Tutorial for Beginners | Ansible Debug Module and Register Variables



```
[root@ansible1 ansible]# ansible-playbook ansibledebug.yml
```

```
PLAY [testservers] *****

TASK [Gathering Facts] *****
ok: [192.168.145.146]

TASK [executing sample command] *****
changed: [192.168.145.146]

TASK [printing variable] *****
ok: [192.168.145.146] => {
    "msg": {
        "changed": true,
        "cmd": "uptime",
        "delta": "0:00:00.037291",
        "end": "2021-06-27 13:21:46.448060",
        "failed": false,
        "rc": 0,
        "start": "2021-06-27 13:21:46.410769",
        "stderr": "",
        "stderr_lines": [],
        "stdout": " 13:21:46 up 15 min,  1 user,  load average: 0.00, 0.04, 0.08",
        "stdout_lines": [
            " 13:21:46 up 15 min,  1 user,  load average: 0.00, 0.04, 0.08"
        ]
    }
}

PLAY RECAP *****
192.168.145.146 : ok=3    changed=1    unreachable=0    failed=0    skipped=0    rescued=0    ignored=0
```

```
[root@ansible1 ansible]#
```

Scroll for details



Ansible Tutorial for Beginners | Ansible Debug Module and Register Variables



```
[root@ansible1 ansible]# ansible-playbook ansibledebug.yml
```

```
PLAY [testservers] ****
```

```
TASK [Gathering Facts] ****  
ok: [192.168.145.146]
```

```
TASK [executing sample command] ****
```

```
changed: [192.168.145.146] !
```

```
TASK [printing variable] ****
```

```
ok: [192.168.145.146] => {
```

```
  "msg": {  
    "changed": true,  
    "cmd": "uptime",  
    "delta": "0:00:00.037291",  
    "end": "2021-06-27 13:21:46.448060",  
    "failed": false,  
    "rc": 0,  
    "start": "2021-06-27 13:21:46.410769",  
    "stderr": "",  
    "stderr_lines": [],  
    "stdout": " 13:21:46 up 15 min,  1 user,  load average: 0.00, 0.04, 0.08",  
    "stdout_lines": [  
      " 13:21:46 up 15 min,  1 user,  load average: 0.00, 0.04, 0.08"  
    ]  
  }  
}
```

```
PLAY RECAP ****
```

```
192.168.145.146 : ok=3    changed=1    unreachable=0    failed=0    skipped=0    rescued=0    ignored=0
```

```
[root@ansible1 ansible]#
```

Scroll for details

Ansible Tutorial for Beginners | Ansible Debug Module and Register Variables



```
[root@ansible1 ansible]# ansible-playbook ansibledebug.yml
```

```
PLAY [testservers] *****

TASK [Gathering Facts] *****
ok: [192.168.145.146]

TASK [executing sample command] *****
changed: [192.168.145.146]

TASK [printing variable] *****
ok: [192.168.145.146] => {
    "msg": [
        {
            "changed": true,
            "cmd": "uptime",
            "delta": "0:00:00.037291",
            "end": "2021-06-27 13:21:46.448060",
            "failed": false,
            "rc": 0,
            "start": "2021-06-27 13:21:46.410769",
            "stderr": "",
            "stderr_lines": [],
            "stdout": " 13:21:46 up 15 min,  1 user,  load average: 0.00, 0.04, 0.08",
            "stdout_lines": [
                " 13:21:46 up 15 min,  1 user,  load average: 0.00, 0.04, 0.08"
            ]
        }
    ]
}
```

```
PLAY RECAP *****
192.168.145.146 : ok=3    changed=1    unreachable=0    failed=0    skipped=0    rescued=0    ignored=0
```

```
[root@ansible1 ansible]#
```

Scroll for details



Ansible Tutorial for Beginners | Ansible Debug Module and Register Variables



```
1 ...
2 - hosts: testservers
3   tasks:
4     - name: executing sample command
5       shell: uptime
6       register: result
7
8     - name: printing variable
9       debug:
10      msg: "{{result}}"
```



```
1
2 - hosts: testservers
3 tasks:
4   - name: executing sample command
5     command: echo -e "This is the first line, \n This is second line"
6     register: result
7
8   - name: printing variable
9     debug:
10    var: result
```

Ansible Tutorial for Beginners | Ansible Debug Module and Register Variables



```
1 ---
2   - hosts: testservers
3     tasks:
4       - name: executing sample command
5         command: echo -e "This is the first line, \n This is second line"
6         register: result
7
8       - name: printing variable
9         debug:
10           var: result
```

Ansible Tutorial for Beginners | Ansible Debug Module and Register Variables

[root@ansible ansible]# vi ansibledebug.yml



◀ ▶ ⏪ 🔍 11:35 / 16:29

Scroll for details



Ansible Tutorial for Beginners | Ansible Debug Module and Register Variables



```
[root@ansible1 ansible]# rm -f ansibledebug.yml  
[root@ansible1 ansible]# clear
```

I

Ansible Tutorial for Beginners | Ansible Debug Module and Register Variables

```
[root@ansible1 ansible]# rm -f ansibledebug.yml  
[root@ansible1 ansible]# vi ansibledebug.yml
```



Ansible Tutorial for Beginners | Ansible Debug Module and Register Variables



```
---
- hosts: testservers
  tasks:
    - name: executing sample command
      command: echo -e "This is the first line, \n This is second line"
      register: result

    - name: printing variable
      debug:
        var: result
```

Ansible Tutorial for Beginners | Ansible Debug Module and Register Variables



```
[root@ansible ansible]# rm -f ansibledebug.yml  
[root@ansible ansible]# vi ansibledebug.yml  
[root@ansible ansible]# ansible-playbook ansibledebug.yml
```

I

Ansible Tutorial for Beginners | Ansible Debug Module and Register Variables

```
ok: [192.168.145.146]

TASK [executing sample command] *****
changed: [192.168.145.146]

TASK [printing variable] *****
ok: [192.168.145.146] => {
    "result": {
        "changed": true,
        "cmd": [
            "echo",
            "-e",
            "This is the first line, \\n This is second line"
        ],
        "delta": "0:00:00.101266",
        "end": "2021-06-27 13:25:07.606455",
        "failed": false,
        "rc": 0,
        "start": "2021-06-27 13:25:07.505169",
        "stderr": "",
        "stderr_lines": [],
        "stdout": "This is the first line, \n This is second line",
        "stdout_lines": [
            "This is the first line, ",
            " This is second line"
        ]
    }
}

PLAY RECAP *****
192.168.145.146 : ok=3    changed=1    unreachable=0    failed=0    skipped=0    rescued=0    ignored=0
```

[root@ansibletutorial ansible]# 11:55 / 16:29

Scroll for details

Ansible Tutorial for Beginners | Ansible Debug Module and Register Variables

```
ok: [192.168.145.146]

TASK [executing sample command] ****
changed: [192.168.145.146]

TASK [printing variable] ****
ok: [192.168.145.146] => {
    "result": {
        "changed": true,
        "cmd": [
            "echo",
            "-e",
            "This is the first line, \\n This is second line"
        ],
        "delta": "0:00:00.101266",
        "end": "2021-06-27 13:25:07.606455",
        "failed": false,
        "rc": 0,
        "start": "2021-06-27 13:25:07.505169",
        "stderr": "",
        "stderr_lines": [],
        "stdout": "This is the first line, \n This is second line",
        "stdout_lines": [
            "This is the first line, ",
            " This is second line"
        ]
    }
}

PLAY RECAP ****
192.168.145.146 : ok=3    changed=1    unreachable=0    failed=0    skipped=0    rescued=0    ignored=0
```

[root@ansibile ansible]# 11:58 / 16:29

Scroll for details

Ansible Tutorial for Beginners | Ansible Debug Module and Register Variables



```
ok: [192.168.145.146]
```

```
TASK [executing sample command] ****
```

```
changed: [192.168.145.146]
```

```
TASK [printing variable] ****
```

```
ok: [192.168.145.146] => {
```

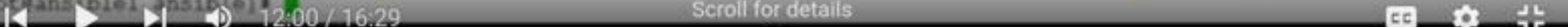
```
  "result": {
    "changed": true,
    "cmd": [
      "echo",
      "-e",
      "This is the first line, \\n This is second line"
    ],
    "delta": "0:00:00.101266",
    "end": "2021-06-27 13:25:07.606455",
    "failed": false,
    "rc": 0,
    "start": "2021-06-27 13:25:07.505189",
    "stderr": "",
    "stderr_lines": [],
    "stdout": "This is the first line, \n This is second line",
    "stdout_lines": [
      "This is the first line, ",
      " This is second line"
    ]
}
```

```
PLAY RECAP ****
```

```
192.168.145.146 : ok=3    changed=1    unreachable=0    failed=0    skipped=0    rescued=0    ignored=0
```

```
[root@ansible1 ansible]#
```

Scroll for details



Ansible Tutorial for Beginners | Ansible Debug Module and Register Variables



```
ok: [192.168.145.146]

TASK [executing sample command] ****
changed: [192.168.145.146]

TASK [printing variable] ****
ok: [192.168.145.146] => {
    "result": [
        "changed": true,
        "cmd": [
            "echo",
            "-e",
            "This is the first line, \\n This is second line"
        ],
        "delta": "0:00:00.101266",
        "end": "2021-06-27 13:25:07.606455",
        "failed": false,
        "rc": 0,
        "start": "2021-06-27 13:25:07.505169",
        "stderr": "",
        "stderr_lines": [],
        "stdout": "This is the first line, \\n This is second line",
        "stdout_lines": [
            "This is the first line, ",
            " This is second line"
        ]
    ]
}
```

```
PLAY RECAP ****
192.168.145.146 : ok=3    changed=1    unreachable=0    failed=0    skipped=0    rescued=0    ignored=0
```

[root@ansible1 ansible]#

Scroll for details



Ansible Tutorial for Beginners | Ansible Debug Module and Register Variables



```
ok: [192.168.145.146]

TASK [executing sample command] ****
changed: [192.168.145.146]

TASK [printing variable] ****
ok: [192.168.145.146] => {
    "result": [
        {
            "changed": true,
            "cmd": [
                "echo",
                "-e",
                "This is the first line, \\n This is second line"
            ],
            "delta": "0:00:00.101266",
            "end": "2021-06-27 13:25:07.606455",
            "failed": false,
            "rc": 0,
            "start": "2021-06-27 13:25:07.505169",
            "stderr": "",
            "stderr_lines": [],
            "stdout": "This is the first line, \\n This is second line",
            "stdout_lines": [
                "This is the first line, ",
                " This is second line"
            ]
        }
    ]
}
```

```
PLAY RECAP ****
192.168.145.146 : ok=3    changed=1    unreachable=0    failed=0    skipped=0    rescued=0    ignored=0
```

[root@ansible1 ansible]#

Scroll for details



Ansible Tutorial for Beginners | Ansible Debug Module and Register Variables



```
ok: [192.168.145.146]

TASK [executing sample command] *****
changed: [192.168.145.146]

TASK [printing variable] *****
ok: [192.168.145.146] => {
    "result": {
        "changed": true,
        "cmd": [
            "echo",
            "-e",
            "This is the first line, \\n This is second line"
        ],
        "delta": "0:00:00.101266",
        "end": "2021-06-27 13:25:07.606455",
        "failed": false,
        "rc": 0,
        "start": "2021-06-27 13:25:07.505169",
        "stderr": "",
        "stderr_lines": [],
        "stdout": "This is the first line, \n This is second line",
        "stdout_lines": [
            "This is the first line, "
            " This is second line"
        ]
    }
}
```

```
PLAY RECAP *****
192.168.145.146 : ok=3    changed=1    unreachable=0    failed=0    skipped=0    rescued=0    ignored=0
```

[root@ansible1 ansible]#

Scroll for details



Ansible Tutorial for Beginners | Ansible Debug Module and Register Variables



```
ok: [192.168.145.146]

TASK [executing sample command] *****
changed: [192.168.145.146]

TASK [printing variable] *****
ok: [192.168.145.146] => {
    "result": {
        "changed": true,
        "cmd": [
            "echo",
            "-e",
            "This is the first line, \\n This is second line"
        ],
        "delta": "0:00:00.101266",
        "end": "2021-06-27 13:25:07.606455",
        "failed": false,
        "rc": 0,
        "start": "2021-06-27 13:25:07.505169",
        "stderr": "",
        "stderr_lines": [],
        "stdout": "This is the first line, \\n This is second line",
        "stdout_lines": [
            "This is the first line, \\n This is second line"
        ]
    }
}
```

```
PLAY RECAP *****
192.168.145.146 : ok=3    changed=1    unreachable=0    failed=0    skipped=0    rescued=0    ignored=0
```

[root@ansible1 ansible]#

Scroll for details



Ansible Tutorial for Beginners | Ansible Debug Module and Register Variables



```
ok: [192.168.145.146]

TASK [executing sample command] ****
changed: [192.168.145.146]

TASK [printing variable] ****
ok: [192.168.145.146] => {
    "result": {
        "changed": true,
        "cmd": [
            "echo",
            "-e",
            "This is the first line, \\n This is second line"
        ],
        "delta": "0:00:00.101266",
        "end": "2021-06-27 13:25:07.606455",
        "failed": false,
        "rc": 0,
        "start": "2021-06-27 13:25:07.505169",
        "stderr": "",
        "stderr_lines": [],
        "stdout": "This is the first line, \\n This is second line",
        "stdout_lines": [
            "This is the first line, ",
            "This is second line"
        ]
    }
}

PLAY RECAP ****
192.168.145.146 : ok=3    changed=1    unreachable=0    failed=0    skipped=0    rescued=0    ignored=0
```

[root@ansible1 ansible]# 12:26 / 16:29

Scroll for details



Ansible Tutorial for Beginners | Ansible Debug Module and Register Variables



```
ok: [192.168.145.146]

TASK [executing sample command] *****
changed: [192.168.145.146]

TASK [printing variable] *****
ok: [192.168.145.146] => {
    "result": {
        "changed": true,
        "cmd": [
            "echo",
            "-e",
            "This is the first line, \n\n This is second line"
        ],
        "delta": "0:00:00.101266",
        "end": "2021-06-27 13:25:07.606455",
        "failed": false,
        "rc": 0,
        "start": "2021-06-27 13:25:07.505189",
        "stderr": "",
        "stderr_lines": [],
        "stdout": "This is the first line, \n\n This is second line",
        "stdout_lines": [
            "This is the first line, \n",
            "This is second line"
        ]
    }
}

PLAY RECAP *****
192.168.145.146 : ok=3    changed=1    unreachable=0    failed=0    skipped=0    rescued=0    ignored=0
```

[root@ansible1 ansible]#

Scroll for details



Ansible Tutorial for Beginners | Ansible Debug Module and Register Variables



```
ok: [192.168.145.146]

TASK [executing sample command] *****
changed: [192.168.145.146]

TASK [printing variable] *****
ok: [192.168.145.146] => {
    "result": {
        "changed": true,
        "cmd": [
            "echo",
            "-e",
            "This is the first line, \n\n This is second line"
        ],
        "delta": "0:00:00.101266",
        "end": "2021-06-27 13:25:07.606455",
        "failed": false,
        "rc": 0,
        "start": "2021-06-27 13:25:07.505189",
        "stderr": "",
        "stderr_lines": [],
        "stdout": "This is the first line, \n\n This is second line",
        "stdout_lines": [
            "This is the first line, ",
            " This is second line"
        ]
    }
}
```

```
PLAY RECAP *****
192.168.145.146 : ok=3    changed=1    unreachable=0    failed=0    skipped=0    rescued=0    ignored=0
```

[root@ansible1 ansible]#

Scroll for details



Ansible Tutorial for Beginners | Ansible Debug Module and Register Variables



```
ok: [192.168.145.146]

TASK [executing sample command] ****
changed: [192.168.145.146]

TASK [printing variable] ****
ok: [192.168.145.146] => {
    "result": [
        "changed": true,
        "cmd": [
            "echo",
            "-e",
            "This is the first line, \\n This is second line"
        ],
        "delta": "0:00:00.101266",
        "end": "2021-06-27 13:25:07.606455",
        "failed": false,
        "rc": 0,
        "start": "2021-06-27 13:25:07.505169",
        "stderr": "",
        "stderr_lines": [],
        "stdout": "This is the first line, \\n This is second line",
        "stdout_lines": [
            "This is the first line, ",
            " This is second line"
        ]
    ]
}

PLAY RECAP ****
192.168.145.146 : ok=3    changed=1    unreachable=0    failed=0    skipped=0    rescued=0    ignored=0
```

[root@ansible1 ansible]# 12:50 / 16:29

Scroll for details



Ansible Tutorial for Beginners | Ansible Debug Module and Register Variables

```
ok: [192.168.145.146]

TASK [executing sample command] *****
changed: [192.168.145.146]

TASK [printing variable] *****
ok: [192.168.145.146] => {
    "result": {
        "changed": true,
        "cmd": [
            "echo",
            "-e",
            "This is the first line, \\n This is second line"
        ],
        "delta": "0:00:00.101266",
        "end": "2021-06-27 13:25:07.606455",
        "failed": false,
        "rc": 0,
        "start": "2021-06-27 13:25:07.505169",
        "stderr": "",
        "stderr_lines": [],
        "stdout": "This is the first line, \\n This is second line",
        "stdout_lines": [
            "This is the first line, *",
            "This is second line"
        ]
    }
}

PLAY RECAP *****
192.168.145.146 : ok=3    changed=1    unreachable=0    failed=0    skipped=0    rescued=0    ignored=0
```

[root@ansible1 ansible]#

Scroll for details



Ansible Tutorial for Beginners | Ansible Debug Module and Register Variables



```
1
2   - hosts: testservers
3     tasks:
4       - name: executing sample command
5         command: echo -e "This is the first line, \n This is second line"
6         register: result
7
8       - name: printing variable
9         debug:
10        var: result
11
12      - name: printing variable with stdout
13        debug:
14          var: result.stdout
15
16      - name: printing variable with stdout.lines
17        debug:
18          msg: "{{result.stdout_lines}}"
```

Ansible Tutorial for Beginners | Ansible Debug Module and Register Variables



```
1 ---
2   - hosts: testservers
3     tasks:
4       - name: executing sample command
5         command: echo -e "This is the first line, \n This is second line"
6         register: result
7
8       - name: printing variable
9         debug:
10            var: result
11
12      - name: printing variable with stdout
13        debug:
14          var: result.stdout
15
16      - name: printing variable with stdout.lines
17        debug:
18          msg: "{{result.stdout_lines}}"
```

Ansible Tutorial for Beginners | Ansible Debug Module and Register Variables



```
1 ...
2 - hosts: testservers
3   tasks:
4     - name: executing sample command
5       command: echo -e "This is the first line, \n This is second line"
6       register: result
7
8     - name: printing variable
9       debug:
10      var: result
11
12    - name: printing variable with stdout
13      debug:
14        var: result.stdout
15
16    - name: printing variable with stdout.lines
17      debug:
18        msg: "{{result.stdout_lines}}"
```

Ansible Tutorial for Beginners | Ansible Debug Module and Register Variables



```
1 ...
2 - hosts: testservers
3   tasks:
4     - name: executing sample command
5       command: echo -e "This is the first line, \n This is second line"
6       register: result
7
8     - name: printing variable
9       debug:
10      var: result
11
12    - name: printing variable with stdout
13      debug:
14        var: result.stdout
15
16    - name: printing variable with stdout.lines
17      debug:
18        msg: "{{result.stdout_lines}}"
```

Ansible Tutorial for Beginners | Ansible Debug Module and Register Variables



```
1
2 - hosts: testservers
3 tasks:
4   - name: executing sample command
5     command: echo -e "This is the first line, \n This is second line"
6     register: result
7
8   - name: printing variable
9     debug:
10    var: result
11
12   - name: printing variable with stdout
13     debug:
14    var: result.stdout
15
16   - name: printing variable with stdout.lines
17     debug:
18     msg: "{{result.stdout_lines}}"
```

Ansible Tutorial for Beginners | Ansible Debug Module and Register Variables



```
1 ...
2 - hosts: testservers
3 tasks:
4 - name: executing sample command
5   command: echo -e "This is the first line, \n This is second line"
6   register: result
7
8 - name: printing variable
9   debug:
10    var: result
11
12 - name: printing variable with stdout
13   debug:
14    var: result.stdout
15
16 - name: printing variable with stdout.lines
17   debug:
18    msg: "{{result.stdout_lines}}"
```

Ansible Tutorial for Beginners | Ansible Debug Module and Register Variables



```
1 ---  
2 hosts: testservers  
3 tasks:  
4   - name: executing sample command  
5     command: echo -e "This is the first line, \n This is second line"  
6     register: result  
7  
8   - name: printing variable  
9     debug:  
10    var: result  
11  
12   - name: printing variable with stdout  
13     debug:  
14     var: result.stdout  
15  
16   - name: printing variable with stdout.lines  
17     debug:  
18     msg: "{{result.stdout_lines}}"
```

Ansible Tutorial for Beginners | Ansible Debug Module and Register Variables

```
[root@ansible1 ansible]# vi ansibledebug.yml
```

I

14:17 / 16:29

Scroll for details

Ansible Tutorial for Beginners | Ansible Debug Module and Register Variables

```
[root@ansible1 ansible]# rm -f ansibledebug.yml
[root@ansible1 ansible]#
```

I

◀ ▶ ⏪ 🔍 14:19 / 16:29

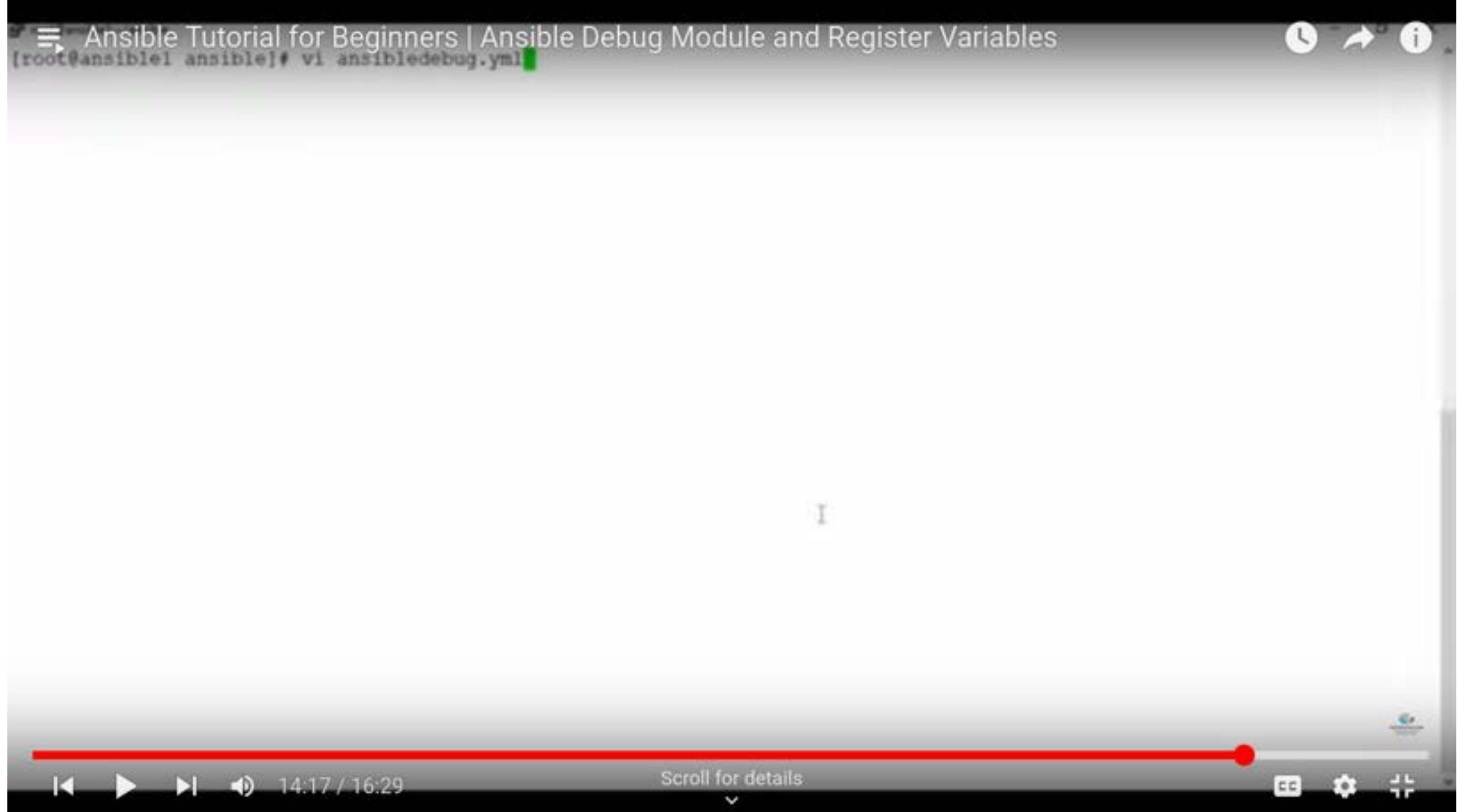
Scroll for details

CC ⚙️ ⌂

Ansible Tutorial for Beginners | Ansible Debug Module and Register Variables

```
[root@ansible1 ansible]# rm -f ansibledebug.yml  
[root@ansible1 ansible]# ansible-playbook ansibledebug.yml
```





Ansible Tutorial for Beginners | Ansible Debug Module and Register Variables

```
[root@ansible1 ansible]# rm -f ansibledebug.yml  
[root@ansible1 ansible]# vi ansibledebug.yml
```



Pull up for previous recording



14:13

◀ ▶ ▶| 🔍 14:20 / 16:29

Scroll for details

CC ⚙️ 🌐

Ansible Tutorial for Beginners | Ansible Debug Module and Register Variables



```
---
```

```
- hosts: testservers
  tasks:
    - name: executing sample command
      command: echo -e "This is the first line, \n This is second line"
      register: result

    - name: printing variable
      debug:
        var: result

    - name: printing variable with stdout
      debug:
        var: result.stdout

    - name: printing variable with stdout.lines
      debug:
        msg: "{{result.stdout_lines}}"
```

Ansible Tutorial for Beginners | Ansible Debug Module and Register Variables



```
[root@ansible ansible]# rm -f ansibledebug.yml  
[root@ansible ansible]# vi ansibledebug.yml  
[root@ansible ansible]# rm -f ansibledebug.yml
```

I

Ansible Tutorial for Beginners | Ansible Debug Module and Register Variables

```
[root@ansible ansible]# rm -f ansibledebug.yml
[root@ansible ansible]# vi ansibledebug.yml
[root@ansible ansible]# ansible-playbook ansibledebug.yml
```

I

The image shows a video player interface. At the bottom left, there are playback controls: a left arrow, a right arrow, a double-left arrow, a double-right arrow, and a volume icon. The time is displayed as 14:29 / 16:29. In the center, there is a red horizontal progress bar with a black slider. Below the progress bar, the text "Scroll for details" is followed by a downward-pointing arrow. On the far right, there are icons for closed captions, settings, and other video controls.

Ansible Tutorial for Beginners | Ansible Debug Module and Register Variables



```
    "stdout": "This is the first line, \n This is second line",
    "stdout_lines": [
        "This is the first line, ",
        " This is second line"
    ]
}

PLAY RECAP ****
192.168.145.146      : ok=3    changed=1    unreachable=0    failed=0    skipped=0    rescued=0    ignored=0

[root@ansible1 ansible]# clear
[root@ansible1 ansible]# rm -f ansibledebug.yml
[root@ansible1 ansible]# vi ansibledebug.yml
[root@ansible1 ansible]# ansible-playbook ansibledebug.yml

PLAY [testservers] ****

TASK [Gathering Facts] ****
ok: [192.168.145.146]

TASK [executing sample command] ****
changed: [192.168.145.146]

TASK [printing variable] ****
ok: [192.168.145.146] => {
    "result": {
        "changed": true,
        "cmd": [
            "echo",
            "-e",
            "This is the first line, \n This is second line"
        ],
        "delta": "0:00:00.000000",
        "end": "2018-01-10T14:34:29.000000Z",
        "rc": 0
    }
}
```

Ansible Tutorial for Beginners | Ansible Debug Module and Register Variables



PLAY [testservers] ****

TASK [Gathering Facts] ****

ok: [192.168.145.146]

TASK [executing sample command] ****

changed: [192.168.145.146]

TASK [printing variable] ****

ok: [192.168.145.146] => {

 "result": {

 "changed": true,

 "cmd": {

 "echo",

 "-e",

 "This is the first line, \n\n This is second line"

 },

 "delta": "0:00:00.004408",

 "end": "2021-06-27 13:27:56.463851",

 "failed": false,

 "rc": 0,

 "start": "2021-06-27 13:27:56.459443",

 "stderr": "",

 "stderr_lines": [],

 "stdout": "This is the first line, \n\n This is second line",

 "stdout_lines": [

 "This is the first line, \n",

 " This is second line"

TASK [printing variable with stdout] ****

Scroll for details

◀ ▶ ⏪ ⏩ 🔍 14:37 / 16:29

CC 🛡️ 🌐

Ansible Tutorial for Beginners | Ansible Debug Module and Register Variables



```
},  
  "delta": "0:00:00.004406",  
  "end": "2021-06-27 13:27:56.463851",  
  "failed": false,  
  "rc": 0,  
  "start": "2021-06-27 13:27:56.459443",  
  "stderr": "",  
  "stderr_lines": [],  
  "stdout": "This is the first line, \n This is second line",  
  "stdout_lines": [  
    "This is the first line, ",  
    " This is second line"  
  ]  
}  
  
TASK [printing variable with stdout] ****  
ok: [192.168.145.146] => {  
  "result.stdout": "This is the first line, \n This is second line"  
}  
  
TASK [printing variable with stdout.lines] ****  
ok: [192.168.145.146] => {  
  "msg": [  
    "This is the first line, ",  
    " This is second line"  
]  
}  
  
PLAY RECAP ****  
192.168.145.146 : ok=5    changed=1    unreachable=0    failed=0    skipped=0    rescued=0    ignored=0  
[root@ansible1 ansible]# 14:51 / 16:29
```

Scroll for details



Ansible Tutorial for Beginners | Ansible Debug Module and Register Variables



```
  },
  "delta": "0:00:00.004408",
  "end": "2021-06-27 13:27:56.463851",
  "failed": false,
  "rc": 0,
  "start": "2021-06-27 13:27:56.459443",
  "stderr": "",
  "stderr_lines": [],
  "stdout": "This is the first line, \n This is second line",
  "stdout_lines": [
    "This is the first line, ",
    " This is second line"
  ]
}

TASK [printing variable with stdout] ****
ok: [192.168.145.146] => {
  "result.stdout": "This is the first line, \n This is second line"
}

TASK [printing variable with stdout.lines] ****
ok: [192.168.145.146] => {
  "msg": [
    "This is the first line, ",
    " This is second line"
  ]
}

PLAY RECAP ****
192.168.145.146 : ok=5    changed=1    unreachable=0    failed=0    skipped=0    rescued=0    ignored=0
```

[root@ansible1 ansible]# 14:54 / 16:29

Scroll for details



Ansible Tutorial for Beginners | Ansible Debug Module and Register Variables



```
},  
  "delta": "0:00:00.004408",  
  "end": "2021-06-27 13:27:56.463851",  
  "failed": false,  
  "rc": 0,  
  "start": "2021-06-27 13:27:56.459443",  
  "stderr": "",  
  "stderr_lines": [],  
  "stdout": "This is the first line, \n This is second line",  
  "stdout_lines": [  
    "This is the first line, ",  
    " This is second line"  
  ]  
}  
  
TASK [printing variable with stdout] ****  
ok: [192.168.145.146] => {  
  "result.stdout": "This is the first line, \n This is second line"  
}  
  
TASK [printing variable with stdout.lines] ****  
ok: [192.168.145.146] => {  
  "msg": [  
    "This is the first line, ",  
    " This is second line"  
]  
}  
  
PLAY RECAP ****  
192.168.145.146 : ok=5    changed=1    unreachable=0    failed=0    skipped=0    rescued=0    ignored=0  
[root@ansible1 ansible]# 14:58 / 16:29
```

Scroll for details



Ansible Tutorial for Beginners | Ansible Debug Module and Register Variables



```
],
  "delta": "0:00:00.004408",
  "end": "2021-06-27 13:27:56.463851",
  "failed": false,
  "rc": 0,
  "start": "2021-06-27 13:27:56.459443",
  "stderr": "",
  "stderr_lines": [],
  "stdout": "This is the first line, \n This is second line",
  "stdout_lines": [
    "This is the first line, ",
    " This is second line"
  ]
}

TASK [printing variable with stdout] ****
ok: [192.168.145.146] => {
  "result.stdout": "This is the first line, \n This is second line"
}

TASK [printing variable with stdout.lines] ****
ok: [192.168.145.146] => {
  "msg": [
    "This is the first line, ",
    " This is second line"
  ]
}

PLAY RECAP ****
192.168.145.146 : ok=5    changed=1    unreachable=0    failed=0    skipped=0    rescued=0    ignored=0
```

[root@ansible1 ansible]# 15:02 / 16:29

Scroll for details



Ansible Tutorial for Beginners | Ansible Debug Module and Register Variables



```
},
  "delta": "0:00:00.004408",
  "end": "2021-06-27 13:27:56.463851",
  "failed": false,
  "rc": 0,
  "start": "2021-06-27 13:27:56.459443",
  "stderr": "",
  "stderr_lines": [],
  "stdout": "This is the first line, \n This is second line",
  "stdout_lines": [
    "This is the first line, ",
    " This is second line"
  ]
}

TASK [printing variable with stdout] ****
ok: [192.168.145.146] => {
  "result.stdout": "This is the first line, \n This is second line"
}

TASK [printing variable with stdout.lines] ****
ok: [192.168.145.146] => {
  "msg": [
    "This is the first line, ",
    " This is second line"
  ]
}

PLAY RECAP ****
192.168.145.146 : ok=5    changed=1    unreachable=0    failed=0    skipped=0    rescued=0    ignored=0
```

[root@ansibile ~]

Scroll for details



Ansible Tutorial For Beginners | When Conditional | Ansible Playbook Tutorial for Beginners



```
! Conditional.yaml •  
C:\Users\k\Desktop\! Conditional.yaml  
1  
2 ---  
3 - hosts: testservers  
4   become: yes  
5   tasks:  
6     - name: Installing Apache on CentOS  
7       shell: name=httpd state=present  
8       when: ansible_os_family == "RedHat"  
9  
10    - name: Install Apache on Ubuntu Server  
11      apt: name=apache2 state=present  
12      when: ansible_os_family == "Debian"  
13
```



◀ ▶ ▷ 🔍 0:08 / 18:33

Scroll for details



AhumbleTutorial | Ansible Tutorial For Beginners | When Conditional | Ansible Playbook Tutorial for Beginners

```
[root@ansible1 ansible]# rm -rf ansibleconditional.yaml
[root@ansible1 ansible]# vi ansibleconditional.yaml
[root@ansible1 ansible]# ansible-playbook ansibleconditional.yaml

PLAY [testservers] ****
TASK [Gathering Facts] ****
ok: [192.168.145.146]

TASK [Installing Apache on CentOS] ****
changed: [192.168.145.146]
I
TASK [Install Apache on Ubuntu Server] ****
skipping: [192.168.145.146]

PLAY RECAP ****
192.168.145.146 : ok=2    changed=1    unreachable=0    failed=0    skipped=1    rescued=0    ignored=0

[root@ansible1 ansible]#
```

0:09 / 18:33

Scroll for details

Ansible Tutorial for Beginners

Ansible Playbook Tutorial- Conditionals

Using ‘when’ conditional statement

- In a playbook, you may want to execute different tasks based on value from facts, variable or output of other tasks
- Why we are not using If else instead using when statement

If else vs when Conditions

If else vs when Conditions

If conditional

If else vs when Conditions

If the system don't have Apache installed

If conditional



Do action: Steps to install Apache

If else vs when Conditions

If the system don't have Apache installed

If conditional



When conditional

Do action: Steps to install Apache

If else vs when Conditions

If the system don't have Apache installed

If conditional



Do action: Steps to install Apache

Define the action: Steps to install Apache

When conditional



Do it when there is no Apache is installed
on the system.

Ansible Playbook Tutorial For Beginners



Red Hat

Ansible Playbook Tutorial For Beginners



Red Hat

! Conditional.yaml ×

C: > Users > k > Desktop > ! Conditional.yaml

```
1
2  ---
3  - hosts: testservers
4    tasks:
5      - name: executing sample command
6        shell: uptime
7        register: result
8
9      - name: printing variable
10        debug:
11          msg: "{{result}}"
```

Ansible Tutorial For Beginners | When Conditional | Ansible Playbook Tutorial for Beginners



! Conditional.yaml x

C:\Users\k\Desktop\! Conditional.yaml

```
1
2
3 ---  
4   - hosts: testservers  
5     tasks:  
6       - name: executing sample command  
7         shell: uptime  
8         register: result  
9  
10      - name: printing variable  
11        debug:  
12          msg: "{{result}}"
```

[root@ansible ansible]# vi ansibleconditional.yaml



I

Ahsible Tutorial For Beginners | When Conditional | Ansible Playbook Tutorial for Beginners

```
---  
- hosts: testservers  
  tasks:  
    - name: executing sample command  
      shell: uptime  
      register: result  
  
    - name: printing variable  
      debug:  
        msg: "{{result}}"
```

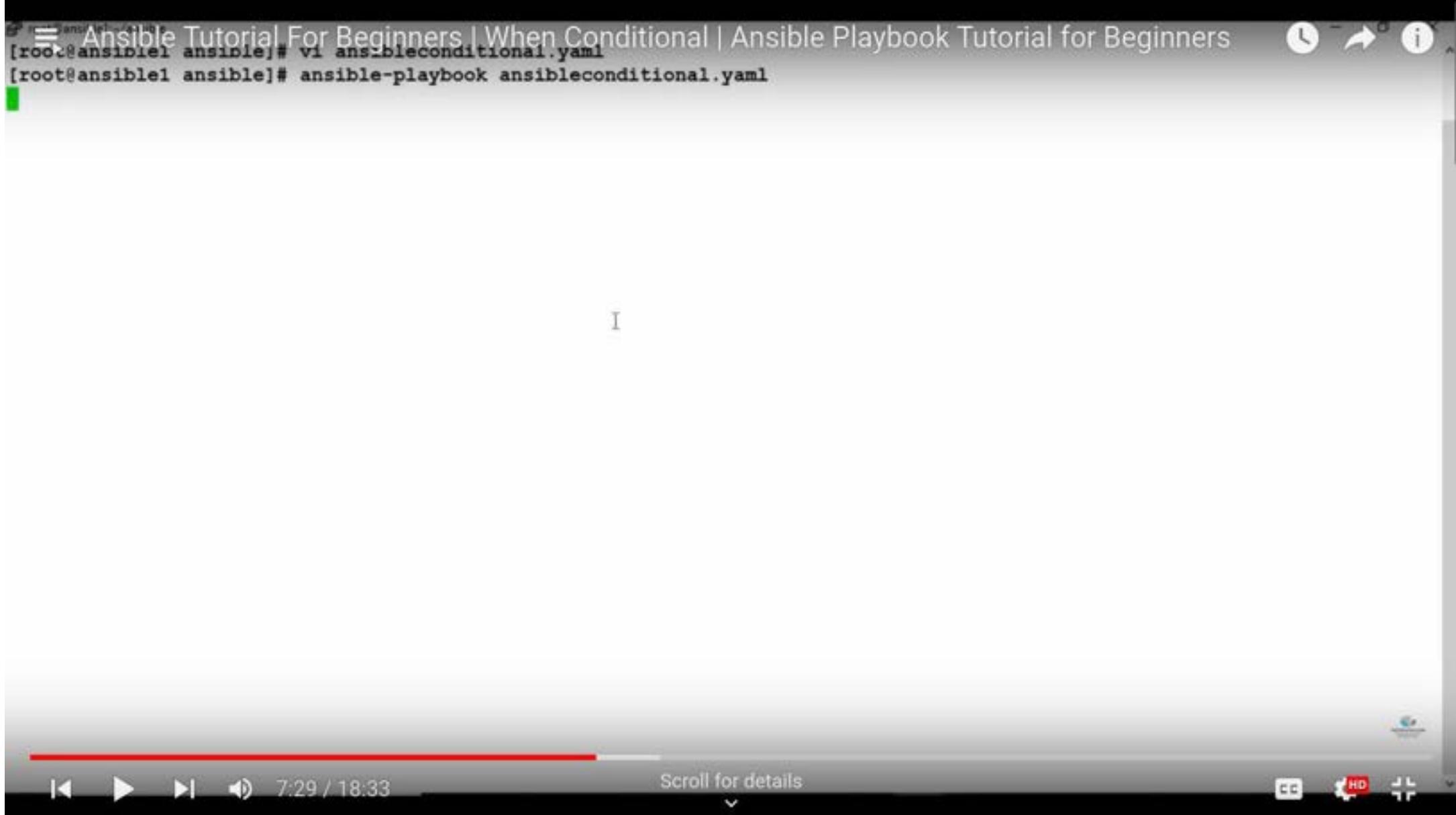
I

```
Ansible Tutorial For Beginners | When Conditional | Ansible Playbook Tutorial for Beginners
PLAY [testservers] ****
TASK [Gathering Facts] ****
ok: [192.168.145.146]

TASK [executing sample command] ****
changed: [192.168.145.146]

TASK [printing variable] ****
ok: [192.168.145.146] => {
    "msg": {
        "changed": true,
        "cmd": "uptime",
        "delta": "0:00:00.023804",
        "end": "2021-07-25 13:50:56.967444",
        "failed": false,
        "rc": 0,
        "start": "2021-07-25 13:50:56.943640",
        "stderr": "",
        "stderr_lines": [],
        "stdout": " 13:50:56 up 1:19, 3 users, load average: 0.00, 0.01, 0.05",
        "stdout_lines": [
            " 13:50:56 up 1:19, 3 users, load average: 0.00, 0.01, 0.05"
        ]
    }
}

PLAY RECAP ****
192.168.145.146 : ok=3    changed=1    unreachable=0    failed=0    skipped=0    rescued=0    ignored=0
```



```
Ansible Tutorial For Beginners | When Conditional | Ansible Playbook Tutorial for Beginners
PLAY [testservers] ****
TASK [Gathering Facts] ****
ok: [192.168.145.146]

TASK [executing sample command] ****
changed: [192.168.145.146]

TASK [printing variable] ****
ok: [192.168.145.146] => {
  "msg": {
    "changed": true,
    "cmd": "uptime",
    "delta": "0:00:00.023804",
    "end": "2021-07-25 13:50:56.967444",
    "failed": false,
    "rc": 0,
    "start": "2021-07-25 13:50:56.943640",
    "stderr": "",
    "stderr_lines": [],
    "stdout": " 13:50:56 up 1:19, 3 users, load average: 0.00, 0.01, 0.05",
    "stdout_lines": [
      " 13:50:56 up 1:19, 3 users, load average: 0.00, 0.01, 0.05"
    ]
  }
}

PLAY RECAP ****
192.168.145.146 : ok=3    changed=1    unreachable=0    failed=0    skipped=0    rescued=0    ignored=0
```

Ansible Tutorial For Beginners | When Conditional | Ansible Playbook Tutorial for Beginners

! Conditional.yaml ●

C: > Users > k > Desktop > ! Conditional.yaml

```
1
2  ---
3  - hosts: testservers
4    tasks:
5      - name: executing sample command
6        shell: uptime
7        register: result
8
9      - name: printing variable
10        debug:
11          msg: "{{result}}"
12          when: ansible_os_family == "RedHat"
```

Scroll for details

Ansible Tutorial For Beginners | When Conditional | Ansible Playbook Tutorial for Beginners



```
! Conditional.yaml ●  
C: > Users > k > Desktop > ! Conditional.yaml  
1  
2 ---  
3 - hosts: testservers  
4 tasks:  
5   - name: executing sample command  
6     shell: uptime  
7     register: result  
8  
9   - name: printing variable  
10    debug:  
11      msg: "{{result}}"  
12      when: ansible_os_family == "RedHat"
```



◀ ▶ ▷ 🔍 8:14 / 18:33

Scroll for details



Ansible Tutorial For Beginners | When Conditional | Ansible Playbook Tutorial for Beginners

! Conditional.yaml

C:\> Users > k > Desktop > ! Conditional.yaml

```
1
2  ---
3  - hosts: testservers
4    tasks:
5      - name: executing sample command
6        shell: uptime
7        register: result
8
9      - name: printing variable
10        debug:
11          msg: "{{result}}"
12        when: ansible_os_family == "RedHat"
```

Scroll for details

Ansible Tutorial For Beginners | When Conditional | Ansible Playbook Tutorial for Beginners

! Conditional.yaml

C:\> Users > k > Desktop > ! Conditional.yaml

```
1
2  ---
3  - hosts: testservers
4    tasks:
5      - name: executing sample command
6        shell: uptime
7        register: result
8
9      - name: printing variable
10        debug:
11          msg: "{{result}}"
12        when: ansible_os_family == "RedHat"
```

Scroll for details

Ahsible Tutorial For Beginners | When Conditional | Ansible Playbook Tutorial for Beginners

```
PLAY [testservers] ****
TASK [Gathering Facts] ****
ok: [192.168.145.146]

TASK [executing sample command] ****
changed: [192.168.145.146]

TASK [printing variable] ****
ok: [192.168.145.146] => {
  "msg": {
    "changed": true,
    "cmd": "uptime",
    "delta": "0:00:00.023804",
    "end": "2021-07-25 13:50:56.967444",
    "failed": false,
    "rc": 0,
    "start": "2021-07-25 13:50:56.943640",
    "stderr": "",
    "stderr_lines": [],
    "stdout": " 13:50:56 up 1:19,  3 users,  load average: 0.00, 0.01, 0.05",
    "stdout_lines": [
      " 13:50:56 up 1:19,  3 users,  load average: 0.00, 0.01, 0.05"
    ]
  }
}

PLAY RECAP ****
192.168.145.146 : ok=3    changed=1    unreachable=0    failed=0    skipped=0    rescued=0    ignored=0
```

[root@ansible1 ansible]# 8.28 / 18:33

Scroll for details

[root@ansible ansible]# ansible testservers -m setup



I

```
Ansible Tutorial For Beginners | When Conditional | Ansible Playbook Tutorial for Beginners
```

```
[root@ansibile ansible]#
```

```
    "cap_lease",
    "cap_audit_write",
    "cap_audit_control",
    "cap_setfcap",
    "cap_mac_override",
    "cap_mac_admin",
    "cap_syslog",
    "35",
    "36+ep"
],
"ansible_system_capabilities_enforced": "True",
"ansible_system_vendor": "VMware, Inc.",
"ansible_uptime_seconds": 4844,
"ansible_user_dir": "/root",
"ansible_user_gecos": "root",
"ansible_user_gid": 0,
"ansible_user_id": "root",
"ansible_user_shell": "/bin/bash",
"ansible_user_uid": 0,
"ansible_userspace_architecture": "x86_64",
"ansible_userspace_bits": "64",
"ansible_virtualization_role": "guest",
"ansible_virtualization_type": "VMware",
"discovered_interpreter_python": "/usr/bin/python",
"gather_subset": [
    "all"
],
"module_setup": true
},
"changed": false
}
```

8.41 / 18:33

Scroll for details

Ahsible Tutorial For Beginners | When Conditional | Ansible Playbook Tutorial for Beginners

```
        "block_used": 1058268,
        "device": "/dev/mapper/centos-root",
        "fstype": "xfs",
        "inode_available": 8880029,
        "inode_total": 8910848,
        "inode_used": 30819,
        "mount": "/",
        "options": "rw,seclabel,relatime,attr2,inode64,noquota",
        "size_available": 13986185216,
        "size_total": 18238930944,
        "uuid": "29afdd6-e829-4d3b-a377-2f4901232fdf"
    },
],
"ansible_nodename": "localhost.localdomain",
"ansible_os_family": "RedHat",
"ansible_pkg_mgr": "yum",
"ansible_proc_cmdline": {
    "BOOT_IMAGE": "/vmlinuz-3.10.0-957.el7.x86_64",
    "LANG": "en_US.UTF-8",
    "crashkernel": "auto",
    "quiet": true,
    "rd.lvm.lv": [
        "centos/root",
        "centos/swap"
    ],
    "rhgb": true,
    "ro": true,
    "root": "/dev/mapper/centos-root"
},
"ansible_processor": [
    "0"
]
```

Ansible Tutorial For Beginners | When Conditional | Ansible Playbook Tutorial for Beginners

```
        "block_used": 1058268,
        "device": "/dev/mapper/centos-root",
        "fstype": "xfs",
        "inode_available": 8880029,
        "inode_total": 8910848,
        "inode_used": 30819,
        "mount": "/",
        "options": "rw,seclabel,relatime,attr2,inode64,noquota",
        "size_available": 13986185216,
        "size_total": 18238930944,
        "uuid": "29afddb6-e829-4d3b-a377-2f4901232fdf"
    },
],
"ansible_nodename": "localhost.localdomain",
"ansible_os_family": "RedHat",
"ansible_pkg_mgr": "yum",
"ansible_proc_cmdline": {
    "BOOT_IMAGE": "/vmlinuz-3.10.0-957.el7.x86_64",
    "LANG": "en_US.UTF-8",
    "crashkernel": "auto",
    "quiet": true,
    "rd.lvm.lv": [
        "centos/root",
        "centos/swap"
    ],
    "rhgb": true,
    "ro": true,
    "root": "/dev/mapper/centos-root"
},
"ansible_processor": [
    "0"
]
```

Ansible Tutorial For Beginners | When Conditional | Ansible Playbook Tutorial for Beginners

! Conditional.yaml

C > Users > k > Desktop > ! Conditional.yaml

```
1
2  ---
3  - hosts: testservers
4    tasks:
5      - name: executing sample command
6        shell: uptime
7        register: result
8
9      - name: printing variable
10        debug:
11          msg: "{{result}}"
12        when: ansible_os_family == "RedHat"
```

Scroll for details

Ansible Tutorial For Beginners | When Conditional | Ansible Playbook Tutorial for Beginners

! Conditional.yaml

C:\> Users > k > Desktop > ! Conditional.yaml

```
1
2  ---
3  - hosts: testservers
4    tasks:
5      - name: executing sample command
6        shell: uptime
7        register: result
8
9      - name: printing variable
10     debug:
11       msg: "{{result}}"
12     when: ansible_os_family == "RedHat"
```

Scroll for details

Ahsible Tutorial For Beginners | When Conditional | Ansible Playbook Tutorial for Beginners



```
---  
- hosts: testservers  
  tasks:  
    - name: executing sample command  
      shell: uptime  
      register: result  
  
    - name: printing variable  
      debug:  
        msg: "{{result}}"  
      when: ansible_os_family == "RedHat"
```

[root@ansible1 ansible]# rm -rf ansibleconditional.yaml
[root@ansible1 ansible]# vi ansibleconditional.yaml
[root@ansible1 ansible]# ansible-playbook ansibleconditional.yaml

Ahsible Tutorial For Beginners | When Conditional | Ansible Playbook Tutorial for Beginners

```
PLAY [testservers] ****
TASK [Gathering Facts] ****
ok: [192.168.145.146]

TASK [executing sample command] ****
changed: [192.168.145.146]

TASK [printing variable] ****
ok: [192.168.145.146] => {
  "msg": {
    "changed": true,
    "cmd": "uptime",
    "delta": "0:00:00.007067",
    "end": "2021-07-25 13:53:37.465323",
    "failed": false,
    "rc": 0,
    "start": "2021-07-25 13:53:37.458256",
    "stderr": "",
    "stderr_lines": [],
    "stdout": " 13:53:37 up 1:21, 3 users, load average: 0.01, 0.02, 0.05",
    "stdout_lines": [
      " 13:53:37 up 1:21, 3 users, load average: 0.01, 0.02, 0.05"
    ]
  }
}

PLAY RECAP ****
192.168.145.146 : ok=3    changed=1    unreachable=0    failed=0    skipped=0    rescued=0    ignored=0
```

[root@ansible ansible]# 9:33 / 18:33

Scroll for details

Ahsible Tutorial For Beginners | When Conditional | Ansible Playbook Tutorial for Beginners

```
---  
- hosts: testservers  
  tasks:  
    - name: executing sample command  
      shell: uptime  
      register: result  
  
    - name: printing variable  
      debug:  
        msg: "{{result}}"  
      when: ansible_os_family == "RedHat"
```

"ansibleconditional.yaml" 13F 25.4C

Scroll for details

Ahsible Tutorial For Beginners | When Conditional | Ansible Playbook Tutorial for Beginners



```
---  
- hosts: testservers  
  tasks:  
    - name: executing sample command  
      shell: uptime  
      register: result  
  
    - name: printing variable  
      debug:  
        msg: "{{result}}"  
      when: ansible_os_family == "RedHat"
```

Ahsible Tutorial For Beginners | When Conditional | Ansible Playbook Tutorial for Beginners

```
---  
- hosts: testservers  
  tasks:  
    - name: executing sample command  
      shell: uptime  
      register: result  
  
    - name: printing variable  
      debug:  
        msg: "{{result}}"  
      when: ansible_os_family == "Debian"
```

9:57 / 18:33

Scroll for details

```
Ansible Tutorial For Beginners | When Conditional | Ansible Playbook Tutorial for Beginners... ⓘ
TASK [executing sample command] ****
changed: [192.168.145.146]

TASK [printing variable] ****
ok: [192.168.145.146] => {
    "msg": {
        "changed": true,
        "cmd": "uptime",
        "delta": "0:00:00.007067",
        "end": "2021-07-25 13:53:37.465323",
        "failed": false,
        "rc": 0,
        "start": "2021-07-25 13:53:37.458256",
        "stderr": "",
        "stderr_lines": [],
        "stdout": " 13:53:37 up 1:21, 3 users, load average: 0.01, 0.02, 0.05",
        "stdout_lines": [
            " 13:53:37 up 1:21, 3 users, load average: 0.01, 0.02, 0.05"
        ]
    }
}

PLAY RECAP ****
192.168.145.146 : ok=3    changed=1    unreachable=0    failed=0    skipped=0    rescued=0    ignored=0

[root@ansible1 ansible]# vi ansibleconditional.yaml
[root@ansible1 ansible]# ansible-playbook ansibleconditional.yaml

PLAY [testservers] ****
TASK [Gathering Facts] ****

```

10:03 / 18:33

Scroll for details

```
Ansible Tutorial For Beginners | When Conditional | Ansible Playbook Tutorial for Beginners
```

```
    "rc": 0,
    "start": "2021-07-25 13:53:37.458256",
    "stderr": "",
    "stderr_lines": [],
    "stdout": " 13:53:37 up 1:21, 3 users, load average: 0.01, 0.02, 0.05",
    "stdout_lines": [
        " 13:53:37 up 1:21, 3 users, load average: 0.01, 0.02, 0.05"
    ]
}

PLAY RECAP ****
192.168.145.146 : ok=3    changed=1    unreachable=0    failed=0    skipped=0    rescued=0    ignored=0

[root@ansible1 ansible]# vi ansibleconditional.yaml
[root@ansible1 ansible]# ansible-playbook ansibleconditional.yaml

PLAY [testservers] **

TASK [Gathering Facts] ****
ok: [192.168.145.146]

TASK [executing sample command] ****
changed: [192.168.145.146]

TASK [printing variable] ****
skipping: [192.168.145.146]

PLAY RECAP ****
192.168.145.146 : ok=2    changed=1    unreachable=0    failed=0    skipped=1    rescued=0    ignored=0
```

[root@ansible1 ansible]# 10:06 / 18:33

Scroll for details

```
Ansible Tutorial For Beginners | When Conditional | Ansible Playbook Tutorial for Beginners
```

```
        "rc": 0,
        "start": "2021-07-25 13:53:37.458256",
        "stderr": "",
        "stderr_lines": [],
        "stdout": " 13:53:37 up 1:21, 3 users, load average: 0.01, 0.02, 0.05",
        "stdout_lines": [
            " 13:53:37 up 1:21, 3 users, load average: 0.01, 0.02, 0.05"
        ]
    }

PLAY RECAP ****
192.168.145.146 : ok=3    changed=1    unreachable=0    failed=0    skipped=0    rescued=0    ignored=0

[root@ansible1 ansible]# vi ansibleconditional.yaml
[root@ansible1 ansible]# ansible-playbook ansibleconditional.yaml

PLAY [testservers] **

TASK [Gathering Facts] ****
ok: [192.168.145.146]

TASK [executing sample command] ****
changed: [192.168.145.146]

TASK [printing variable] ****
skipping: [192.168.145.146]

PLAY RECAP ****
192.168.145.146 : ok=2    changed=1    unreachable=0    failed=0    skipped=1    rescued=0    ignored=0
```

[root@ansible1 ansible]# 10:15 / 18:33

Scroll for details

Ansible Tutorial For Beginners | When Conditional | Ansible Playbook Tutorial for Beginners

! Conditional.yaml

C > Users > k > Desktop > ! Conditional.yaml

```
1
2  ---
3  - hosts: testservers
4    become: yes
5    tasks:
6      - name: Installing Apache on CentOS
7        shell: name=httpd state=present
8
9
10     - name: Install Apache on Ubuntu Server
11       apt: name=apache2 state=present
12   |
```

Scroll for details

Ansible Tutorial For Beginners | When Conditional | Ansible Playbook Tutorial for Beginners

! Conditional.yaml ●

C: > Users > k > Desktop > ! Conditional.yaml

```
1
2  ---
3  - hosts: testservers
4    become: yes
5    tasks:
6      - name: Installing Apache on CentOS
7        shell: name=httpd state=present
8
9
10     - name: Install Apache on Ubuntu Server
11       apt: name=apache2 state=present
12
```

Scroll for details

Ansible Tutorial For Beginners | When Conditional | Ansible Playbook Tutorial for Beginners



! Conditional.yaml •

C: > Users > k > Desktop > ! Conditional.yaml

```
1
2  ---
3  - hosts: testservers
4    become: yes
5    tasks:
6      - name: Installing Apache on CentOS
7        yum: name=httpd state=present
8
9
10     - name: Install Apache on Ubuntu Server
11       apt: name=apache2 state=present
12
```



◀ ▶ ▶| 🔍 11:14 / 18:33

Scroll for details



Ansible Tutorial For Beginners | When Conditional | Ansible Playbook Tutorial for Beginners



```
! Conditional.yaml •  
C: > Users > k > Desktop > ! Conditional.yaml  
1  
2 ---  
3 - hosts: testservers  
4   become: yes  
5   tasks:  
6     - name: Installing Apache on CentOS  
7       shell: name=httpd state=present  
8       when: ansible_os_family == "RedHat"  
9  
10    - name: Install Apache on Ubuntu Server  
11      apt: name=apache2 state=present  
12      when: ansible_os_family == "Debian"  
13
```



◀ ▶ ▷ 🔍 12:36 / 18:33

Scroll for details



Ansible Tutorial For Beginners | When Conditional | Ansible Playbook Tutorial for Beginners



! Conditional.yaml ●



C: > Users > k > Desktop > ! Conditional.yaml



```
1
2  ---
3  - hosts: testservers
4    become: yes
5    tasks:
6      - name: Installing Apache on CentOS
7        shell: name=httpd state=present
8        when: ansible_os_family == "RedHat"
9
10     - name: Install Apache on Ubuntu Server
11       apt: name=apache2 state=present
12       when: ansible_os_family == "Debian"
13
```



◀ ▶ ▶| 🔍 12:45 / 18:33

Scroll for details



Ansible Tutorial For Beginners | When Conditional | Ansible Playbook Tutorial for Beginners



! Conditional.yaml



C:\> Users > k > Desktop > ! Conditional.yaml



```
1
2
3 - hosts: testservers
4   become: yes
5   tasks:
6     - name: Installing Apache on CentOS
7       shell: name=httpd state=present
8       when: ansible_os_family == "RedHat"
9
10    - name: Install Apache on Ubuntu Server
11      apt: name=apache2 state=present
12      when: ansible_os_family == "Debian"
13
```



◀ ▶ ▶| ⟲ 12:50 / 18:33

Scroll for details



Ahsible Tutorial For Beginners | When Conditional | Ansible Playbook Tutorial for Beginners



```
---  
- hosts: testservers  
become: yes  
tasks:  
- name: Installing Apache on Centos  
shell: name=httpd state=present  
when: ansible_os_family == "RedHat"  
  
- name: Install Apache on Ubuntu Server  
apt: name=apache2 state=present  
when: ansible_os_family == "Debian"
```

I

Ansible Tutorial For Beginners | When Conditional | Ansible Playbook Tutorial for Beginners

```
[root@ansible1 ansible]# rm -rf ansibleconditional.yaml
[root@ansible1 ansible]# vi ansibleconditional.yaml
[root@ansible1 ansible]# ansible-playbook ansibleconditional.yaml
```

Ansible Tutorial For Beginners | When Conditional | Ansible Playbook Tutorial for Beginners

```
[root@ansible1 ansible]# rm -rf ansibleconditional.yaml
[root@ansible1 ansible]# vi ansibleconditional.yaml
[root@ansible1 ansible]# ansible-playbook ansibleconditional.yaml

PLAY [testservers] ****
TASK [Gathering Facts] ****
ok: [192.168.145.146]

TASK [Installing Apache on CentOS] ****
changed: [192.168.145.146]

TASK [Install Apache on Ubuntu Server] ****
skipping: [192.168.145.146]

PLAY RECAP ****
192.168.145.146 : ok=2    changed=1    unreachable=0    failed=0    skipped=1    rescued=0    ignored=0

[root@ansible1 ansible]#
```

Ansible Tutorial For Beginners | When Conditional | Ansible Playbook Tutorial for Beginners

```
[root@ansible1 ansible]# rm -rf ansibleconditional.yaml
[root@ansible1 ansible]# vi ansibleconditional.yaml
[root@ansible1 ansible]# ansible-playbook ansibleconditional.yaml

PLAY [testservers] ****
TASK [Gathering Facts] ****
ok: [192.168.145.146]

TASK [Installing Apache on CentOS] ****
changed: [192.168.145.146]

TASK [Install Apache on Ubuntu Server] ****
skipping: [192.168.145.146]

PLAY RECAP ****
192.168.145.146      : ok=2    changed=1    unreachable=0    failed=0    skipped=1    rescued=0    ignored=0

[root@ansible1 ansible]#
```

Ansible Tutorial For Beginners | When Conditional | Ansible Playbook Tutorial for Beginners



! Conditional.yaml •

C:\> Users > k > Desktop > ! Conditional.yaml

```
1  ---
2  - hosts: testservers
3    tasks:
4      - name: Run with items greater than 5
5        command: echo {{ item }}
6        loop: [ 0, 2, 4, 6, 8, 10 ]
7        when: item > 5
```



◀ ▶ ▶| 🔍 13:50 / 18:33

Scroll for details

CC HD

Ansible Tutorial For Beginners | When Conditional | Ansible Playbook Tutorial for Beginners

! Conditional.yaml

C:\> Users > k > Desktop > ! Conditional.yaml

```
1 ---  
2 - hosts: testservers  
3   tasks:  
4     - name: Run with items greater than 5  
5       command: echo {{ item }}  
6       loop: [ 0, 2, 4, 6, 8, 10 ]  
7       when: item > 5
```

Scroll for details

[root@ansible1 ansible]# rm -rf ansibleconditional.yaml
[root@ansible1 ansible]# vi ansibleconditional.yaml
[root@ansible1 ansible]# ansible-playbook ansibleconditional.yaml

```
PLAY [testservers] ****
TASK [Gathering Facts] ****
ok: [192.168.145.146]

TASK [Installing Apache on CentOS] ****
changed: [192.168.145.146]

TASK [Install Apache on Ubuntu Server] ****
skipping: [192.168.145.146]

PLAY RECAP ****
192.168.145.146 : ok=2    changed=1    unreachable=0    failed=0    skipped=1    rescued=0    ignored=0
```

[root@ansible1 ansible]#

Ansible Tutorial For Beginners | When Conditional | Ansible Playbook Tutorial for Beginners

```
[root@ansible1 ansible]# rm -rf ansibleconditional.yaml
[root@ansible1 ansible]# vi ansibleconditional.yaml
[root@ansible1 ansible]# ansible-playbook ansibleconditional.yaml

PLAY [testservers] ****
TASK [Gathering Facts] ****
ok: [192.168.145.146]

TASK [Installing Apache on CentOS] ****
changed: [192.168.145.146]

TASK [Install Apache on Ubuntu Server] ****
skipping: [192.168.145.146]

PLAY RECAP ****
192.168.145.146 : ok=2    changed=1    unreachable=0    failed=0    skipped=1    rescued=0    ignored=0

[root@ansible1 ansible]# rm -rf ansibleconditional.yaml
[root@ansible1 ansible]# vi ansibleconditional.yaml
[root@ansible1 ansible]# ansible-playbook ansibleconditional.yaml
```

14:24 / 18:33

Scroll for details

```
Ansible Tutorial For Beginners | When Conditional | Ansible Playbook Tutorial for Beginners
ok: [192.168.145.146]

TASK [Installing Apache on CentOS] ****
changed: [192.168.145.146]

TASK [Install Apache on Ubuntu Server] ****
skipping: [192.168.145.146]

PLAY RECAP ****
192.168.145.146 : ok=2    changed=1    unreachable=0    failed=0    skipped=1    rescued=0    ignored=0

[root@ansible1 ansible]# rm -rf ansibleconditional.yaml
[root@ansible1 ansible]# vi ansibleconditional.yaml
[root@ansible1 ansible]# ansible-playbook ansibleconditional.yaml

PLAY [testservers] ****

TASK [Gathering Facts] ****
ok: [192.168.145.146]

TASK [Run with items greater than 5] ****
skipping: [192.168.145.146] => (item=0)
skipping: [192.168.145.146] => (item=2)
skipping: [192.168.145.146] => (item=4)
changed: [192.168.145.146] => (item=6)
changed: [192.168.145.146] => (item=8)
changed: [192.168.145.146] => (item=10)

PLAY RECAP ****
192.168.145.146 : ok=2    changed=1    unreachable=0    failed=0    skipped=0    rescued=0    ignored=0

[root@ansible1 ansible]# 14:29 / 18:33
```

Scroll for details

Ansible Tutorial For Beginners | When Conditional | Ansible Playbook Tutorial for Beginners

! Conditional.yaml •

C > Users > k > Desktop > ! Conditional.yaml

```
1 ---  
2 - hosts: testservers  
3   become: yes  
4   tasks:  
5     - name: Checking the contents of folder  
6       command: "ls /root/ansible"  
7       register: contents  
8  
9     - name: Display the message  
10    debug:  
11      msg: "Directory is empty"  
12      when: contents.stdout == ""
```

Scroll for details

Ansible Tutorial For Beginners | When Conditional | Ansible Playbook Tutorial for Beginners



! Conditional.yaml

C:\> Users > k > Desktop > ! Conditional.yaml

```
1  ---
2  - hosts: testservers
3    become: yes
4    tasks:
5      - name: Checking the contents of folder
6        command: "ls /root/ansible"
7        register: contents
8
9      - name: Display the message
10        debug:
11          msg: "Directory is empty"
12          when: contents.stdout == ""
```



◀ ▶ ▶| 🔍 14:58 / 18:33

Scroll for details



Ansible Tutorial For Beginners | When Conditional | Ansible Playbook Tutorial for Beginners

! Conditional.yaml

C > Users > k > Desktop > ! Conditional.yaml

```
1 ---  
2 - hosts: testservers  
3   become: yes  
4   tasks:  
5     - name: Checking the contents of folder  
6       command: "ls /root/ansible"  
7       register: contents  
8  
9     - name: Display the message  
10    debug:  
11      msg: "Directory is empty"  
12      when: contents.stdout == ""
```

Scroll for details

Ansible Tutorial For Beginners | When Conditional | Ansible Playbook Tutorial for Beginners

! Conditional.yaml •

C:\> Users > k > Desktop > ! Conditional.yaml

```
1 ---  
2 - hosts: testservers  
3   become: yes  
4   tasks:  
5     - name: Checking the contents of folder  
6       command: "ls /root/ansible"  
7       register: contents  
8  
9     - name: Display the message  
10    debug:  
11      msg: "Directory is empty"  
12      when: contents.stdout == ""
```

Scroll for details

Ansible Tutorial For Beginners | When Conditional | Ansible Playbook Tutorial for Beginners



! Conditional.yaml

C: > Users > k > Desktop > ! Conditional.yaml

```
1  ---
2  - hosts: testservers
3    become: yes
4    tasks:
5      - name: Checking the contents of folder
6        command: "ls /root/ansible"
7        register: contents
8
9      - name: Display the message
10        debug:
11          msg: "Directory is empty"
12          when: contents.stdout == ""
```



◀ ▶ ▶| ⏪ 15:37 / 18:33

Scroll for details



Ansible Tutorial For Beginners | When Conditional | Ansible Playbook Tutorial for Beginners

! Conditional.yaml ●

C: > Users > k > Desktop > ! Conditional.yaml

```
1 ---  
2 - hosts: testservers  
3   become: yes  
4   tasks:  
5     - name: Checking the contents of folder  
6       command: ls -la /var/www/html  
7       register: folder_content  
8     - name: Displaying the contents of folder  
9       debug:  
10         msg: {{ folder_content.stdout }}  
11         when: folder_content.rc == 0
```

Change All Occurrences Ctrl+F2

Cut Ctrl+X

Copy Ctrl+C

Paste Ctrl+V

Command Palette... Ctrl+Shift+P

15:49 / 18:33

Scroll for details

Ansible Tutorial For Beginners | When Conditional | Ansible Playbook Tutorial for Beginners
[root@ansible1 ansible]# rm -rf ansibleconditional.yaml
[root@ansible1 ansible]# vi ansibleconditional.yaml
[root@ansible1 ansible]# clear

I

Ansible Tutorial For Beginners | When Conditional | Ansible Playbook Tutorial for Beginners

```
[root@ansible ansible]# rm -rf ansibleconditional.yaml
[root@ansible ansible]# vi ansibleconditional.yaml
[root@ansible ansible]# ansible-playbook ansibleconditional.yaml

PLAY [testservers] ****
TASK [Gathering Facts] ****
[
```

I

16:04 / 18:33

Scroll for details

CC HD

[root@ansible1 ansible]# rm -rf ansibleconditional.yaml
[root@ansible1 ansible]# vi ansibleconditional.yaml
[root@ansible1 ansible]# ansible-playbook ansibleconditional.yaml

```
PLAY [testservers] ****
TASK [Gathering Facts] ****
ok: [192.168.145.146]

TASK [Checking the contents of folder] ****
changed: [192.168.145.146]

TASK [Display the message] ****
ok: [192.168.145.146] => {
    "msg": "Directory is empty"
}

PLAY RECAP ****
192.168.145.146 : ok=3    changed=1    unreachable=0    failed=0    skipped=0    rescued=0    ignored=0
```

[root@ansible1 ansible]#

[root@localhost ansible]# ls
[root@localhost ansible]# pwd
/root/ansible
[root@localhost ansible]#



I

[root@localhost ansible]# ls
[root@localhost ansible]# pwd
/root/ansible
[root@localhost ansible]# vi test.txt

I

[root@localhost ansible]# ls
[root@localhost ansible]# pwd
/root/ansible
[root@localhost ansible]# vi test.txt

I

Ansible Tutorial For Beginners | When Conditional | Ansible

```
[root@localhost ansible]# ls
[root@localhost ansible]# pwd
/root/ansible
[root@localhost ansible]# vi test.txt
[root@localhost ansible]# ls
test.txt
[root@localhost ansible]#
```

Suggested: Ansible Tutorial For Beginners | Using Variables and F...

I

◀ ▶ ⏪ 🔍 16:36 / 18:33

Scroll for details

CC HD

[root@ansible1 ansible]# rm -rf ansibleconditional.yaml
[root@ansible1 ansible]# vi ansibleconditional.yaml
[root@ansible1 ansible]# ansible-playbook ansibleconditional.yaml

```
PLAY [testservers] ****
TASK [Gathering Facts] ****
ok: [192.168.145.146]

TASK [Checking the contents of folder] ****
changed: [192.168.145.146]

TASK [Display the message] ****
ok: [192.168.145.146] => {
    "msg": "Directory is empty"
}

PLAY RECAP ****
192.168.145.146 : ok=3    changed=1    unreachable=0    failed=0    skipped=0    rescued=0    ignored=0
```

[root@ansible1 ansible]# ansible-playbook ansibleconditional.yaml

[root@ansible1 ansible]# rm -rf ansibleconditional.yaml
[root@ansible1 ansible]# vi ansibleconditional.yaml
[root@ansible1 ansible]# ansible-playbook ansibleconditional.yaml

```
PLAY [testservers] *****  
  
TASK [Gathering Facts] *****  
ok: [192.168.145.146]  
  
TASK [Checking the contents of folder] *****  
changed: [192.168.145.146]  
  
TASK [Display the message] *****  
ok: [192.168.145.146] => {  
    "msg": "Directory is empty"  
}  
  
PLAY RECAP *****  
192.168.145.146 : ok=3    changed=1    unreachable=0    failed=0    skipped=0    rescued=0    ignored=0
```

[root@ansible1 ansible]# ansible-playbook ansibleconditional.yaml

```
PLAY [testservers] *****  
  
TASK [Gathering Facts] *****
```

```
Ansible Tutorial For Beginners | When Conditional | Ansible Playbook Tutorial for Beginners
```

```
TASK [Gathering Facts] ****
ok: [192.168.145.146]

TASK [Checking the contents of folder] ****
changed: [192.168.145.146]

TASK [Display the message] ****
ok: [192.168.145.146] => {
    "msg": "Directory is empty"
}

PLAY RECAP ****
192.168.145.146 : ok=3    changed=1    unreachable=0    failed=0    skipped=0    rescued=0    ignored=0

[root@ansible1 ansible]# ansible-playbook ansibleconditional.yaml

PLAY [testservers] ****

TASK [Gathering Facts] ****
ok: [192.168.145.146]

TASK [Checking the contents of folder] ****
changed: [192.168.145.146]

TASK [Display the message] ****
skipping: [192.168.145.146]

PLAY RECAP ****
192.168.145.146 : ok=2    changed=1    unreachable=0    failed=0    skipped=1    rescued=0    ignored=0
```

[root@ansible1 ansible]# 16:46 / 18:33

Scroll for details

```
Ahsible Tutorial For Beginners | When Conditional | Ansible Playbook Tutorial for Beginners
```

```
TASK [Gathering Facts] ****
ok: [192.168.145.146]

TASK [Checking the contents of folder] ****
changed: [192.168.145.146]

TASK [Display the message] ****
ok: [192.168.145.146] => {
    "msg": "Directory is empty"
}

PLAY RECAP ****
192.168.145.146 : ok=3    changed=1    unreachable=0    failed=0    skipped=0    rescued=0    ignored=0

[root@ansible1 ansible]# ansible-playbook ansibleconditional.yaml

PLAY [testservers] ****

TASK [Gathering Facts] ****
ok: [192.168.145.146]

TASK [Checking the contents of folder] ****
changed: [192.168.145.146]

TASK [Display the message] ****
skipping: [192.168.145.146]

PLAY RECAP ****
192.168.145.146 : ok=2    changed=1    unreachable=0    failed=0    skipped=1    rescued=0    ignored=0
```

[root@ansible1 ansible]# 16:53 / 18:33

Scroll for details

Ansible Tutorial For Beginners | When Conditional | Ansible Playbook Tutorial for Beginners

! Conditional.yaml

C:\> Users > k > Desktop > ! Conditional.yaml

```
1 ---  
2 - hosts: testservers  
3   become: yes  
4   tasks:  
5     - name: Checking the contents of folder  
6       command: "ls /root/ansible"  
7       register: contents  
8  
9     - name: Display the message  
10    debug:  
11      msg: "Directory is empty"  
12      when: contents.stdout == ""
```

Scroll for details

Ansible Tutorial For Beginners | When Conditional | Ansible Playbook Tutorial for Beginners

! Conditional.yaml ●

C:\> Users > k > Desktop > ! Conditional.yaml

```
1 ---  
2 - hosts: testservers  
3   become: yes  
4   tasks:  
5     - name: Checking the contents of folder  
6       command: "ls /root/ansible"  
7       register: contents  
8  
9     - name: Display the message  
10    debug:  
11      msg: "Directory is empty"  
12      when: contents.stdout == ""
```

Scroll for details

Ansible Tutorial For Beginners | When Conditional | Ansible Playbook Tutorial for Beginners



! Conditional.yaml ●

C:\> Users > k > Desktop > ! Conditional.yaml

```
1 - hosts: testservers
2   become: yes
3   roles:
4     - role: Installing Apache2
5       when: ansible_os_family == "RedHat"
```



◀ ▶ ▶| 🔍 17:22 / 18:33

Scroll for details



Ansible Vault Tutorial

- What is Ansible Vault?
- How to encrypt files using Ansible Vault
- How to view encrypted files

Ansible Vault Tutorial

- What is Ansible Vault?
- How to encrypt files using Ansible Vault
- How to view encrypted files
- How to decrypt files
- How to run Ansible playbook which are encrypted

[root@ansible1 ansiblevault]# ansible-vault encrypt ansiblevault.yml
New Vault password:
Confirm New Vault password:
Encryption successful
[root@ansible1 ansiblevault]# ca



◀ ▶ ⏪ ⏩ 🔍 0:05 / 13:37

Scroll for details



[root@ansible1 ansiblevault]# ansible-vault encrypt ansiblevault.yml
New Vault password:
Confirm New Vault password:
Encryption successful
[root@ansible1 ansiblevault]# cat ansiblevault.yml



◀ ▶ ⏪ ⏩ 0:07 / 13:37

Scroll for details



[root@ansible1 ansiblevault]# ansible-vault encrypt ansiblevault.yml
New Vault password:
Confirm New Vault password:
Encryption successful
[root@ansible1 ansiblevault]# cat ansiblevault.yml



```
[root@ansible1 ansiblevault]# ansible-vault encrypt ansiblevault.yml
New Vault password:
Confirm New Vault password:
Encryption successful
[root@ansible1 ansiblevault]# cat ansiblevault.yml
$ANSIBLE_VAULT;1.1;AES256
37373665666630646235343564653562363763343433353937326538623666613661373734333561
3737353936653937363662643064353233353837356237310a613639333063383233303836376436
35383330356364383437663631643433363039343863363733626566313062326265386261656635
3463353239333835310a353635613162376634663764303762356131356636353937653536393162
33346439326661363235393831333665353063633565623162626364353937326565633363343465
65333032313765373936396166366434353862366336656534613835373663646630343734316133
64376638653262376166363163356334663138663931316232323931393638326332343965386637
62333630626431326638633031393932646239373263376164343131393932333466643365393535
30353135303837623634356638333637643264663033633964656436636466343563353339313839
303535376132626366636232303534653613336393333364373061363361303630396534613537
62363832316331383731333663366566626161636632343265393539383139636235313261366230
39663066343331363938363538646465383637383034653036363264636134373638323263653437
65653265383835363133373832316463393632666664346463343866646131663139336635306334
353930356262313638666262313738373864343236646463306435366435633264623766336663
31306338383130636262653337336439313661656532613634366336653237303935306635316134
37386666323133386562623435363164613137363963316165336334363739613536633730373964
643161666616336664613236623861323236363237643963323962323138313039393734373763
656565396632303966393364663232613163636132653663346264393931363930396434336263
6438623164326266323761366666661656239623731646537326463643961653862363630393863
34623438373832343461
[root@ansible1 ansiblevault]#
```

Scroll for details

[root@ansible1 ansiblevault]# cat ansiblevault.yml

```
---
- hosts: testservers
  become: yes
  vars_files:
    - secret.yml

  tasks:
    - name: Installing Apache2
      yum:
        name: httpd
        state: installed

    - name: Enabling and starting the apache service
      service:
        name: httpd
        enabled: yes
        state: started
```

[root@ansible1 ansiblevault]# ansible-playbook ansiblevault.yml
ERROR! Attempting to decrypt but no vault secrets found
[root@ansible1 ansiblevault]# ansible-playbook --ansiblevault.yml

[root@ansible1 ansiblevault]# cat ansiblevault.yml

```
---
- hosts: testservers
  become: yes
  vars_files:
    - secret.yml

  tasks:
    - name: Installing Apache2
      yum:
        name: httpd
        state: installed

    - name: Enabling and starting the apache service
      service:
        name: httpd
        enabled: yes
        state: started
```

[root@ansible1 ansiblevault]# ansible-playbook ansiblevault.yml

ERROR! Attempting to decrypt but no vault secrets found

[root@ansible1 ansiblevault]# ansible-playbook --ask-vault-pass ansiblevault.yml

Vault password: [REDACTED]

Ansible Tutorial For Beginners

Ansible Vault Tutorial

- What is Ansible Vault?
- How to encrypt files using Ansible Vault
- How to view encrypted files
- How to decrypt files
- How to run Ansible playbook which are encrypted

Encrypting Content With Ansible Vault

Encrypting Content With Ansible Vault

- Ansible Vault is a feature that allows users to encrypt values and data structures within Ansible
- A utility called `ansible-vault` secures confidential data by encrypting it on disk allowing us to make use of in Ansible Ad-Hoc commands and Ansible Playbooks.
- Ansible Vault can encrypt variables, or even entire files and YAML playbooks

Encrypting Content With Ansible Vault

- Ansible Vault is a feature that allows users to encrypt values and data structures within Ansible
- A utility called `ansible-vault` secures confidential data by encrypting it on disk allowing us to make use of in Ansible Ad-Hoc commands and Ansible Playbooks.
- Ansible Vault can encrypt variables, or even entire files and YAML playbooks

Encrypting Content With Ansible Vault

- Ansible Vault is a feature that allows users to encrypt values and data structures within Ansible
- A utility called `ansible-vault` secures confidential data by encrypting it on disk allowing us to make use of in Ansible Ad-Hoc commands and Ansible Playbooks.
- Ansible Vault can encrypt variables, or even entire files and YAML playbooks

Ansible Tutorial For Beginners | Ansible Vault

AnsibleVault.yml - Demo - Visual Studio Code



! AnsibleVault.yml x ! AnsibleVault_files.yml
D: > Youtube Downloads > Yaml files > ! AnsibleVault.yml

```
1  ---
2  - hosts: testservers
3    become: yes
4    vars:
5      - ansible_sudo_pass: password
6
7
8    tasks:
9
10   - name: Installing Apache2
11     yum:
12       name: httpd
13       state: installed
14
15   - name: Enabling and starting the apache service
16     service:
17       name: httpd
18       enabled: yes
19       state: started
20
```

Ansible Tutorial For Beginners | Ansible Vault

AnsibleVault.yml - Demo - Visual Studio Code



! AnsibleVault.yml x ! AnsibleVault_files.yml
D: > Youtube Downloads > Yaml files > ! AnsibleVault.yml

```
1  ---
2  - hosts: testservers
3    become: yes
4    vars:
5      - ansible_sudo_pass: password
6
7
8    tasks:
9
10   - name: Installing Apache2
11     yum:
12       name: httpd
13       state: installed
14
15   - name: Enabling and starting the apache service
16     service:
17       name: httpd
18       enabled: yes
19       state: started
20
```

Ansible Tutorial For Beginners | Ansible Vault

AnsibleVault.yml - Demo - Visual Studio Code

AnsibleVault.yml X AnsibleVault_files.yml

D: > Youtube Downloads > Yaml files > AnsibleVault.yml

```
1  ---
2  - hosts: testservers
3    become: yes
4    vars:
5      - ansible_sudo_pass: password
6
7
8    tasks:
9
10   - name: Installing Apache2
11     yum:
12       name: httpd
13       state: installed
14
15   - name: Enabling and starting the apache service
16     service:
17       name: httpd
18       enabled: yes
19       state: started
20
```

◀ ▶ ▷ 🔍 3:28 / 13:37

Scroll for details

▶ 🔍 HD 🔍

Ansible Tutorial For Beginners | Ansible Vault

AnsibleVault.yml - Demo - Visual Studio Code

! AnsibleVault.yml x ! AnsibleVault_files.yml

D: > Youtube Downloads > Yaml files > ! AnsibleVault.yml

```
1 ---  
2 - hosts: all  
3   become: yes  
4   vars:  
5     va:  
6       ta:  
7         - name: Enabling and starting the apache service  
8           service:  
9             name: httpd  
10            enabled: yes  
11            state: started  
12  
13 - name: Apache Configuration  
14   yum:  
15     name: httpd  
16     state: installed  
17  
18 - name: Enabling and starting the apache service  
19   service:  
20     name: httpd  
21     enabled: yes  
22     state: started
```

- Change All Occurrences Ctrl+F2
- Cut Ctrl+X
- Copy Ctrl+C
- Paste Ctrl+V

Command Palette... Ctrl+Shift+P

◀ ▶ ▷ 🔍 4:03 / 13:37

Scroll for details



[root@ansible1 ansiblevault]# vi ansiblevault.yml



I

Ahsible Tutorial For Beginners | Ansible Vault



```
---
- hosts: testservers
  become: yes
  vars:
    - ansible_sudo_pass: password

  tasks:
    - name: Installing Apache2
      yum:
        name: httpd
        state: installed

    - name: Enabling and starting the apache service
      service:
        name: httpd
        enabled: yes
        state: started
```



Ansible Tutorial For Beginners | Ansible Vault

[root@ansible1 ansiblevault]# vi ansiblevault.yml

```
[root@ansible1 ansiblevault]# cat ansiblevault.yml

---

- hosts: testservers
  become: yes
  vars:
    - ansible_sudo_pass: password

  tasks:

    - name: Installing Apache2
      yum:
        name: httpd
        state: installed

    - name: Enabling and starting the apache service
      service:
        name: httpd
        enabled: yes
        state: started
[root@ansible1 ansiblevault]#
```



[root@ansible1 ansiblevault]# ansible-vault encrypt ansiblevault.yml
New Vault password:
Confirm New Vault password:
Encryption successful
[root@ansible1 ansiblevault]#



Ansible Tutorial For Beginners | Ansible Vault

```
[root@ansible1 ansiblevault]# ansible-vault encrypt ansiblevault.yml
New Vault password:
Confirm New Vault password:
Encryption successful
[root@ansible1 ansiblevault]# cat ansiblevault.yml
$ANSIBLE_VAULT;1.1;AES256
37373665666630646235343564653562363763343433353937326538623666613661373734333561
3737353936653937363662643064353233353837356237310a613639333063383233303836376436
35383330356364383437663631643433363039343863363733626566313062326265386261656635
3463353239333835310a353635613162376634663764303762356131356636353937653536393162
33346439326661363235393831333665353063633565623162626364353937326565633363343465
65333032313765373936396166366434353862366336656534613835373663646630343734316133
64376638653262376166363163356334663138663931316232323931393638326332343965386637
62333630626431326638633031393932646239373263376164343131393932333466643365393535
3035313530383762363435663833363764326466303363396465643663466343563353339313839
3035353761326263666362323035346536613336393333364373061363361303630396534613537
62363832316331383731333663366566626161636632343265393539383139636235313261366230
39663066343331363938363538646465383637383034653036363264636134373638323263653437
65653265383835363133373832316463393632666664346463343866646131663139336635306334
353930356262313638666262313738373864343236646463306435366435633264623766336663
31306338383130636262653337336439313661656532613634366336653237303935306635316134
37386666323133386562623435363164613137363963316165336334363739613536633730373964
643161666616336664613236623861323236363237643963323962323138313039393734373763
65656535396632303966393364663232613163636132653663346264393931363930396434336263
6438623164326266323761366666661656239623731646537326463643961653862363630393863
34623438373832343461
[root@ansible1 ansiblevault]#
```

[root@ansible1 ansiblevault]# ansible-playbook --ask-vault-pass ansiblevault.yml



I

◀ ▶ ⏪ 🔍 6:21 / 13:37

Scroll for details

▶ CC HD

Ansible Tutorial For Beginners | Ansible Vault

```
[root@ansible1 ansiblevault]# ansible-playbook --ask-vault-pass ansiblevault.yml
Vault password:

PLAY [testservers] ****
TASK [Gathering Facts] ****
ok: [192.168.145.146]

TASK [Installing Apache2] ****
ok: [192.168.145.146]

TASK [Enabling and starting the apache service] ****
ok: [192.168.145.146]

PLAY RECAP ****
192.168.145.146 : ok=3    changed=0    unreachable=0    failed=0    skipped=0    rescued=0    ignored=0

[root@ansible1 ansiblevault]#
```

[root@ansible1 ansiblevault]# ansible-vault decrypt ansiblevault.yml



I

◀ ▶ ⏪ ⏩ 🔍 7:15 / 13:37

Scroll for details

▶ CC HD

[root@ansible1 ansiblevault]# ansible-vault decrypt ansiblevault.yml
Vault password:
Decryption successful
[root@ansible1 ansiblevault]#



I

◀ ▶ ⏪ ⏩ 🔍 7:19 / 13:37

Scroll for details

▶ CC HD

Ansible Tutorial For Beginners | Ansible Vault
[root@ansible1 ansiblevault]# ansible-vault decrypt ansiblevault.yml
Vault password:
Decryption successful
[root@ansible1 ansiblevault]# cat ansiblevault.yml



I

◀ ▶ ⏪ ⏩ 🔍 7:24 / 13:37

Scroll for details

▶ CC HD

[root@ansible1 ansiblevault]# ansible-vault decrypt ansiblevault.yml
Vault password:
Decryption successful
[root@ansible1 ansiblevault]# cat ansiblevault.yml

```
---
```

```
- hosts: testservers
  become: yes
  vars:
    - ansible_sudo_pass: password
```

```
tasks:
```

```
- name: Installing Apache2
  yum:
    name: httpd
    state: installed
```

```
I
```

```
- name: Enabling and starting the apache service
  service:
    name: httpd
    enabled: yes
    state: started
```

```
[root@ansible1 ansiblevault]# █
```

Ansible Tutorial For Beginners | Ansible Vault
[root@ansible1 ansiblevault]# cat ansiblevault.yml

```
---
```

```
- hosts: testservers
  become: yes
  vars:
    - ansible_sudo_pass: password
```

```
tasks:
```

```
- name: Installing Apache2
  yum:
    name: httpd
    state: installed

- name: Enabling and starting the apache service
  service:
    name: httpd
    enabled: yes
    state: started
```

```
[root@ansible1 ansiblevault]#
```

[root@ansible1 ansiblevault]# cat ansiblevault.yml

```
---
```

```
- hosts: testservers
  become: yes
  vars:
    - ansible_sudo_pass: password
```

```
tasks:
```

```
- name: Installing Apache2
  yum:
    name: httpd
    state: installed

- name: Enabling and starting the apache service
  service:
    name: httpd
    enabled: yes
    state: started
```

```
[root@ansible1 ansiblevault]#
```

[root@ansible1 ansiblevault]# cat ansiblevault.yml

```
---
```

```
- hosts: testservers
  become: yes
  vars:
    - ansible_sudo_pass: password
```

```
tasks:
```

```
- name: Installing Apache2
  yum:
    name: httpd
    state: installed
```

```
- name: Enabling and starting the apache service
  service:
    name: httpd
    enabled: yes
    state: started
```

```
[root@ansible1 ansiblevault]# rm -f ansiblevault.yml
[root@ansible1 ansiblevault]# c
```

[root@ansible1 ansiblevault]# vi secret.yml

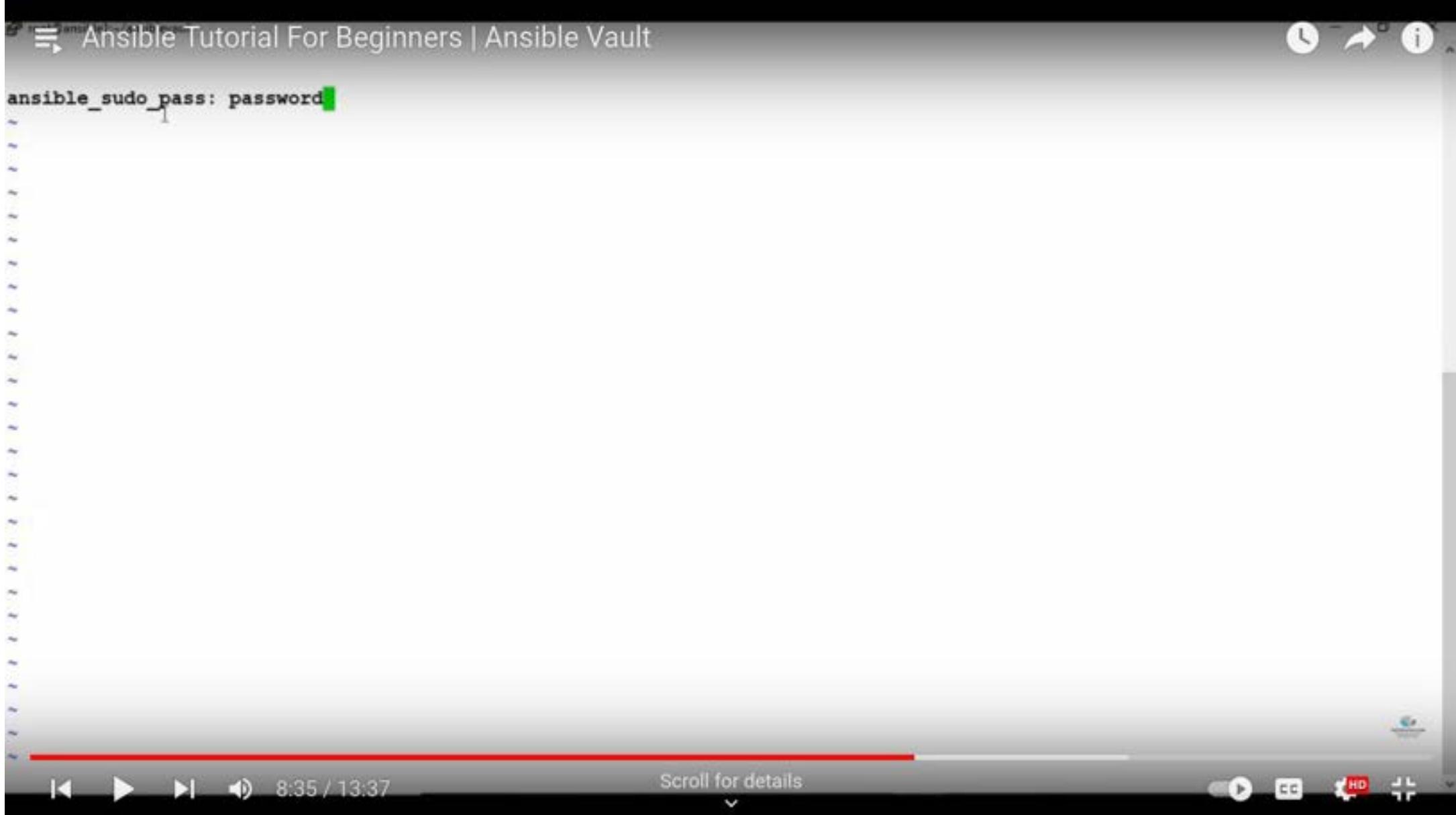


I

◀ ▶ ⏪ ⏩ 🔍 8:31 / 13:37

Scroll for details

▶ CC HD



Ansible Tutorial For Beginners | Ansible Vault
[root@ansible1 ansiblevault]# vi secret.yml
[root@ansible1 ansiblevault]# ls
secret.yml
[root@ansible1 ansiblevault]# vi



Ansible Tutorial For Beginners | Ansible Vault
[root@ansible1 ansiblevault]# vi secret.yml
[root@ansible1 ansiblevault]# ls
secret.yml
[root@ansible1 ansiblevault]# vi ansiblevault.yml



Ansible Tutorial For Beginners | Ansible Vault

AnsibleVault_files.yml - Demo - Visual Studio Code

! AnsibleVault.yml ! AnsibleVault_files.yml X

D: > Youtube Downloads > Yaml files > ! AnsibleVault_files.yml

```
1  ---
2  - hosts: testservers
3    become: yes
4    vars_files:
5      - secret.yml
6
7
8  tasks:
9
10   - name: Installing Apache2
11     yum:
12       name: httpd
13       state: installed
14
15   - name: Enabling and starting the apache service
16     service:
17       name: httpd
18       enabled: yes
19       state: started
20
```

Ansible Tutorial For Beginners | Ansible Vault

AnsibleVault_files.yml - Demo - Visual Studio Code

! AnsibleVault.yml ! AnsibleVault_files.yml X

D: > Youtube Downloads > Yaml files > ! AnsibleVault_files.yml

```
1  ---
2  - hosts: testservers
3    become: yes
4    vars_files:
5      - secret.yml
6
7
8  tasks:
9
10 - name: Installing Apache2
11   yum:
12     name: httpd
13     state: installed
14
15 - name: Enabling and starting the apache service
16   service:
17     name: httpd
18     enabled: yes
19     state: started
```



I

Ansible Tutorial For Beginners | Ansible Vault

```
---  
- hosts: testservers  
become: yes  
vars_files:  
  - secret.yml  
  
tasks:  
  
  - name: Installing Apache2  
    yum:  
      name: httpd  
      state: installed  
  
  - name: Enabling and starting the apache service  
    service:  
      name: httpd  
      enabled: yes  
      state: started
```

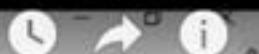
Ansible Tutorial For Beginners | Ansible Vault

```
[root@ansible1 ansiblevault]# vi secret.yml
[root@ansible1 ansiblevault]# ls
secret.yml
[root@ansible1 ansiblevault]# vi ansiblevault.yml
[root@ansible1 ansiblevault]# cat secret.yml

ansible_sudo_pass: password
[root@ansible1 ansiblevault]#
```



Ansible Tutorial For Beginners | Ansible Vault



```
[root@ansible1 ansiblevault]# vi secret.yml  
[root@ansible1 ansiblevault]# ls  
secret.yml  
[root@ansible1 ansiblevault]# vi ansiblevault.yml  
[root@ansible1 ansiblevault]# cat secret.yml
```

```
ansible_sudo_pass: password  
[root@ansible1 ansiblevault]# █
```

Ansible Tutorial For Beginners | Ansible Vault



```
[root@ansible1 ansiblevault]# vi secret.yml  
[root@ansible1 ansiblevault]# ls  
secret.yml  
[root@ansible1 ansiblevault]# vi ansiblevault.yml  
[root@ansible1 ansiblevault]# cat secret.yml
```

```
ansible_sudo_pass: password  
[root@ansible1 ansiblevault]# rm -f secret.yml  
[root@ansible1 ansiblevault]# █
```

[root@ansible1 ansiblevault]# ls
ansiblevault.yml
[root@ansible1 ansiblevault]# ansible-vault create secret.yml



I

◀ ▶ ▶| 🔍 10:04 / 13:37

Scroll for details

▶ CC HD

Ansible Tutorial For Beginners | Ansible Vault
[root@ansible1 ansiblevault]# ls
ansiblevault.yml
[root@ansible1 ansiblevault]# ansible-vault create secret.yml
New Vault password: █



I

◀ ▶ ⏪ ⏩ 10:07 / 13:37

Scroll for details

▶ CC HD

```
[root@ansible1 ansiblevault]# ls  
ansiblevault.yml  
[root@ansible1 ansiblevault]# ansible-vault create secret.yml  
New Vault password:  
Confirm New Vault password: [REDACTED]
```



I



```
[root@ansible1 ansiblevault]# ls  
ansiblevault.yml  
[root@ansible1 ansiblevault]# ansible-vault create secret.yml  
New Vault password:  
Confirm New Vault password:  
[root@ansible1 ansiblevault]#
```

Ansible Tutorial For Beginners | Ansible Vault



```
[root@ansible1 ansiblevault]# ls  
ansiblevault.yml  
[root@ansible1 ansiblevault]# ansible-vault create secret.yml  
New Vault password:  
Confirm New Vault password:  
[root@ansible1 ansiblevault]# cat secret.yml  
$ANSIBLE_VAULT;1.1;AES256  
34396163363665383631383634633230366330376263646538643062613036393037643837363663  
3439353137666438613537633064336431633731323034360a323666356537363337663462366265  
3534313038656232663465313263666663438616631376664646539633461366438353737643438  
3831326564663832620a373962623735623265303162343337356439383865353165663966663138  
3832633233613666313532303031356233333643466306539303762376236353937  
[root@ansible1 ansiblevault]# █
```

[root@ansible1 ansiblevault]# cat ansiblevault.yml

```
---
- hosts: testservers
  become: yes
  vars_files:
    - secret.yml

  tasks:
    - name: Installing Apache2
      yum:
        name: httpd
        state: installed

    - name: Enabling and starting the apache service
      service:
        name: httpd
        enabled: yes
        state: started
```

[root@ansible1 ansiblevault]#

[root@ansible1 ansiblevault]# cat ansiblevault.yml

```
---
- hosts: testservers
  become: yes
  vars_files:
    - secret.yml

  tasks:

    - name: Installing Apache2
      yum:
        name: httpd
        state: installed

    - name: Enabling and starting the apache service
      service:
        name: httpd
        enabled: yes
        state: started
```

[root@ansible1 ansiblevault]#

[root@ansible1 ansiblevault]# cat ansiblevault.yml

```
---
- hosts: testservers
  become: yes
  vars_files:
    - secret.yml

  tasks:

    - name: Installing Apache2
      yum:
        name: httpd
        state: installed

    - name: Enabling and starting the apache service
      service:
        name: httpd
        enabled: yes
        state: started
```

[root@ansible1 ansiblevault]# ansible-playbook ansiblevault.yml
ERROR! Attempting to decrypt but no vault secrets found
[root@ansible1 ansiblevault]#

[root@ansible1 ansiblevault]# cat ansiblevault.yml

```
---
- hosts: testservers
  become: yes
  vars_files:
    - secret.yml

  tasks:

    - name: Installing Apache2
      yum:
        name: httpd
        state: installed

    - name: Enabling and starting the apache service
      service:
        name: httpd
        enabled: yes
        state: started
```

[root@ansible1 ansiblevault]# ansible-playbook ansiblevault.yml
ERROR! Attempting to decrypt but no vault secrets found
[root@ansible1 ansiblevault]#

[root@ansible1 ansiblevault]# cat ansiblevault.yml

```
---
- hosts: testservers
  become: yes
  vars_files:
    - secret.yml

  tasks:

    - name: Installing Apache2
      yum:
        name: httpd
        state: installed

    - name: Enabling and starting the apache service
      service:
        name: httpd
        enabled: yes
        state: started
```

[root@ansible1 ansiblevault]# ansible-playbook ansiblevault.yml

ERROR! Attempting to decrypt but no vault secrets found

[root@ansible1 ansiblevault]# ansible-playbook --ask-vault-pass ansiblevault.yml

Vault password: [REDACTED]

Ansible Tutorial For Beginners | Ansible Vault



```
- name: Installing Apache2
  yum:
    name: httpd
    state: installed

- name: Enabling and starting the apache service
  service:
    name: httpd
    enabled: yes
    state: started
```

```
[root@ansible1 ansiblevault]# ansible-playbook ansiblevault.yml
ERROR! Attempting to decrypt but no vault secrets found
[root@ansible1 ansiblevault]# ansible-playbook --ask-vault-pass ansiblevault.yml
Vault password:
```

```
PLAY [testservers] ****
TASK [Gathering Facts] ****
ok: [192.168.145.146]

TASK [Installing Apache2] ****
ok: [192.168.145.146]

TASK [Enabling and starting the apache service] ****
ok: [192.168.145.146]
```

```
PLAY RECAP ****
192.168.145.146 : ok=3    changed=0    unreachable=0    failed=0    skipped=0    rescued=0    ignored=0
```

```
[root@ansible1 ansiblevault]#
```

Scroll for details





[root@ansible1 ansiblevault]# ansible-vault edit secret.yml
Vault password: [REDACTED]



I

Pull up for previous recording

◀ ▶ ⏪ ⏩ 🔍 11:44 / 13:37

Scroll for details

11:31

▶ CC HD

Ahsible Tutorial For Beginners | Ansible Vault

ansible_sudo_pass: password



I

"~/ansible/tmp/ansible-local-755eJYUUG5/tmp_r3_ANN.yml" Scroll 100% details



[root@ansible1 ansiblevault]# ansible-vault edit secret.yml
Vault password:
[root@ansible1 ansiblevault]# ansible-vault rekey secret.yml



I

◀ ▶ ▶| 🔍 12:17 / 13:37

Scroll for details

▶ CC HD

```
[root@ansible1 ansiblevault]# ansible-vault edit secret.yml
Vault password:
[root@ansible1 ansiblevault]# ansible-vault rekey secret.yml
Vault password:
New Vault password:
Confirm New Vault password:
Rekey successful
[root@ansible1 ansiblevault]#
```

I