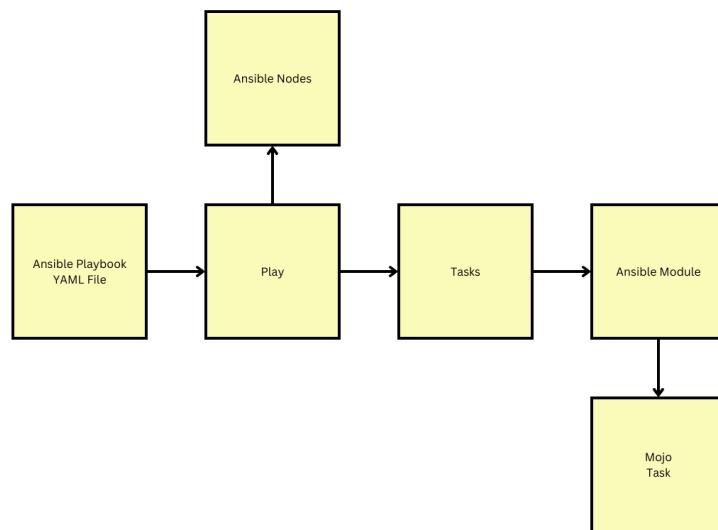


Day 4

Info - Ansible Idempotency property

- Idempotency is a feature of all Configuration Management tools including Ansible
- Majority of the ansible modules support idempotency but not all of them
 - For example
 - copy, service, apt, yum modules supports idempotency
 - shell modules doesn't idempotency
- Ansible color status
 - Green - Success with no change
 - Yellow - Success with change
- Example
 - when we install nginx web server in an Ubuntu Virtual machine via Ansible Playbook
 - ansible will check if already the Virtual machine has latest version of nginx installed, in case it found that the machine already has latest version then ansible do will nothing, it will simply report the task was successful executed and it didn't change anything on the machine(it will report in green color)
 - if suppose ansible found that the virtual machine has an older version of nginx, then ansible will upgrade the nginx to latest
 - if suppose ansible found that the virtual machine has no nginx then, it will install the latest version of nginx
 - once ansible has installed latest nginx when we execute the playbook it will simply report the status in green i.e it won't repeat the installation
 - this property is called Idempotency, we don't need to anything for this, ansible does this automatically

Info - Ansible Playbook Structure



Lab - Listing all the modules ansible supports

```
ansible-doc -l
```

Expected output

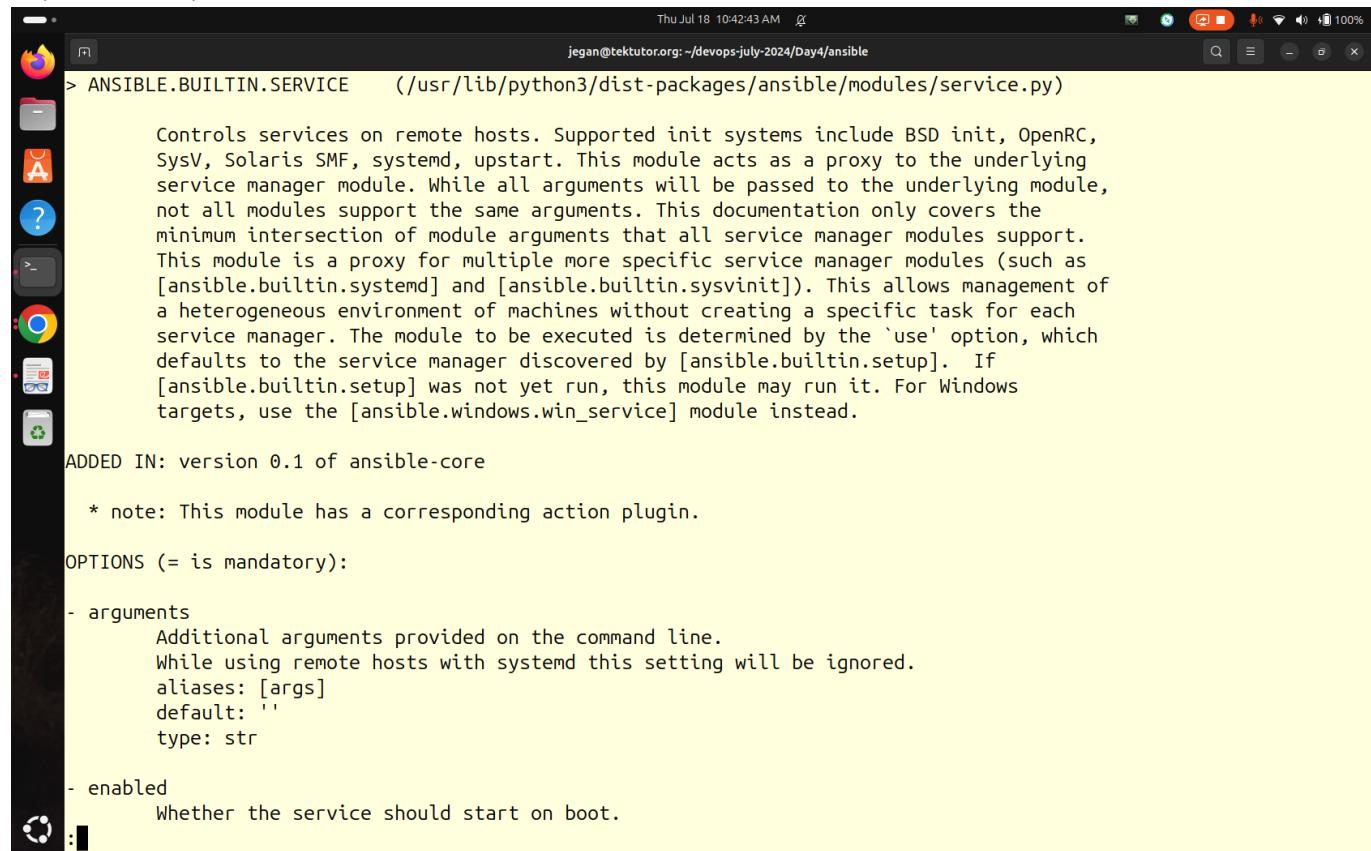
jegan@tektutor.org:~/devops-july-2024/Day4/ansible\$ ansible-doc -l

amazon.aws.autoscaling_group	Create or ...
amazon.aws.autoscaling_group_info	Gather inf...
amazon.aws.aws_az_info	Gather inf...
amazon.aws.aws_caller_info	Get inform...
amazon.aws.aws_region_info	Gather inf...
amazon.aws.backup_plan	Manage AWS...
amazon.aws.backup_plan_info	Describe A...
amazon.aws.backup_restore_job_info	List infor...
amazon.aws.backup_selection	Create, de...
amazon.aws.backup_selection_info	Describe A...
amazon.aws.backup_tag	Manage tag...
amazon.aws.backup_tag_info	List tags ...
amazon.aws.backup_vault	Manage AWS...
amazon.aws.backup_vault_info	Describe A...
amazon.aws.cloudformation	Create or ...
amazon.aws.cloudformation_info	Obtain inf...
amazon.aws.cloudtrail	manage Clo...
amazon.aws.cloudtrail_info	Gather inf...
amazon.aws.cloudwatch_metric_alarm	Create/upd...
amazon.aws.cloudwatch_metric_alarm_info	Gather inf...
amazon.aws.cloudwatchevent_rule	Manage Clo...
amazon.aws.cloudwatchlogs_log_group	create or ...
amazon.aws.cloudwatchlogs_log_group_info	Get inform...
amazon.aws.cloudwatchlogs_log_group_metric_filter	Manage Clo...
amazon.aws.ec2_ami	Create or ...
amazon.aws.ec2_ami_info	Gather inf...
amazon.aws.ec2_eip	manages EC...
amazon.aws.ec2_eip_info	List EC E...
amazon.aws.ec2_eni	Create and...
amazon.aws.ec2_eni_info	Gather inf...

Lab - Finding syntax details of a particular ansible module

```
ansible-doc service
ansible-doc shell
ansible-doc yum
ansible-doc apt
ansible-doc file
ansible-doc template
ansible-doc copy
ansible-doc command
```

Exptected output



The screenshot shows a terminal window on a Linux desktop environment. The title bar indicates it's running on a Firefox browser window titled 'jegan@tektutor.org: ~/devops-july-2024/Day4/ansible'. The terminal content displays the documentation for the 'ANSIBLE.BUILTIN.SERVICE' module, which is located at '/usr/lib/python3/dist-packages/ansible/modules/service.py'. The documentation describes the module's purpose as controlling services on remote hosts across various init systems like BSD init, OpenRC, SysV, Solaris SMF, systemd, and upstart. It acts as a proxy to the underlying service manager module, though not all modules support the same arguments. The documentation covers the minimum intersection of module arguments supported by all service manager modules. It also notes that this module is a proxy for more specific service manager modules such as [ansible.builtin.systemd] and [ansible.builtin.sysvinit]. This allows management of a heterogeneous environment of machines without creating a specific task for each service manager. The module to be executed is determined by the 'use' option, which defaults to the service manager discovered by [ansible.builtin.setup]. If [ansible.builtin.setup] was not yet run, this module may run it. For Windows targets, the [ansible.windows.win_service] module instead.

ADDED IN: version 0.1 of ansible-core

* note: This module has a corresponding action plugin.

OPTIONS (= is mandatory):

- **arguments**
Additional arguments provided on the command line.
While using remote hosts with systemd this setting will be ignored.
aliases: [args]
default: ''
type: str
- **enabled**
Whether the service should start on boot.

```
Thu Jul 18 10:43:02 AM jegan@tektutor.org: ~/devops-july-2024/Day4/ansible

> ANSIBLE.BUILTIN.SHELL      (/usr/lib/python3/dist-packages/ansible/modules/shell.py)

The [ansible.builtin.shell] module takes the command name followed by a list of
space-delimited arguments. Either a free form command or 'cmd' parameter is
required, see the examples. It is almost exactly like the [ansible.builtin.command]
module but runs the command through a shell ('/bin/sh') on the remote node. For
Windows targets, use the [ansible.windows.win_shell] module instead.

ADDED IN: version 0.2 of ansible-core

* note: This module has a corresponding action plugin.

OPTIONS (= is mandatory):

- chdir
    Change into this directory before running the command.
    default: null
    type: path
    added in: version 0.6 of ansible-core

- cmd
    The command to run followed by optional arguments.
    default: null
    type: str

- creates
    A filename, when it already exists, this step will *not* be run.
    default: null
    type: path
```

```
Thu Jul 18 10:43:14 AM jegan@tektutor.org: ~/devops-july-2024/Day4/ansible

> ANSIBLE.BUILTIN.YUM      (/usr/lib/python3/dist-packages/ansible/modules/yum.py)

Installs, upgrade, downgrades, removes, and lists packages and groups with the 'yum'
package manager. This module only works on Python 2. If you require Python 3 support
see the [ansible.builtin.dnf] module.

ADDED IN: historical

* note: This module has a corresponding action plugin.

OPTIONS (= is mandatory):

- allow_downgrade
    Specify if the named package and version is allowed to downgrade a maybe already
    installed higher version of that package. Note that setting allow_downgrade=True can
    make this module behave in a non-idempotent way. The task could end up with a set of
    packages that does not match the complete list of specified packages to install
    (because dependencies between the downgraded package and others can cause changes to
    the packages which were in the earlier transaction).
    default: 'no'
    type: bool
    added in: version 2.4 of ansible-core

- autoremove
    If 'true', removes all "leaf" packages from the system that were originally
    installed as dependencies of user-installed packages but which are no longer
    required by any such package. Should be used alone or when 'state' is 'absent'
    NOTE: This feature requires yum >= 3.4.3 (RHEL/CentOS 7+)
    default: 'no'
```

```
Thu Jul 18 10:43:25 AM  ✘
jegan@tektutor.org: ~/devops-july-2024/Day4/ansible

> ANSIBLE.BUILTIN.APT    (/usr/lib/python3/dist-packages/ansible/modules/apt.py)
    Manages `apt` packages (such as for Debian/Ubuntu).

ADDED IN: version 0.0.2 of ansible-core

OPTIONS (= is mandatory):

- allow_change_hold_packages
    Allows changing the version of a package which is on the apt hold list
    default: 'no'
    type: bool
    added in: version 2.13 of ansible-core

- allow_downgrade
    Corresponds to the `--allow-downgrades` option for `apt`.
    This option enables the named package and version to replace an already installed
    higher version of that package.
    Note that setting `allow_downgrade=true` can make this module behave in a non-
    idempotent way.
    (The task could end up with a set of packages that does not match the complete list
    of specified packages to install).
    `allow_downgrade` is only supported by `apt` and will be ignored if `aptitude` is
    detected or specified.
    aliases: [allow-downgrade, allow_downgrades, allow-downgrades]
    default: 'no'
    type: bool
    added in: version 2.12 of ansible-core

Thu Jul 18 10:43:38 AM  ✘
jegan@tektutor.org: ~/devops-july-2024/Day4/ansible

> ANSIBLE.BUILTIN.FILE    (/usr/lib/python3/dist-packages/ansible/modules/file.py)
    Set attributes of files, directories, or symlinks and their targets. Alternatively,
    remove files, symlinks or directories. Many other modules support the same options
    as the [ansible.builtin.file] module - including [ansible.builtin.copy],
    [ansible.builtin.template], and [ansible.builtin.assemble]. For Windows targets, use
    the [ansible.windows.win_file] module instead.

ADDED IN: historical

OPTIONS (= is mandatory):

- access_time
    This parameter indicates the time the file's access time should be set to.
    Should be 'preserve' when no modification is required, 'YYYYMMDDHHMM.SS' when using
    default time format, or 'now'.
    Default is 'None' meaning that 'preserve' is the default for
    `state=[file,directory,link,hard]' and 'now' is default for `state=touch'.
    default: null
    type: str
    added in: version 2.7 of ansible-core

- access_time_format
    When used with `access_time`, indicates the time format that must be used.
    Based on default Python format (see time.strftime doc).
    default: '%Y%m%d%H%M.%S'
    type: str
    added in: version 2.7 of ansible-core
```

```
Thu Jul 18 10:43:47 AM ✓
jegan@tektutor.org: ~/devops-july-2024/Day4/ansible

> ANSIBLE.BUILTIN.TEMPLATE    (/usr/lib/python3/dist-packages/ansible/modules/template.py)

Templates are processed by the Jinja2 templating language
<http://jinja.pocoo.org/docs/>. Documentation on the template formatting can be
found in the Template Designer Documentation
<http://jinja.pocoo.org/docs/templates/>. Additional variables listed below can be
used in templates. `ansible_managed` (configurable via the `defaults` section of
`ansible.cfg`) contains a string which can be used to describe the template name,
host, modification time of the template file and the owner uid. `template_host`
contains the node name of the template's machine. `template_uid` is the numeric user
id of the owner. `template_path` is the path of the template. `template_fullpath` is
the absolute path of the template. `template_destpath` is the path of the template
on the remote system (added in 2.8). `template_run_date` is the date that the
template was rendered.

ADDED IN: historical

* note: This module has a corresponding action plugin.

OPTIONS (= is mandatory):

- attributes
  The attributes the resulting filesystem object should have.
  To get supported flags look at the man page for 'chattr' on the target system.
  This string should contain the attributes in the same order as the one displayed by
  'lsattr'.
  The '=' operator is assumed as default, otherwise '+' or '-' operators need to be
  included in the string.
  aliases: [attr]
  default: null
```

```
Thu Jul 18 10:43:54 AM ✓
jegan@tektutor.org: ~/devops-july-2024/Day4/ansible

> ANSIBLE.BUILTIN.COPY    (/usr/lib/python3/dist-packages/ansible/modules/copy.py)

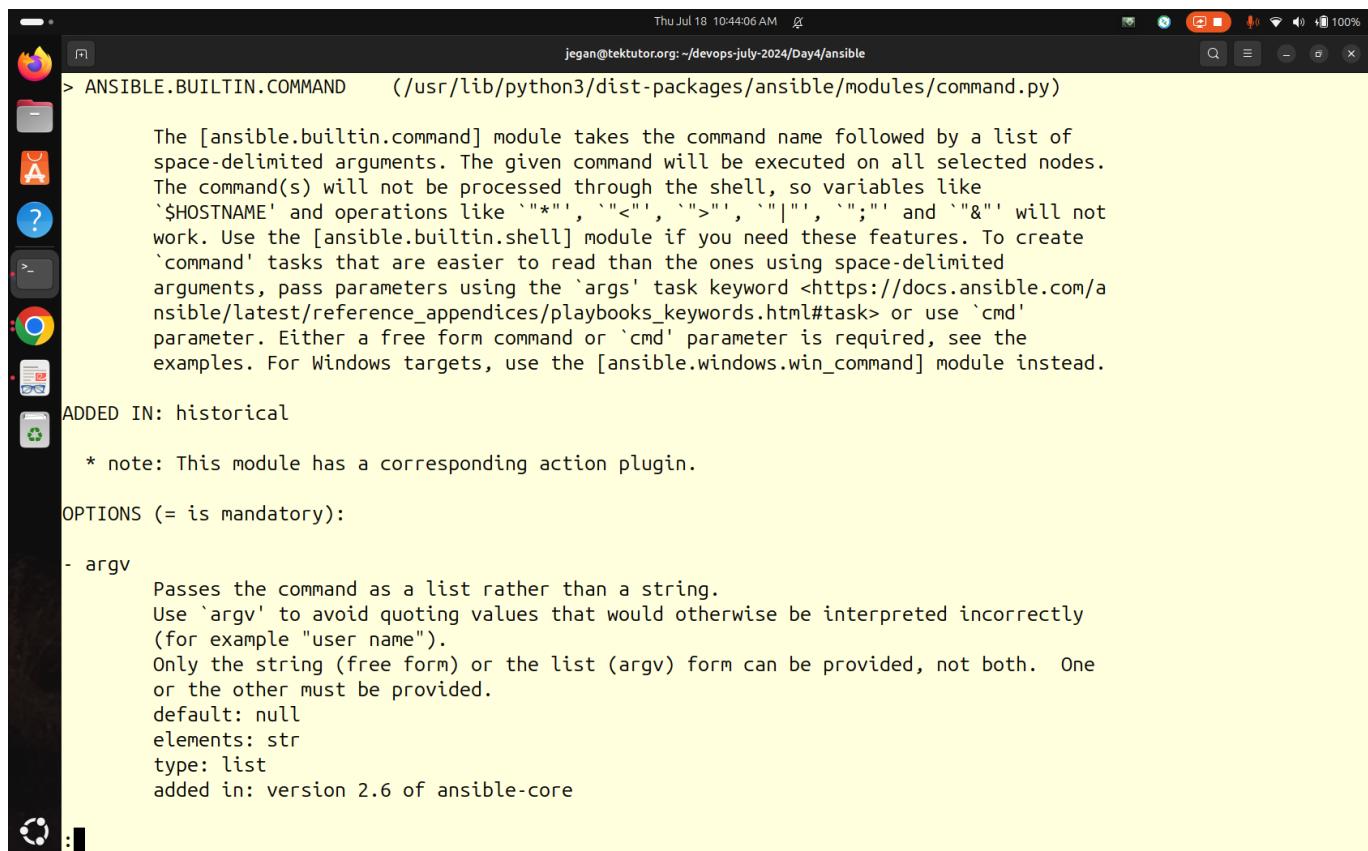
The [ansible.builtin.copy] module copies a file or a directory structure from the
local or remote machine to a location on the remote machine. File system meta-
information (permissions, ownership, etc.) may be set, even when the file or
directory already exists on the target system. Some meta-information may be copied
on request. Get meta-information with the [ansible.builtin.stat] module. Set meta-
information with the [ansible.builtin.file] module. Use the [ansible.builtin.fetch]
module to copy files from remote locations to the local box. If you need variable
interpolation in copied files, use the [ansible.builtin.template] module. Using a
variable with the 'content' parameter produces unpredictable results. For Windows
targets, use the [ansible.windows.win_copy] module instead.

ADDED IN: historical

* note: This module has a corresponding action plugin.

OPTIONS (= is mandatory):

- attributes
  The attributes the resulting filesystem object should have.
  To get supported flags look at the man page for 'chattr' on the target system.
  This string should contain the attributes in the same order as the one displayed by
  'lsattr'.
  The '=' operator is assumed as default, otherwise '+' or '-' operators need to be
  included in the string.
  aliases: [attr]
  default: null
  type: str
  added in: version 2.3 of ansible-core
```



The screenshot shows a terminal window with the following content:

```
Thu Jul 18 10:44:06 AM 2024
jegan@tektutor.org: ~/devops-july-2024/Day4/ansible

> ANSIBLE.BUILTIN.COMMAND (/usr/lib/python3/dist-packages/ansible/modules/command.py)

The [ansible.builtin.command] module takes the command name followed by a list of space-delimited arguments. The given command will be executed on all selected nodes. The command(s) will not be processed through the shell, so variables like '$HOSTNAME' and operations like `'*'`, `<'` , `>'` , `|` , `;` and `&` will not work. Use the [ansible.builtin.shell] module if you need these features. To create 'command' tasks that are easier to read than the ones using space-delimited arguments, pass parameters using the 'args' task keyword <https://docs.ansible.com/ansible/latest/reference\_appendices/playbooks\_keywords.html#task> or use 'cmd' parameter. Either a free form command or 'cmd' parameter is required, see the examples. For Windows targets, use the [ansible.windows.win_command] module instead.

ADDED IN: historical

* note: This module has a corresponding action plugin.

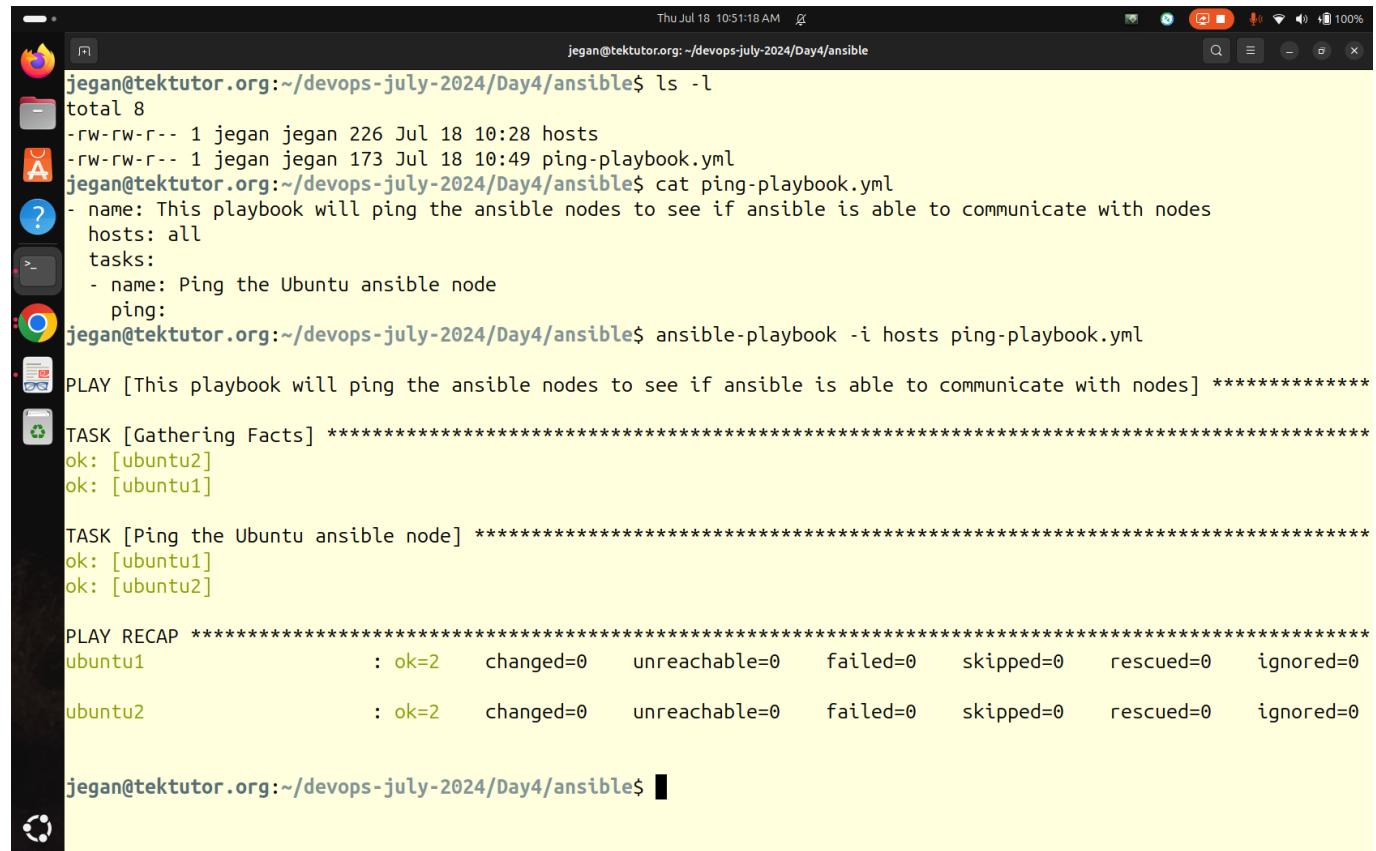
OPTIONS (= is mandatory):

- argv
    Passes the command as a list rather than a string.
    Use 'argv' to avoid quoting values that would otherwise be interpreted incorrectly (for example "user name").
    Only the string (free form) or the list (argv) form can be provided, not both. One or the other must be provided.
    default: null
    elements: str
    type: list
    added in: version 2.6 of ansible-core
```

Lab - Running your first ansible playbook

```
cd ~/devops-july-2024
git pull
cd Day4/ansible
cat ping-playbook.yml
ansible-playbook -i hosts ping-playbook.yml
```

Expected output



The screenshot shows a terminal window on a Linux desktop environment. The terminal title is "jegan@tektutor.org:~/devops-july-2024/Day4/ansible\$". The window contains the following text:

```
Thu Jul 18 10:51:18 AM 2024
jegan@tektutor.org:~/devops-july-2024/Day4/ansible$ ls -l
total 8
-rw-rw-r-- 1 jegan jegan 226 Jul 18 10:28 hosts
-rw-rw-r-- 1 jegan jegan 173 Jul 18 10:49 ping-playbook.yml
jegan@tektutor.org:~/devops-july-2024/Day4/ansible$ cat ping-playbook.yml
- name: This playbook will ping the ansible nodes to see if ansible is able to communicate with nodes
  hosts: all
  tasks:
    - name: Ping the Ubuntu ansible node
      ping:
jegan@tektutor.org:~/devops-july-2024/Day4/ansible$ ansible-playbook -i hosts ping-playbook.yml
PLAY [This playbook will ping the ansible nodes to see if ansible is able to communicate with nodes] ****
TASK [Gathering Facts] ****
ok: [ubuntu2]
ok: [ubuntu1]

TASK [Ping the Ubuntu ansible node] ****
ok: [ubuntu1]
ok: [ubuntu2]

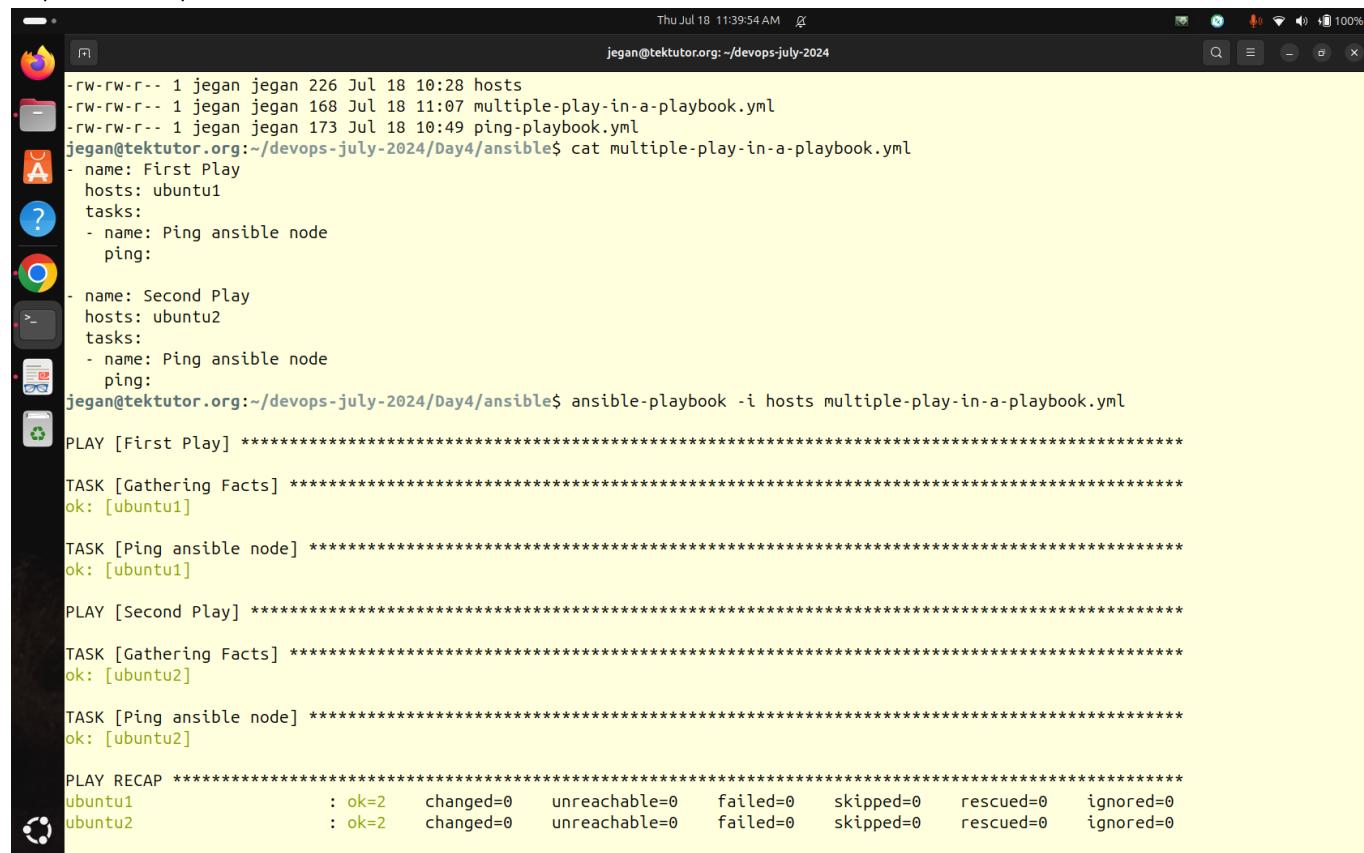
PLAY RECAP ****
ubuntu1           : ok=2    changed=0    unreachable=0    failed=0    skipped=0    rescued=0    ignored=0
ubuntu2           : ok=2    changed=0    unreachable=0    failed=0    skipped=0    rescued=0    ignored=0

jegan@tektutor.org:~/devops-july-2024/Day4/ansible$ █
```

Lab - Multiple Plays in a single Playbook

```
cd ~/devops-july-2024
git pull
cd Day4/ansible
cat multiple-play-in-a-playbook.yml
ansible-playbook -i hosts multiple-play-in-a-playbook.yml
```

Expected output



The screenshot shows a terminal window on a Linux desktop environment. The title bar indicates it's running on a Firefox browser window titled 'jegan@tektutor.org: ~/devops-july-2024'. The terminal content displays the execution of an Ansible playbook named 'multiple-play-in-a-playbook.yml'. The playbook defines two plays: 'First Play' targeting host 'ubuntu1' with a ping task, and 'Second Play' targeting host 'ubuntu2' with a ping task. The terminal output shows the playbook being loaded, the plays being executed, and finally a recap of the results. The recap table shows that both hosts ('ubuntu1' and 'ubuntu2') had 2 tasks successful ('ok=2'), 0 changed, 0 unreachable, 0 failed, 0 skipped, 0 rescued, and 0 ignored.

```
-rw-rw-r-- 1 jegan jegan 226 Jul 18 10:28 hosts
-rw-rw-r-- 1 jegan jegan 168 Jul 18 11:07 multiple-play-in-a-playbook.yml
-rw-rw-r-- 1 jegan jegan 173 Jul 18 10:49 ping-playbook.yml
jegan@tektutor.org:~/devops-july-2024$ cat multiple-play-in-a-playbook.yml
- name: First Play
  hosts: ubuntu1
  tasks:
    - name: Ping ansible node
      ping:

- name: Second Play
  hosts: ubuntu2
  tasks:
    - name: Ping ansible node
      ping:
jegan@tektutor.org:~/devops-july-2024$ ansible-playbook -i hosts multiple-play-in-a-playbook.yml
PLAY [First Play] ****
TASK [Gathering Facts] ****
ok: [ubuntu1]

TASK [Ping ansible node] ****
ok: [ubuntu1]

PLAY [Second Play] ****
TASK [Gathering Facts] ****
ok: [ubuntu2]

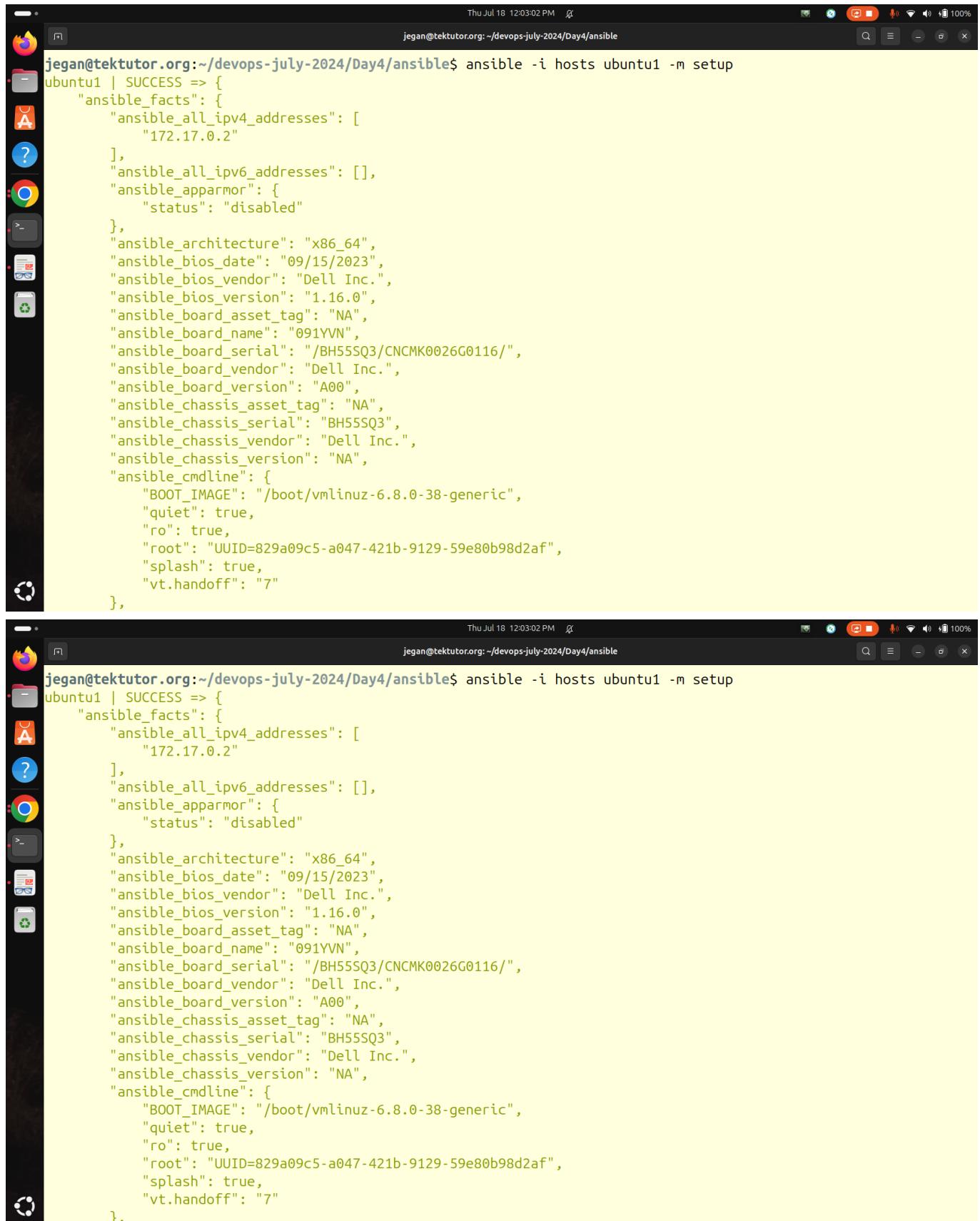
TASK [Ping ansible node] ****
ok: [ubuntu2]

PLAY RECAP ****
ubuntu1           : ok=2    changed=0    unreachable=0    failed=0     skipped=0    rescued=0    ignored=0
ubuntu2           : ok=2    changed=0    unreachable=0    failed=0     skipped=0    rescued=0    ignored=0
```

Lab - Understanding Ansible Setup module

```
cd ~/devops-july-2024
git pull
cd Day4/ansible
ansible -i hosts ubuntu1 -m setup
```

Expected output



```
jegan@tektutor.org:~/devops-july-2024/Day4/ansible$ ansible -i hosts ubuntu1 -m setup
ubuntu1 | SUCCESS => {
    "ansible_facts": {
        "ansible_all_ipv4_addresses": [
            "172.17.0.2"
        ],
        "ansible_all_ipv6_addresses": [],
        "ansible_apparmor": {
            "status": "disabled"
        },
        "ansible_architecture": "x86_64",
        "ansible_bios_date": "09/15/2023",
        "ansible_bios_vendor": "Dell Inc.",
        "ansible_bios_version": "1.16.0",
        "ansible_board_asset_tag": "NA",
        "ansible_board_name": "091YVN",
        "ansible_board_serial": "/BH55SQ3/CNCMK0026G0116/",
        "ansible_board_vendor": "Dell Inc.",
        "ansible_board_version": "A00",
        "ansible_chassis_asset_tag": "NA",
        "ansible_chassis_serial": "BH55SQ3",
        "ansible_chassis_vendor": "Dell Inc.",
        "ansible_chassis_version": "NA",
        "ansible_cmdline": {
            "BOOT_IMAGE": "/boot/vmlinuz-6.8.0-38-generic",
            "quiet": true,
            "ro": true,
            "root": "UUID=829a09c5-a047-421b-9129-59e80b98d2af",
            "splash": true,
            "vt.handoff": "7"
        },
    }
}

jegan@tektutor.org:~/devops-july-2024/Day4/ansible$ ansible -i hosts ubuntu1 -m setup
ubuntu1 | SUCCESS => {
    "ansible_facts": {
        "ansible_all_ipv4_addresses": [
            "172.17.0.2"
        ],
        "ansible_all_ipv6_addresses": [],
        "ansible_apparmor": {
            "status": "disabled"
        },
        "ansible_architecture": "x86_64",
        "ansible_bios_date": "09/15/2023",
        "ansible_bios_vendor": "Dell Inc.",
        "ansible_bios_version": "1.16.0",
        "ansible_board_asset_tag": "NA",
        "ansible_board_name": "091YVN",
        "ansible_board_serial": "/BH55SQ3/CNCMK0026G0116/",
        "ansible_board_vendor": "Dell Inc.",
        "ansible_board_version": "A00",
        "ansible_chassis_asset_tag": "NA",
        "ansible_chassis_serial": "BH55SQ3",
        "ansible_chassis_vendor": "Dell Inc.",
        "ansible_chassis_version": "NA",
        "ansible_cmdline": {
            "BOOT_IMAGE": "/boot/vmlinuz-6.8.0-38-generic",
            "quiet": true,
            "ro": true,
            "root": "UUID=829a09c5-a047-421b-9129-59e80b98d2af",
            "splash": true,
            "vt.handoff": "7"
        },
    }
}
```

Things to note

- ansible setup module will be the default module which will be executed in every Playbook
- the setup module will retrieve many details about the ansible node

- just to give an idea
 - setup module collects details like
 - hostname of the ansible node
 - Operating System details
 - System Hardware details
 - Operating System Family
 - OS version
 - Python version installed on the node
 - Package Manager supported on the node

Lab - Install nginx in Ansible nodes using Ansible playbook

Install curl utility locally

```
sudo apt install -y curl
```

Now you may proceed with the below exercise

```
cd ~/devops-july-2024
git pull
cd Day4/ansible
cat install-nginx-playbook.yml
ansible-playbook -i hosts install-nginx-playbook.yml
curl http://localhost:8001
curl http://localhost:8002
```

Expected output

```
Thu Jul 18 12:17:24 PM ✘
jegan@tektutor.org:~/devops-july-2024/Day4/ansible$ ls
hosts install-nginx-playbook.yml multiple-play-in-a-playbook.yml ping-playbook.yml
jegan@tektutor.org:~/devops-july-2024/Day4/ansible$ ansible-playbook -i hosts install-nginx-playbook.yml

PLAY [This playbook will install nginx, configure web root folder and deploys custom web page] ****
TASK [Gathering Facts] ****
ok: [ubuntu1]
ok: [ubuntu2]

TASK [Install nginx in Ubuntu node] ****
changed: [ubuntu2]
changed: [ubuntu1]

PLAY RECAP ****
ubuntu1 : ok=2    changed=1    unreachable=0    failed=0    skipped=0    rescued=0    ignored=0
ubuntu2 : ok=2    changed=1    unreachable=0    failed=0    skipped=0    rescued=0    ignored=0

jegan@tektutor.org:~/devops-july-2024/Day4/ansible$ curl http://localhost:8001
curl: (56) Recv failure: Connection reset by peer
jegan@tektutor.org:~/devops-july-2024/Day4/ansible$ curl http://localhost:8002
curl: (56) Recv failure: Connection reset by peer
jegan@tektutor.org:~/devops-july-2024/Day4/ansible$ ansible -i hosts ubuntu1 -m shell -a "service nginx status"
ubuntu1 | FAILED | rc=3 >>
 * nginx is not running non-zero return code
jegan@tektutor.org:~/devops-july-2024/Day4/ansible$ ansible -i hosts ubuntu2 -m shell -a "service nginx status"
ubuntu2 | FAILED | rc=3 >>
 * nginx is not running non-zero return code
jegan@tektutor.org:~/devops-july-2024/Day4/ansible$
```

```
Thu Jul 18 12:34:11 PM ✘
jegan@tektutor.org:~/devops-july-2024/Day4/ansible$ ansible-playbook -i hosts install-nginx-playbook.yml

PLAY [This playbook will install nginx, configure web root folder and deploys custom web page] ****
TASK [Gathering Facts] ****
ok: [ubuntu2]
ok: [ubuntu1]

TASK [Install nginx in Ubuntu nodes] ****
ok: [ubuntu1]
ok: [ubuntu2]

TASK [Install curl utility in Ubuntu nodes] ****
changed: [ubuntu2]
changed: [ubuntu1]

TASK [Start the nginx service] ****
changed: [ubuntu2]
changed: [ubuntu1]

PLAY RECAP ****
ubuntu1 : ok=4    changed=2    unreachable=0    failed=0    skipped=0    rescued=0    ignored=0
ubuntu2 : ok=4    changed=2    unreachable=0    failed=0    skipped=0    rescued=0    ignored=0

jegan@tektutor.org:~/devops-july-2024/Day4/ansible$ curl http://localhost:8001
<!DOCTYPE html>
<html>
<head>
<title>Welcome to nginx!</title>
```

```
Thu Jul 18 12:34:51 PM ✓
jegan@tektutor.org:~/devops-july-2024/Day4/ansible$ vim install-nginx-playbook.yml
jegan@tektutor.org:~/devops-july-2024/Day4/ansible$ ansible-playbook -i hosts install-nginx-playbook.yml
PLAY [This playbook will install nginx, configure web root folder and deploys custom web page] ****
TASK [Gathering Facts] ****
ok: [ubuntu2]
ok: [ubuntu1]

TASK [Install nginx in Ubuntu nodes] ****
ok: [ubuntu2]
ok: [ubuntu1]

TASK [Install curl utility in Ubuntu nodes] ****
ok: [ubuntu1]
ok: [ubuntu2]

TASK [Start the nginx service] ****
ok: [ubuntu2]
ok: [ubuntu1]

PLAY RECAP ****
ubuntu1 : ok=4    changed=0    unreachable=0    failed=0    skipped=0    rescued=0    ignored=0
ubuntu2 : ok=4    changed=0    unreachable=0    failed=0    skipped=0    rescued=0    ignored=0

jegan@tektutor.org:~/devops-july-2024/Day4/ansible$
```

```
Thu Jul 18 12:38:51 PM ✓
jegan@tektutor.org:~/devops-july-2024/Day4/ansible$ curl http://localhost:8001
<!DOCTYPE html>
<html>
<head>
<title>Welcome to nginx!</title>
<style>
html { color-scheme: light dark; }
body { width: 35em; margin: 0 auto;
font-family: Tahoma, Verdana, Arial, sans-serif; }
</style>
</head>
<body>
<h1>Welcome to nginx!</h1>
<p>If you see this page, the nginx web server is successfully installed and
working. Further configuration is required.</p>

<p>For online documentation and support please refer to
<a href="http://nginx.org/">nginx.org</a>.<br/>
Commercial support is available at
<a href="http://nginx.com/">nginx.com</a>.</p>

<p><em>Thank you for using nginx.</em></p>
</body>
</html>
jegan@tektutor.org:~/devops-july-2024/Day4/ansible$ curl http://localhost:8002
<!DOCTYPE html>
<html>
<head>
<title>Welcome to nginx!</title>
<style>
html { color-scheme: light dark; }
```

```
</html>
<html>
<!DOCTYPE html>
<head>
<title>Welcome to nginx!</title>
<style>
html { color-scheme: light dark; }
body { width: 35em; margin: 0 auto;
font-family: Tahoma, Verdana, Arial, sans-serif; }
</style>
</head>
<body>
<h1>Welcome to nginx!</h1>
<p>If you see this page, the nginx web server is successfully installed and
working. Further configuration is required.</p>

<p>For online documentation and support please refer to
<a href="http://nginx.org/">nginx.org</a>.<br/>
Commercial support is available at
<a href="http://nginx.com/">nginx.com</a>.</p>

<p><em>Thank you for using nginx.</em></p>
</body>
</html>
jegan@tektutor.org:~/devops-july-2024/Day4/ansible$ vim install-nginx-playbook.yml
jegan@tektutor.org:~/devops-july-2024/Day4/ansible$ ansible-playbook -i hosts install-nginx-playbook.yml

PLAY [This playbook will install nginx, configure web root folder and deploys custom web page] ****
TASK [Gathering Facts] ****
ok: [ubuntut2]
```

Things to note

- ansible apt module is idempotent, hence only the first time it will install latest version of nginx, subsequent times we execute the playbook, ansible will report in green color(meaning - it won't reinstall i.e ansible will report - success with no change)
- shell module isn't idempotent, hence must be used only if no other option is available/possible

Running the playbook with multiple play, the first play will install curl on your local machine

```
cd ~/devops-july-2024
git pull
cd Day4/ansible
cat install-nginx-playbook.yml
ansible-playbook -i hosts install-nginx-playbook.yml --ask-become-pass
```

Expected output



The screenshot shows a terminal window on a Linux desktop environment. The title bar indicates it's running on jegan@tektutor.org at 12:44:21 PM. The terminal displays the execution of an Ansible playbook named 'install-nginx-playbook.yml'. The output shows various tasks being run on hosts 'localhost', 'ubuntu1', and 'ubuntu2', including gathering facts, installing curl, and configuring nginx. The final 'PLAY RECAP' section summarizes the results for each host.

```
jegan@tektutor.org:~/devops-july-2024/Day4/ansible$ ansible-playbook -i hosts install-nginx-playbook.yml --ask-become-pass
BECOME password:

PLAY [This play will install curl utility on your RPS Ubuntu cloud machine] ****
TASK [Gathering Facts] ****
ok: [localhost]

TASK [Install curl] ****
changed: [localhost]

PLAY [This playbook will install nginx, configure web root folder and deploys custom web page] ****
TASK [Gathering Facts] ****
ok: [ubuntu1]
ok: [ubuntu2]

TASK [Install nginx in Ubuntu nodes] ****
ok: [ubuntu2]
ok: [ubuntu1]

TASK [Install curl utility in Ubuntu nodes] ****
ok: [ubuntu1]
ok: [ubuntu2]

TASK [Start the nginx service] ****
ok: [ubuntu2]
ok: [ubuntu1]

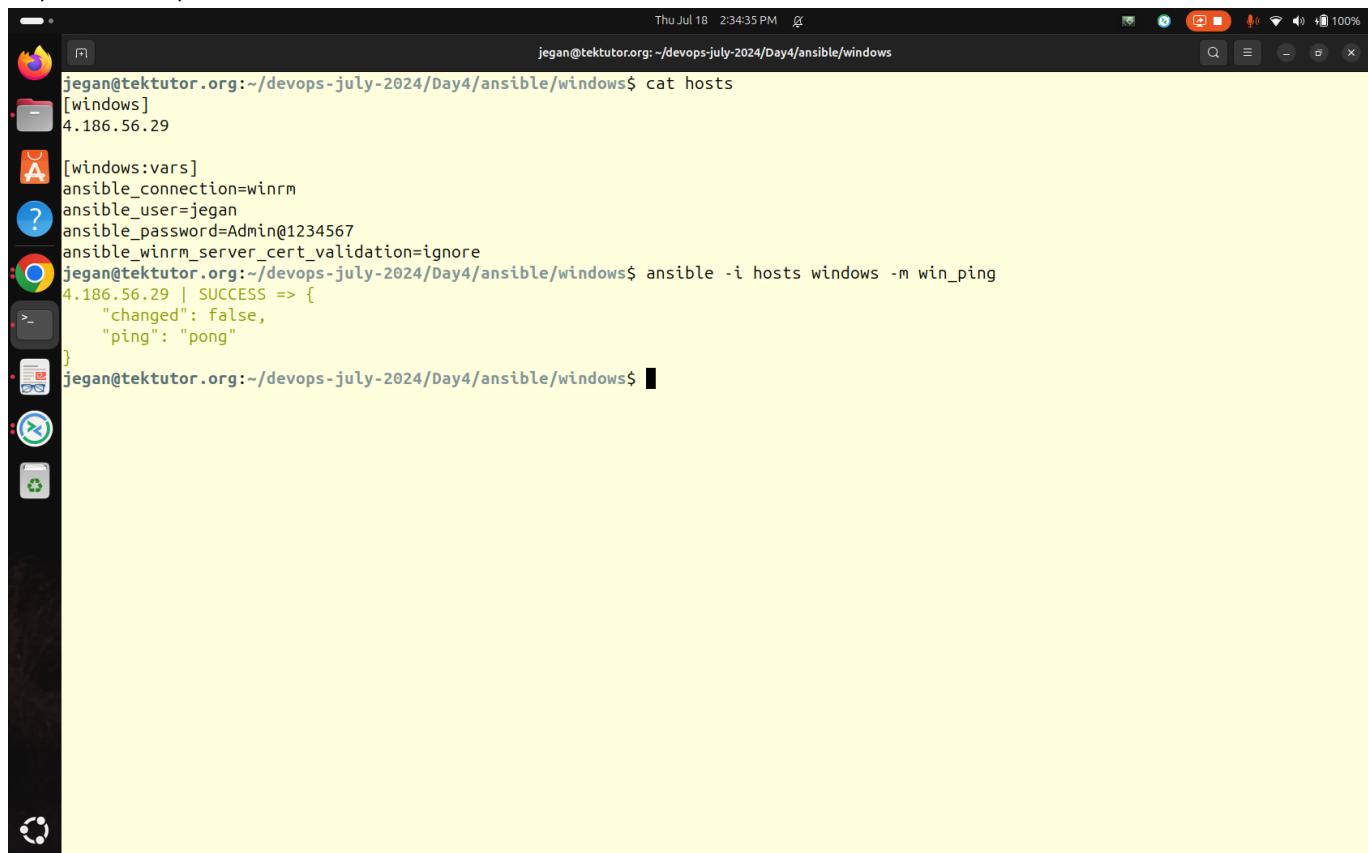
PLAY RECAP ****
localhost              : ok=2    changed=1    unreachable=0    failed=0    skipped=0    rescued=0    ignored=0
ubuntu1                : ok=4    changed=0    unreachable=0    failed=0    skipped=0    rescued=0    ignored=0
ubuntu2                : ok=4    changed=0    unreachable=0    failed=0    skipped=0    rescued=0    ignored=0

jegan@tektutor.org:~/devops-july-2024/Day4/ansible$
```

Demo - Ansible ping a windows virtual machine from Azure portal

```
cd ~/devops-july-2024
git pull
cd Day4/ansible/windows
cat hosts
ansible -i hosts windows -m win_ping
```

Expected output



The screenshot shows a terminal window on a Linux desktop environment. The terminal title is "jegan@tektutor.org:~/devops-july-2024/Day4/ansible/windows\$". The window contains the following text:

```
jegan@tektutor.org:~/devops-july-2024/Day4/ansible/windows$ cat hosts
[windows]
4.186.56.29

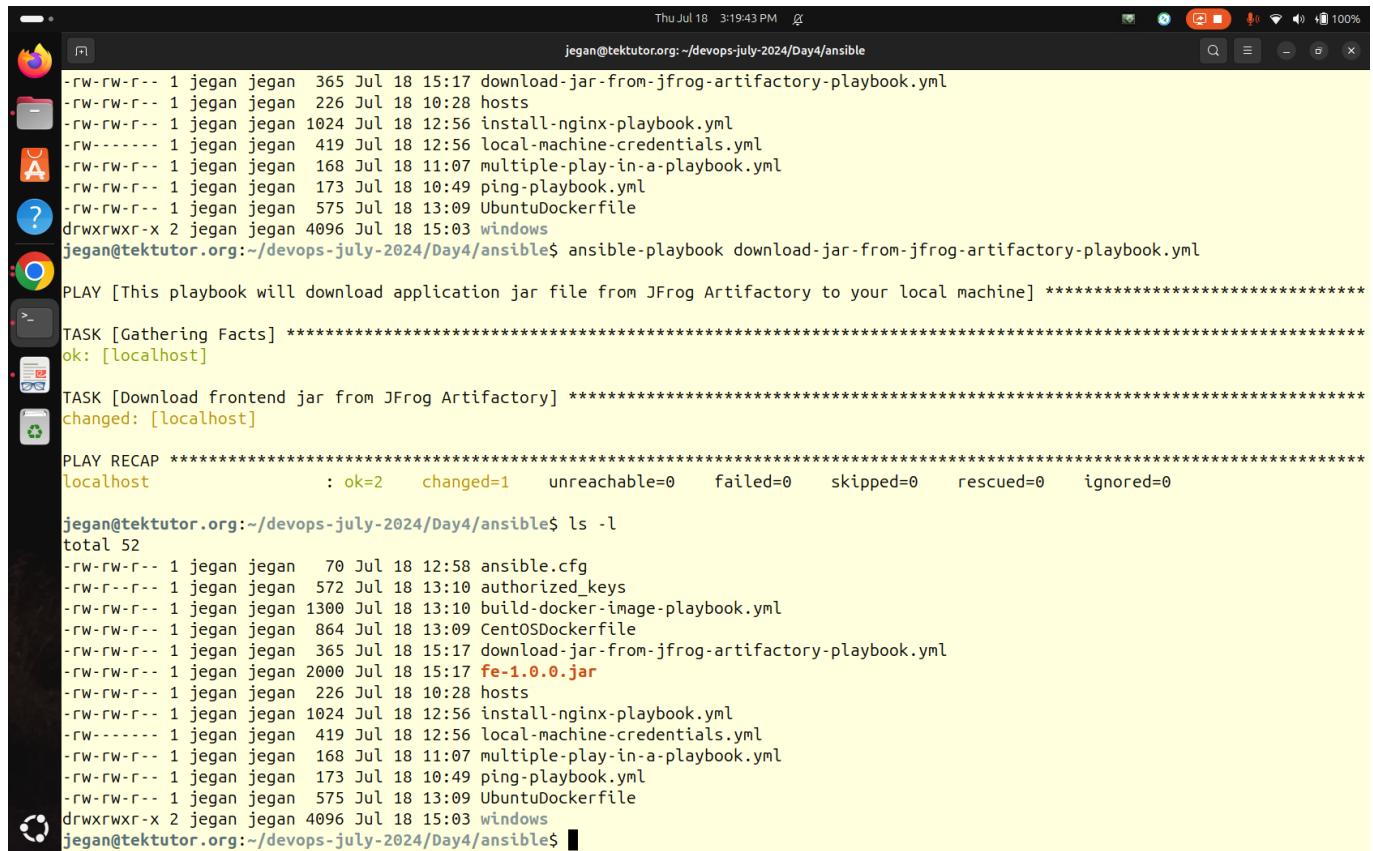
[jenkins:vars]
ansible_connection=winrm
ansible_user=jegan
ansible_password=Admin@1234567
ansible_winrm_server_cert_validation=ignore
jegan@tektutor.org:~/devops-july-2024/Day4/ansible/windows$ ansible -i hosts windows -m win_ping
4.186.56.29 | SUCCESS => {
    "changed": false,
    "ping": "pong"
}
jegan@tektutor.org:~/devops-july-2024/Day4/ansible/windows$
```

Lab - Downloading jar from JFrog Artifactory

Make sure your JFrog Artifactory container is running and it has the jar file we are downloading.

```
cd ~/devops-july-2024
git pull
cd Day4/ansible
cat download-jar-from-jfrog-artifactory-playbook.yml
ls -l
ansible-playbook download-jar-from-jfrog-artifactory-playbook.yml
ls -l
```

Expected output



```
Thu Jul 18 3:19:43 PM 100%
jegan@tektutor.org:~/devops-july-2024/Day4/ansible
-rw-rw-r-- 1 jegan jegan 365 Jul 18 15:17 download-jar-from-jfrog-artifactory-playbook.yml
-rw-rw-r-- 1 jegan jegan 226 Jul 18 10:28 hosts
-rw-rw-r-- 1 jegan jegan 1024 Jul 18 12:56 install-nginx-playbook.yml
-rw----- 1 jegan jegan 419 Jul 18 12:56 local-machine-credentials.yml
-rw-rw-r-- 1 jegan jegan 168 Jul 18 11:07 multiple-play-in-a-playbook.yml
-rw-rw-r-- 1 jegan jegan 173 Jul 18 10:49 ping-playbook.yml
-rw-rw-r-- 1 jegan jegan 575 Jul 18 13:09 UbuntuDockerfile
drwxrwxr-x 2 jegan jegan 4096 Jul 18 15:03 windows
jegan@tektutor.org:~/devops-july-2024/Day4/ansible$ ansible-playbook download-jar-from-jfrog-artifactory-playbook.yml

PLAY [This playbook will download application jar file from JFrog Artifactory to your local machine] ****
TASK [Gathering Facts] ****
ok: [localhost]

TASK [Download frontend jar from JFrog Artifactory] ****
changed: [localhost]

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

jegan@tektutor.org:~/devops-july-2024/Day4/ansible$ ls -l
total 52
-rw-rw-r-- 1 jegan jegan 70 Jul 18 12:58 ansible.cfg
-rw-r--r-- 1 jegan jegan 572 Jul 18 13:10 authorized_keys
-rw-rw-r-- 1 jegan jegan 1300 Jul 18 13:10 build-docker-image-playbook.yml
-rw-rw-r-- 1 jegan jegan 864 Jul 18 13:09 CentOSDockerfile
-rw-rw-r-- 1 jegan jegan 365 Jul 18 15:17 download-jar-from-jfrog-artifactory-playbook.yml
-rw-rw-r-- 1 jegan jegan 2000 Jul 18 15:17 fe-1.0.0.jar
-rw-rw-r-- 1 jegan jegan 226 Jul 18 10:28 hosts
-rw-rw-r-- 1 jegan jegan 1024 Jul 18 12:56 install-nginx-playbook.yml
-rw----- 1 jegan jegan 419 Jul 18 12:56 local-machine-credentials.yml
-rw-rw-r-- 1 jegan jegan 168 Jul 18 11:07 multiple-play-in-a-playbook.yml
-rw-rw-r-- 1 jegan jegan 173 Jul 18 10:49 ping-playbook.yml
-rw-rw-r-- 1 jegan jegan 575 Jul 18 13:09 UbuntuDockerfile
drwxrwxr-x 2 jegan jegan 4096 Jul 18 15:03 windows
jegan@tektutor.org:~/devops-july-2024/Day4/ansible$
```

Lab - Passing arguments to ansible-playbook from command line

```
ansible-playbook passing-arguments-to-playbook-from-cli-playbook.yml -e
greeting_msg=Hello

export GREETING_MSG="Hello!"
ansible-playbook passing-arguments-to-playbook-from-cli-playbook.yml -e
greeting_msg=$GREETING_MSG
```

Expected output

```
Thu Jul 18 3:22:16 PM 🍀 jegan@tektutor.org:~/devops-july-2024/Day4/ansible$ ls
ansible.cfg download-jar-from-jfrog-artifactory-playbook.yml local-machine-credentials.yml windows
authorized_keys fe-1.0.0.jar multiple-play-in-a-playbook.yml
build-docker-image-playbook.yml hosts ping-playbook.yml
CentOSDockerfile install-nginx-playbook.yml UbuntuDockerfile
jegan@tektutor.org:~/devops-july-2024/Day4/ansible$ vim passing-arguments-to-playbook-from-cli-playbook.yml
jegan@tektutor.org:~/devops-july-2024/Day4/ansible$ ansible-playbook passing-arguments-to-playbook-from-cli-playbook.yml -e greeting_m
sg=Hello

PLAY [Demonstrates using command line extra arguments in playbook] ****
TASK [Gathering Facts] ****
ok: [localhost]

TASK [debug] ****
ok: [localhost] => {
    "greeting_msg": "Hello"
}

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

jegan@tektutor.org:~/devops-july-2024/Day4/ansible$ cat passing-arguments-to-playbook-from-cli-playbook.yml
- name: Demonstrates using command line extra arguments in playbook
  hosts: localhost
  tasks:
    - debug: var=greeting_msg
jegan@tektutor.org:~/devops-july-2024/Day4/ansible$ █
```

```
Thu Jul 18 3:24:02 PM 🍀 jegan@tektutor.org:~/devops-july-2024/Day4/ansible$ ls
ansible.cfg download-jar-from-jfrog-artifactory-playbook.yml local-machine-credentials.yml windows
authorized_keys fe-1.0.0.jar multiple-play-in-a-playbook.yml
build-docker-image-playbook.yml hosts ping-playbook.yml
CentOSDockerfile install-nginx-playbook.yml UbuntuDockerfile
jegan@tektutor.org:~/devops-july-2024/Day4/ansible$ vim passing-arguments-to-playbook-from-cli-playbook.yml
jegan@tektutor.org:~/devops-july-2024/Day4/ansible$ export GREETING_MSG="Hello!"
jegan@tektutor.org:~/devops-july-2024/Day4/ansible$ ansible-playbook passing-arguments-to-playbook-from-cli-playbook.yml -e greeting_m
sg=$GREETING_MSG

PLAY [Demonstrates using command line extra arguments in playbook] ****
TASK [Gathering Facts] ****
ok: [localhost]

TASK [debug] ****
ok: [localhost] => {
    "greeting_msg": "Hello!"
}

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

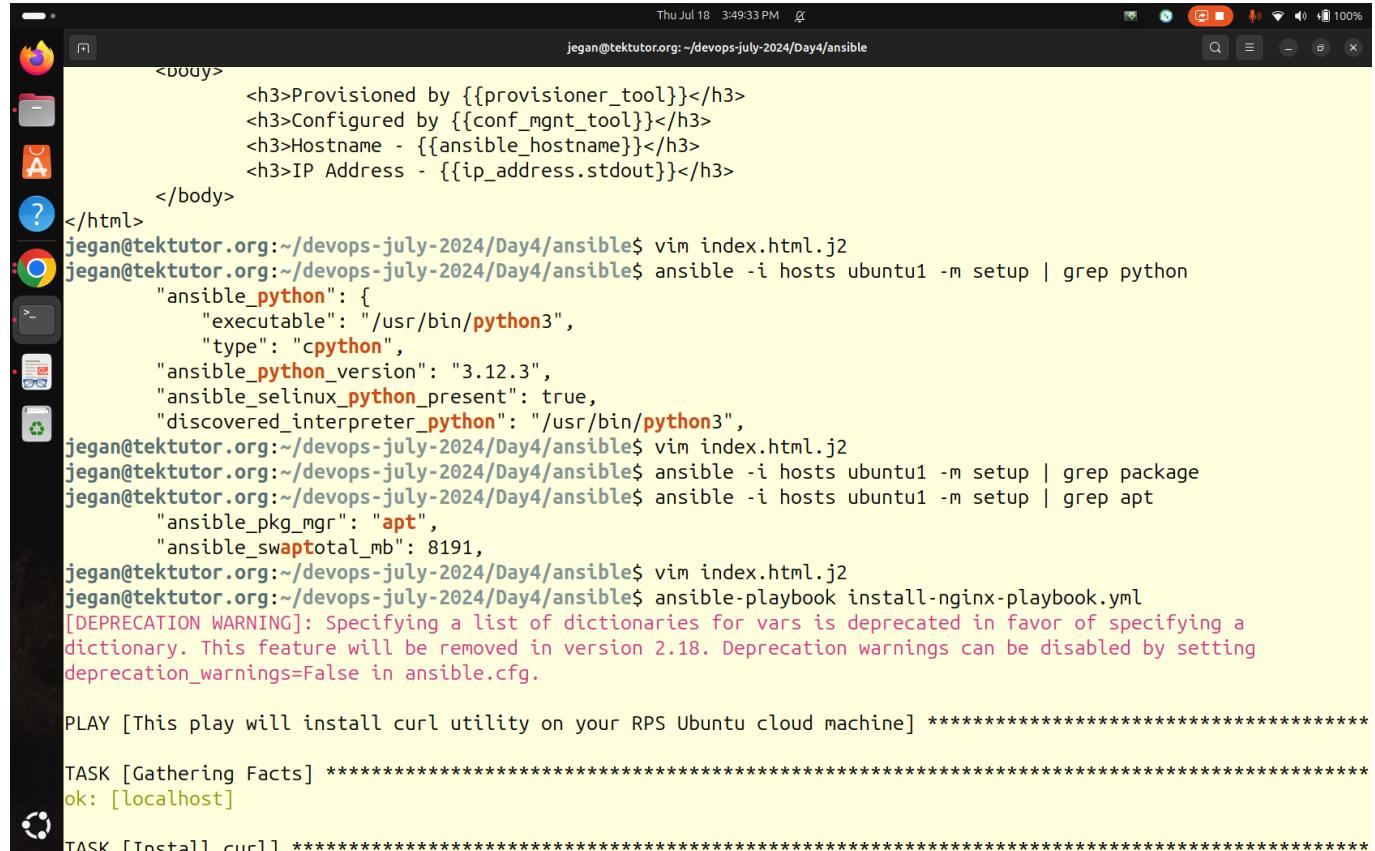
jegan@tektutor.org:~/devops-july-2024/Day4/ansible$ cat passing-arguments-to-playbook-from-cli-playbook.yml
- name: Demonstrates using command line extra arguments in playbook
  hosts: localhost
  tasks:
    - debug: var=greeting_msg
jegan@tektutor.org:~/devops-july-2024/Day4/ansible$ █
```

Lab - Using template module in playbooks

Create a file /home/rps/.my-vault-password with gedit and type 'root' without quotes and save the file.

```
cd ~/devops-july-2024
git pull
cd Day4/ansible
ansible-playbook install-nginx-playbook.yml --ask-become-pass
```

Expected output



The screenshot shows a terminal window on a Linux desktop environment. The title bar indicates the session is running on a host named 'jegan@tektutor.org' with the command 'vim index.html.j2'. The terminal content displays the following sequence of commands and their outputs:

- Initial configuration output from 'index.html.j2':

```
<body>
    <h3>Provisioned by {{provisioner_tool}}</h3>
    <h3>Configured by {{conf_mgnt_tool}}</h3>
    <h3>Hostname - {{ansible_hostname}}</h3>
    <h3>IP Address - {{ip_address.stdout}}</h3>
</body>
</html>
```
- Execution of 'ansible -i hosts ubuntu1 -m setup | grep python':

```
jegan@tektutor.org:~/devops-july-2024/Day4/ansible$ vim index.html.j2
jegan@tektutor.org:~/devops-july-2024/Day4/ansible$ ansible -i hosts ubuntu1 -m setup | grep python
  "ansible_python": {
      "executable": "/usr/bin/python3",
      "type": "cpython",
      "ansible_python_version": "3.12.3",
      "ansible_selinux_python_present": true,
      "discovered_interpreter_python": "/usr/bin/python3",
```
- Execution of 'ansible -i hosts ubuntu1 -m setup | grep package':

```
jegan@tektutor.org:~/devops-july-2024/Day4/ansible$ vim index.html.j2
jegan@tektutor.org:~/devops-july-2024/Day4/ansible$ ansible -i hosts ubuntu1 -m setup | grep package
jegan@tektutor.org:~/devops-july-2024/Day4/ansible$ ansible -i hosts ubuntu1 -m setup | grep apt
  "ansible_pkg_mgr": "apt",
  "ansible_swaptotal_mb": 8191,
```
- Execution of 'ansible-playbook install-nginx-playbook.yml':

```
jegan@tektutor.org:~/devops-july-2024/Day4/ansible$ vim index.html.j2
jegan@tektutor.org:~/devops-july-2024/Day4/ansible$ ansible-playbook install-nginx-playbook.yml
[DEPRECATION WARNING]: Specifying a list of dictionaries for vars is deprecated in favor of specifying a dictionary. This feature will be removed in version 2.18. Deprecation warnings can be disabled by setting deprecation_warnings=False in ansible.cfg.

PLAY [This play will install curl utility on your RPS Ubuntu cloud machine] ****
TASK [Gathering Facts] ****
ok: [localhost]
TASK [Install curl1] ****
```

```
Thu Jul 18 3:47:31 PM ✓
jegan@tektutor.org:~/devops-july-2024/Day4/ansible$ ansible-playbook install-nginx-playbook.yml
[DEPRECATION WARNING]: Specifying a list of dictionaries for vars is deprecated in favor of specifying a
dictionary. This feature will be removed in version 2.18. Deprecation warnings can be disabled by setting
deprecation_warnings=False in ansible.cfg.

PLAY [This play will install curl utility on your RPS Ubuntu cloud machine] ****
TASK [Gathering Facts] ****
ok: [localhost]

TASK [Install curl] ****
ok: [localhost]

PLAY [This playbook will install nginx, configure web root folder and deploys custom web page] ****
TASK [Gathering Facts] ****
ok: [ubuntu1]
ok: [ubuntu2]

TASK [Install nginx in Ubuntu nodes] ****
ok: [ubuntu1]
ok: [ubuntu2]

TASK [Install curl utility in Ubuntu nodes] ****
ok: [ubuntu1]
ok: [ubuntu2]

TASK [Start the nginx service] ****
ok: [ubuntu1]
ok: [ubuntu2]
```

```
Thu Jul 18 3:48:54 PM ✓
jegan@tektutor.org:~/devops-july-2024/Day4/ansible

}
ok: [ubuntu2] => {
    "ip_address": {
        "changed": true,
        "cmd": "hostname -i",
        "delta": "0:00:00.002341",
        "end": "2024-07-18 10:17:16.118189",
        "failed": false,
        "msg": "",
        "rc": 0,
        "start": "2024-07-18 10:17:16.115848",
        "stderr": "",
        "stderr_lines": [],
        "stdout": "172.17.0.4",
        "stdout_lines": [
            "172.17.0.4"
        ]
    }
}

TASK [Deploy custom web page] ****
changed: [ubuntu1]
changed: [ubuntu2]

PLAY [Let's test if nginx is up and running on the ansible nodes] ****
TASK [Gathering Facts] ****
ok: [localhost]

TASK [Check if we are able to access web page from ubuntu1 ansible node] ****
changed: [localhost]
```

```
Thu Jul 18 3:49:02 PM ⓘ
jegan@tektutor.org:~/devops-july-2024/Day4/ansible
```

TASK [Check if we are able to access web page from ubuntu1 ansible node] ****
changed: [localhost]

TASK [debug] ****
ok: [localhost] => {
 "output": {
 "changed": true,
 "cmd": "curl localhost:8001",
 "delta": "0:00:00.006723",
 "end": "2024-07-18 15:47:17.190101",
 "failed": false,
 "msg": "",
 "rc": 0,
 "start": "2024-07-18 15:47:17.183378",
 "stderr": " % Total % Received % Xferd Average Speed Time Time Time Current\n Dload Upload Total Spent Left Speed\n\r0 0 277 100 277 0 0 414k 0 --:-- --:-- --:-- 270k",
 "stderr_lines": [
 " % Total % Received % Xferd Average Speed Time Time Time Current",
 " Dload Upload Total Spent Left Speed",
 "",
 " 0 0 0 0 0 0 0 --:-- --:-- --:-- 0",
 "100 277 100 277 0 0 414k 0 --:-- --:-- --:-- 270k"
],
 "stdout": "<html>\n<head>\n<title>Welcome to DevOps!</title>\n</head>\n<body>\n<h3>Provisioned by Docker</h3>\n<h3>Configured by Ansible</h3>\n<h3>Hostname - ubuntu1</h3>\n<h3>IP Address - 172.17.0.3</h3>\n<h3>Python v- 3.12.3</h3>\n<h3>Package Manager - apt</h3>\n</body>\n</html>",
 "stdout_lines": [
 "<html>",
 "<head>",
 "<title>Welcome to DevOps!</title>",
 "</head>",
 "<body>",
 "<h3>Provisioned by Docker</h3>",
 "<h3>Configured by Ansible</h3>",
 "<h3>Hostname - ubuntu1</h3>",
 "<h3>IP Address - 172.17.0.3</h3>",
 "<h3>Python v- 3.12.3</h3>",
 "<h3>Package Manager - apt</h3>",
 "</body>",
 "</html>"
]
 }
}

TASK [Check if we are able to access web page from ubuntu2 ansible node] ****
changed: [localhost]

TASK [debug] ****
ok: [localhost] => {
 "output": {
 "changed": true,
 "cmd": "curl localhost:8002",
 "delta": "0:00:00.007171",
 "end": "2024-07-18 15:47:17.320078",
 "rc": 0
 }

```

Thu Jul 18 3:49:17 PM
jegan@tektutor.org:~/devops-july-2024/Day4/ansible

TASK [Check if we are able to access web page from ubuntu2 ansible node] *****
changed: [localhost]

TASK [debug] *****
ok: [localhost] => {
    "output": {
        "changed": true,
        "cmd": "curl localhost:8002",
        "delta": "0:00:00.007171",
        "end": "2024-07-18 15:47:17.320078",
        "failed": false,
        "msg": "",
        "rc": 0,
        "start": "2024-07-18 15:47:17.312907",
        "stderr": " % Total % Received % Xferd Average Speed Time Time Time Current\n          Dload Upload Total Spent Left Speed\n\r100 277 100 277 0 0 362k 0 --:-- --:-- --:-- 270k",
        "stderr_lines": [
            " % Total % Received % Xferd Average Speed Time Time Time Current",
            "          Dload Upload Total Spent Left Speed",
            "",
            " 0 0 0 0 0 0 0 --:-- --:-- --:-- 0",
            "100 277 100 277 0 0 362k 0 --:-- --:-- --:-- 270k"
        ],
        "stdout": "<html>\n<head>\n<title>Welcome to DevOps!</title>\n</head>\n<body>\n<h3>Provisioned by Docker</h3>\n<h3>Configured by Ansible</h3>\n<h3>Hostname - ubuntu2</h3>\n<h3>IP Address - 172.17.0.4</h3>\n<h3>Python v- 3.12.3</h3>\n<h3>Package Manager - apt</h3>\n</body>\n</html>",
        "stdout_lines": [
            "<html>",
            "<head>",
            "<title>Welcome to DevOps!</title>",
            "</head>",
            "<body>",
            "<h3>Provisioned by Docker</h3>",
            "<h3>Configured by Ansible</h3>",
            "<h3>Hostname - ubuntu2</h3>",
            "<h3>IP Address - 172.17.0.4</h3>",
            "<h3>Python v- 3.12.3</h3>",
            "<h3>Package Manager - apt</h3>",
            "</body>",
            "</html>"
        ]
    }
}

PLAY RECAP *****
localhost : ok=7 changed=2 unreachable=0 failed=0 skipped=0 rescued=0 ignored=0
ubuntu1 : ok=10 changed=3 unreachable=0 failed=0 skipped=0 rescued=0 ignored=0
ubuntu2 : ok=10 changed=3 unreachable=0 failed=0 skipped=0 rescued=0 ignored=0

jegan@tektutor.org:~/devops-july-2024/Day4/ansible$ 

```

Info - SCRUM - Daily stand-up meeting

- is an inspect and adapt meeting
- in other words, it is fail-fast meeting
- the team inspects the yesterday's plan against the real status as on yesterday

- if the team finds a deviation, if required the plan must be updated

Info - DevOps

- Developers
 - they need to automate unit/integrating testing using Test Frameworks
 - developers are expected to learn little bit of QA skills
 - Test Frameworks
 - Developers should follow TDD(Test Driven Development)
 - Java
 - JUnit/TestNg/Mockito/PowerMock/EasyMock/JMock, etc
 - C/C++
 - CUnit, CppUnit, Google Test/Mock
 - JavaScript (Angular, NodeJS)
 - Jasmine, Karma
 - Python
 - PyTest
 - C#
 - NUnit/MSTest/Moq
- QA
 - they need to automate e2e functionality test, API Test, Stress/Load Test, Regression Test, Sanity Check, Smoke Test
 - the test automation expects to convert the testing effort into source code that can be pushed to Version control
 - QA folks are expected to learn little of development skills
 - Frameworks
 - Behaviour Driven Development Frameworks (BDD)
 - Cucumber, Specflow, Selenium, Raft
- System Administrators
 - they need to automate OS installation, provisioning, software installations, user management, etc
 - Provisioning
 - System Adminstrators are expected to use Infrastructure as a Code Tools like Cloudformation, Terraform, Docker, Vagrant, etc.,
 - System Administrators are expected to use Configuration Management Tools like Ansible, Puppet/Chef, etc to install softwares on an existing Virtual Machine or a OS on OnPrem environment(Private cloud), public cloud and Hybrid Cloud
 - System Administrators also has to know coding

Info - Continuous Integration (CI)

- the code developed by each developer is continuously integrated in the dev branch
- Jenkins or CI Build Server grabs the latest code and it is going run the build along with unit and integration test cases
- If any test case fails, the build will fail, all the test cases are passing then the build will pass

Info - Continuous Deployment (CD)

- the CI certified application binaries are automatically deployed into QA environment for further QA automated test execution

Info - Continuous Delivery (CD)

- the QA certified release(application binaries) are deployed into customer's environment which is close to production environment (pre-prod) environment
- the Customer will verify the functionality and if the customer found the application is working as expected and found to be stable, they can decide if they can make it live in production environment

Info - Jenkins Overview

- Jenkins is a Continuous Integration Build Server
- it was developed in Java by Kohsuke Kawaguchi while he was working for Sun Microsystems
- the initial product developed was called Hudson
- it is an opensource tool
- Before leaving Oracle, Kohsuke Kawaguchi created a branch of Hudson source code called Jenkins
- Kohsuke founded a company called Cloudbees
- Today Hudson and Jenkins shares lot of common code, but Jenkins has become very popular than Hudson
- Jenkins has over 10000+ active opensource contributors
- there are 2 variants of Jenkins
 1. Jenkins open-source
 2. Cloudbees (Enterprise edition of Jenkins)
- it supports creating CI build for any programming language

Info - Jenkins alternates

- Team City
- Bamboo
- Microsoft's Team Foundation Server (TFS)
- CircleCI

Info - Just in case you need to reset Jenkins Administrator password

<https://www.jenkins.io/doc/book/system-administration/admin-password-reset-instructions/>

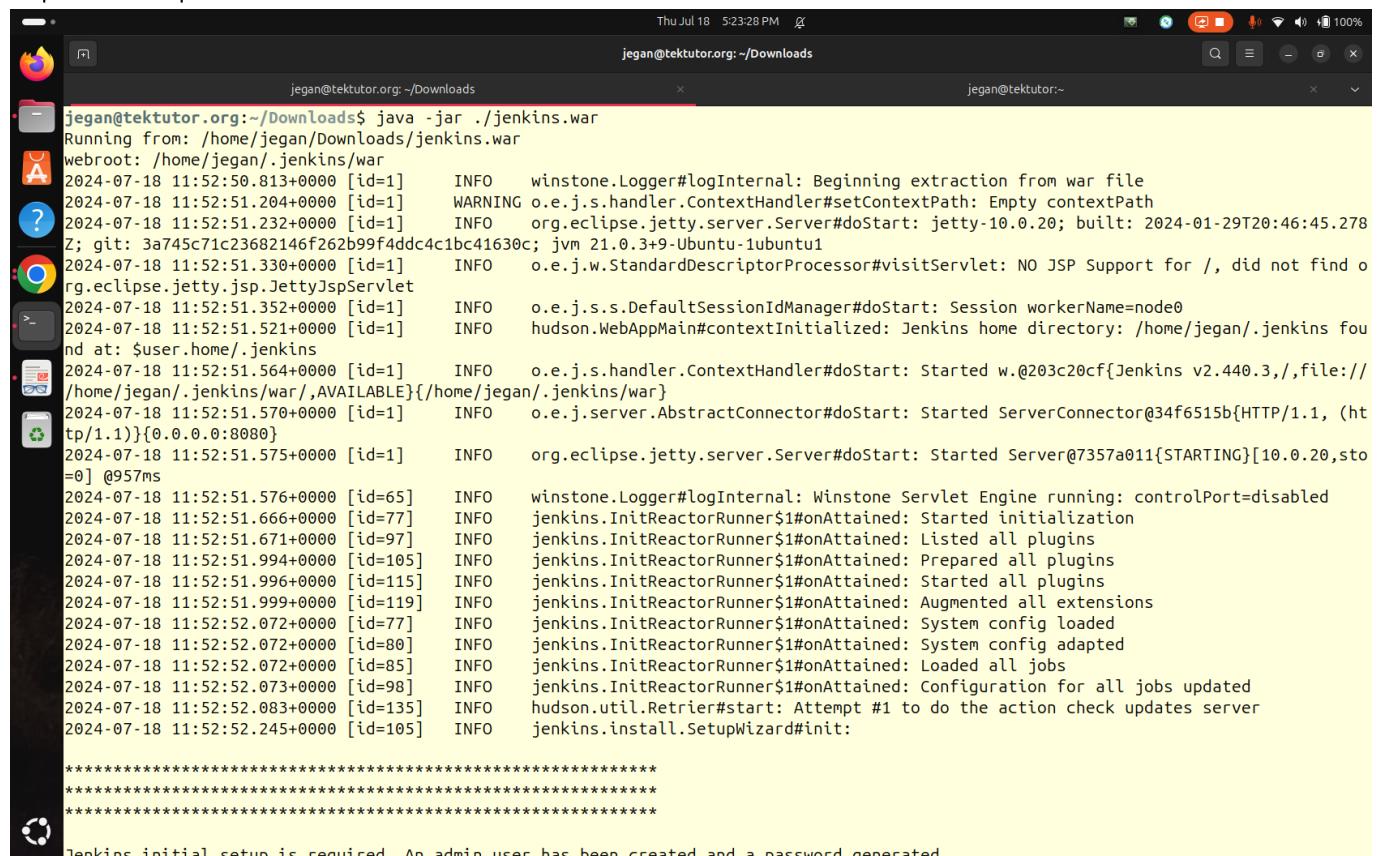
Lab - Launching Jenkins from terminal

Verify if already any application is using the port 8080

The below command is a blocking command, hence don't press Ctrl+C, you may open new terminal tab to run other commands

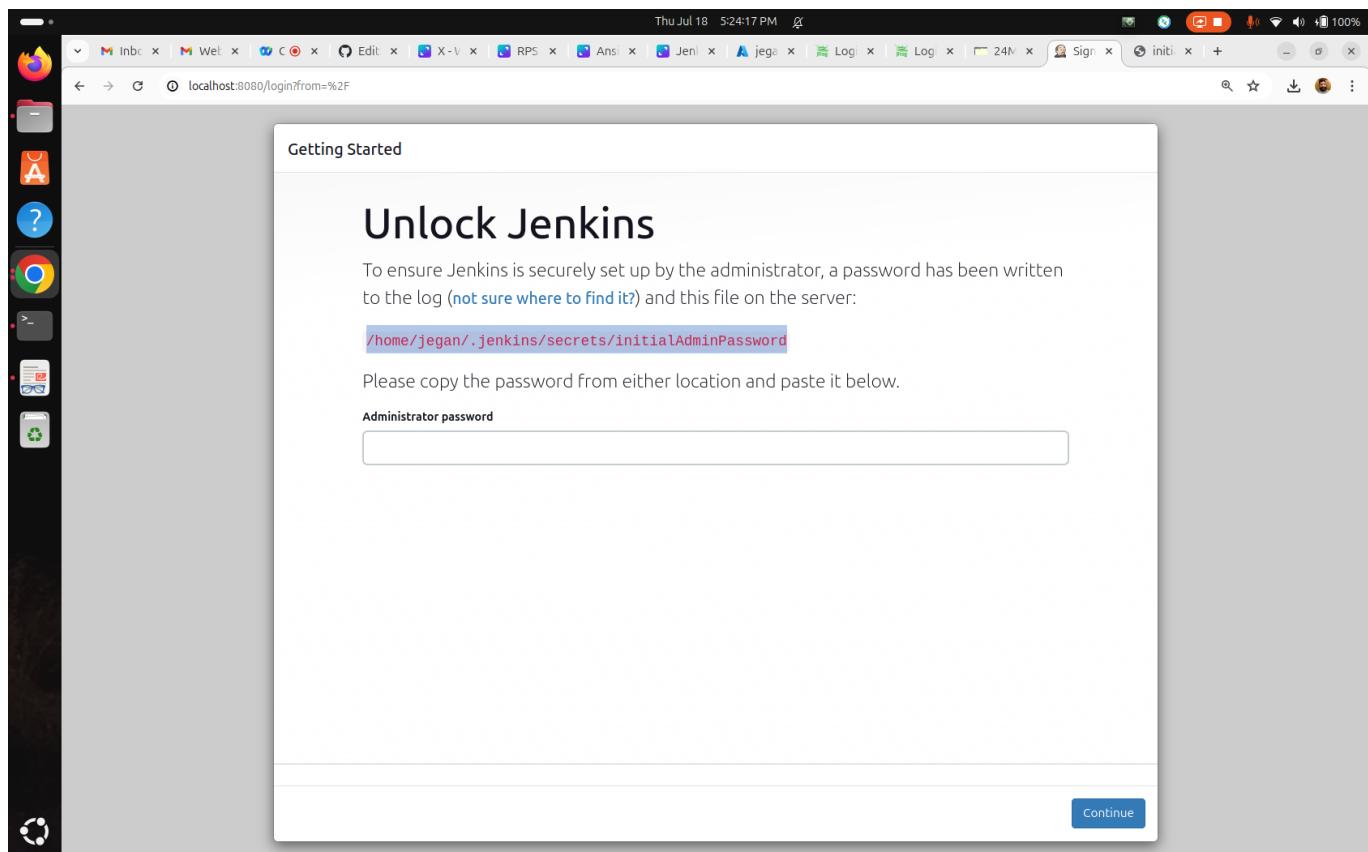
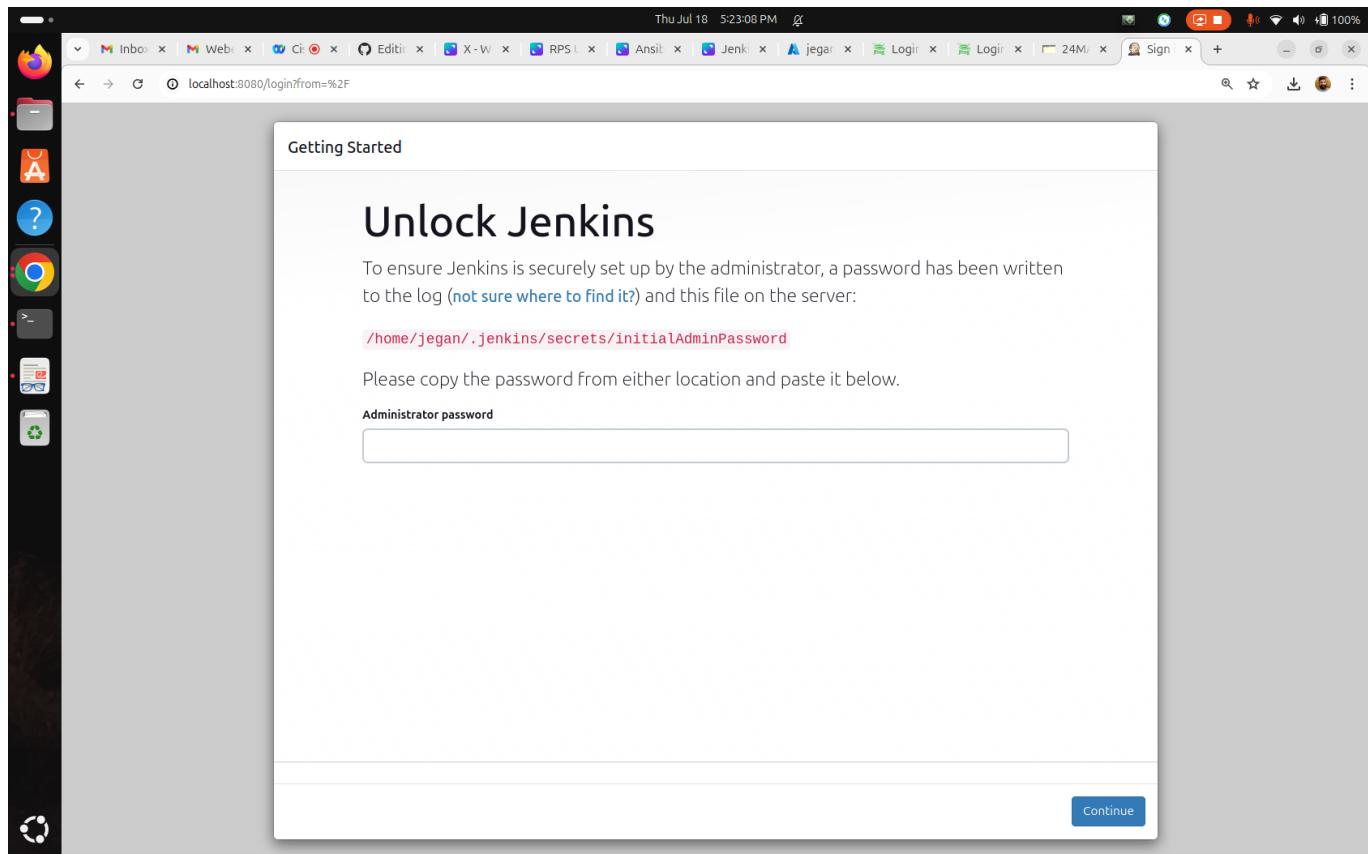
```
cd ~/Downloads  
java -jar ./jenkins.jar
```

Expected output



A screenshot of a terminal window titled "jegan@tektutor.org: ~/Downloads". The window shows the command "java -jar ./jenkins.jar" being run. The terminal output is a log of Jenkins's startup process, including the extraction of the war file, configuration of the Winstone servlet engine, and the loading of various plugins and jobs. The log ends with a message about initial setup and password generation.

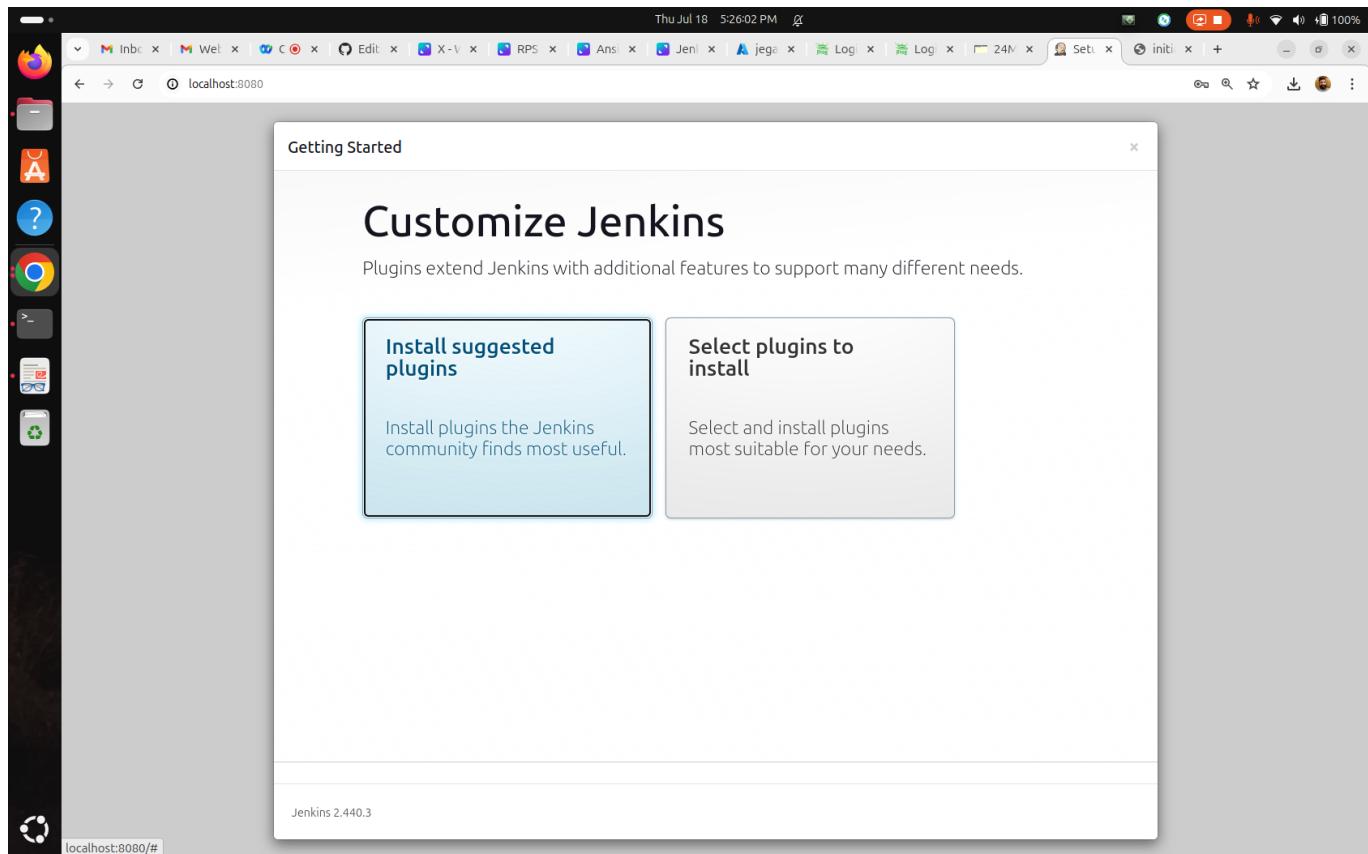
```
jegan@tektutor.org:~/Downloads$ java -jar ./jenkins.war  
Running from: /home/jegan/Downloads/jenkins.war  
webroot: /home/jegan/.jenkins/war  
2024-07-18 11:52:50.813+0000 [id=1] INFO winstone.Logger#logInternal: Beginning extraction from war file  
2024-07-18 11:52:51.204+0000 [id=1] WARNING o.e.j.s.handler.ContextHandler#setContextPath: Empty contextPath  
2024-07-18 11:52:51.232+0000 [id=1] INFO org.eclipse.jetty.server.Server#doStart: jetty-10.0.20; built: 2024-01-29T20:46:45.278Z; git: 3a745c71c23682146f262b99f4ddc4c1bc41630c; jvm 21.0.3+9-Ubuntu-1ubuntu1  
2024-07-18 11:52:51.330+0000 [id=1] INFO o.e.j.w.StandardDescriptorProcessor#visitServlet: NO JSP Support for /, did not find org.eclipse.jetty.jsp.JettyJspServlet  
2024-07-18 11:52:51.352+0000 [id=1] INFO o.e.j.s.s.DefaultSessionIdManager#doStart: Session workerName=node0  
2024-07-18 11:52:51.521+0000 [id=1] INFO hudson.WebAppMain#contextInitialized: Jenkins home directory: /home/jegan/.jenkins found at: $user.home/.jenkins  
2024-07-18 11:52:51.564+0000 [id=1] INFO o.e.j.s.handler.ContextHandler#doStart: Started w.@0203c20cf{Jenkins v2.440.3,/file:///home/jegan/.jenkins/war/,AVAILABLE}{/home/jegan/.jenkins/war}  
2024-07-18 11:52:51.570+0000 [id=1] INFO o.e.j.server.AbstractConnector#doStart: Started ServerConnector@34f6515b{HTTP/1.1, (http/1.1)}{0.0.0.0:8080}  
2024-07-18 11:52:51.575+0000 [id=1] INFO org.eclipse.jetty.server.Server#doStart: Started Server@7357a011{STARTING}[10.0.20,sto=0] @957ms  
2024-07-18 11:52:51.576+0000 [id=65] INFO winstone.Logger#logInternal: Winstone Servlet Engine running: controlPort=disabled  
2024-07-18 11:52:51.666+0000 [id=77] INFO jenkins.InitReactorRunner$1#onAttained: Started initialization  
2024-07-18 11:52:51.671+0000 [id=97] INFO jenkins.InitReactorRunner$1#onAttained: Listed all plugins  
2024-07-18 11:52:51.994+0000 [id=105] INFO jenkins.InitReactorRunner$1#onAttained: Prepared all plugins  
2024-07-18 11:52:51.996+0000 [id=115] INFO jenkins.InitReactorRunner$1#onAttained: Started all plugins  
2024-07-18 11:52:51.999+0000 [id=119] INFO jenkins.InitReactorRunner$1#onAttained: Augmented all extensions  
2024-07-18 11:52:52.072+0000 [id=77] INFO jenkins.InitReactorRunner$1#onAttained: System config loaded  
2024-07-18 11:52:52.072+0000 [id=80] INFO jenkins.InitReactorRunner$1#onAttained: System config adapted  
2024-07-18 11:52:52.072+0000 [id=85] INFO jenkins.InitReactorRunner$1#onAttained: Loaded all jobs  
2024-07-18 11:52:52.073+0000 [id=98] INFO jenkins.InitReactorRunner$1#onAttained: Configuration for all jobs updated  
2024-07-18 11:52:52.083+0000 [id=135] INFO hudson.util.Retriger#start: Attempt #1 to do the action check updates server  
2024-07-18 11:52:52.245+0000 [id=105] INFO jenkins.install.SetupWizard#init:  
  
*****  
*****  
*****  
Jenkins initial setup is required. An admin user has been created and a password generated.
```



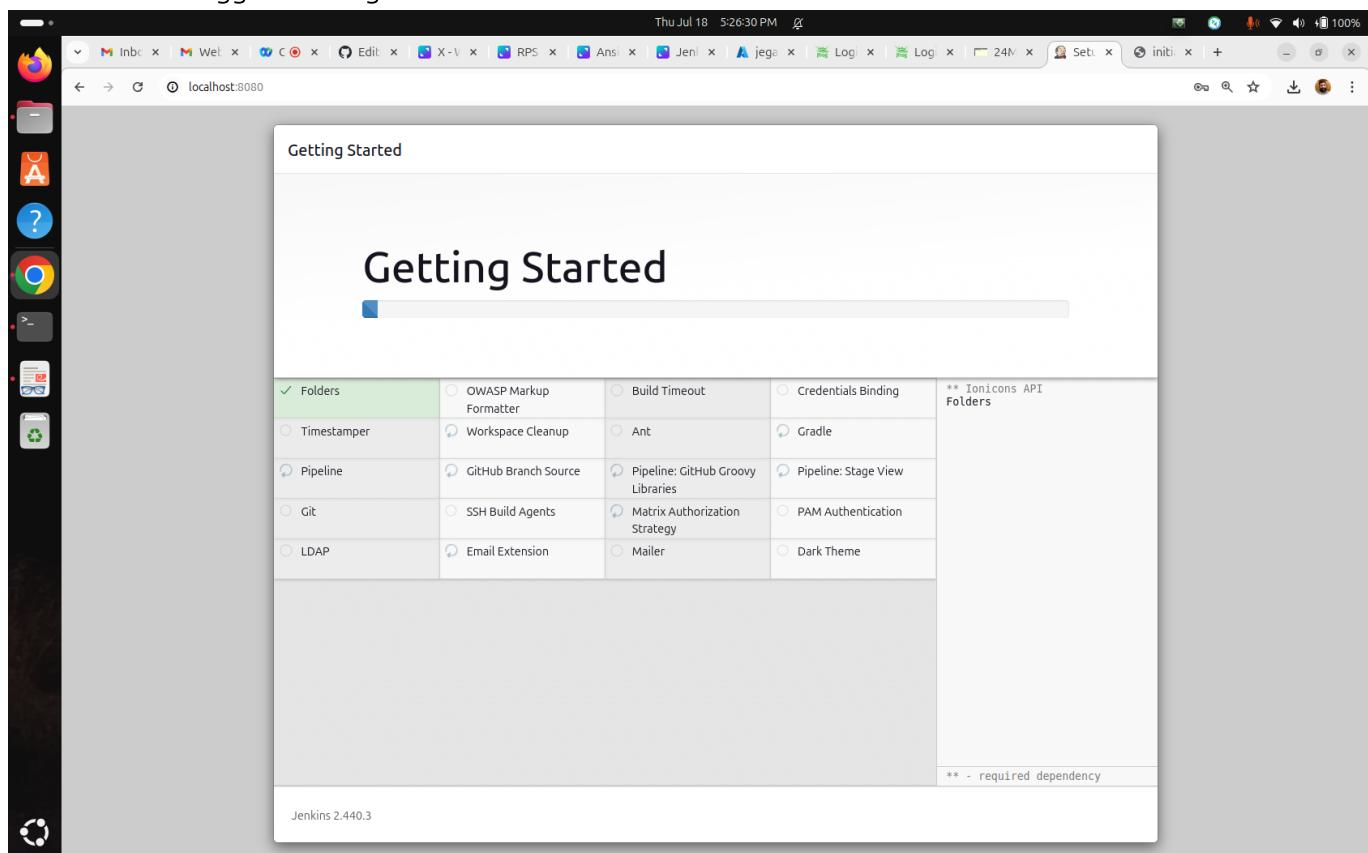
The screenshot shows a Linux desktop environment with a dark theme. On the left is a vertical dock containing icons for various applications like a terminal, file manager, and system tools. Two terminal windows are open at the top:

- The first terminal window shows the command `File /home/jegan/jenkins/secrets/initialAdminPassword` and its output: `2192cb7f38594de28a31abbcfc294ed4`.
- The second terminal window shows the command `localhost:8080/login?from=%2F`.

Below the terminals is a browser window displaying the Jenkins "Getting Started" page. The title is "Unlock Jenkins". It instructs the user to copy the password from either the log or this file. The password is listed as `/home/jegan/.jenkins/secrets/initialAdminPassword`. A text input field labeled "Administrator password" contains the password value. A "Continue" button is at the bottom right of the dialog.



Select "Install Suggested Plugins"



The screenshot shows the Jenkins 'Getting Started' page. A table lists various Jenkins plugins with checkboxes indicating they are installed. The table has four columns: Folders, OWASP Markup Formatter, Build Timeout, and Credentials Binding. The Pipeline column contains several sub-plugins with checkboxes: GitHub Branch Source, Pipeline: GitHub Groovy Libraries, Pipeline: Stage View, Matrix Authorization Strategy, PAM Authentication, and Mailer. A tooltip for the Mailer plugin lists numerous Jenkins API endpoints. The Jenkins version is 2.440.3.

✓ Folders	✓ OWASP Markup Formatter	✓ Build Timeout	✓ Credentials Binding
✓ Timestamper	✓ Workspace Cleanup	✓ Ant	✓ Gradle
✓ Pipeline	GitHub Branch Source	Pipeline: GitHub Groovy Libraries	Pipeline: Stage View
Git	SSH Build Agents	Matrix Authorization Strategy	PAM Authentication
LDAP	Email Extension	Mailer	Dark Theme

** Instance Identity
** Pipeline: SCM Step
** Pipeline: Groovy
** Pipeline: Job
** Jakarta Activation API
** Jakarta Mail API
** Jakarta HttpComponents Client 4.x API
** Pipeline: Basic Steps
Gradle
** Pipeline: Milestone Step
** Pipeline: Build Step
** Pipeline: Groovy Libraries
** Pipeline: Stage Step
** Pipeline: Model API
** Pipeline: Declarative Extension Points API
** Branch API
** Pipeline: Multibranch
** Pipeline: Stage Tags
Metadata
** Pipeline: Input Step
** Pipeline: Declarative Pipeline
** Java JSON Web Token (JJWT)
** OkHttp
** GitHub API
** Mina SSHD API :: Common
** Mina SSHD API :: Core
** - required dependency

Jenkins 2.440.3 rpsconsulting1.webex.com is sharing your screen. Stop sharing Hide

The screenshot shows the 'Create First Admin User' page. It features five input fields: 'Username' (with placeholder '/'), 'Password', 'Confirm password', 'Full name', and 'E-mail address'. At the bottom right are 'Skip and continue as admin' and 'Save and Continue' buttons. The Jenkins version is 2.440.3.

Getting Started

Create First Admin User

Username

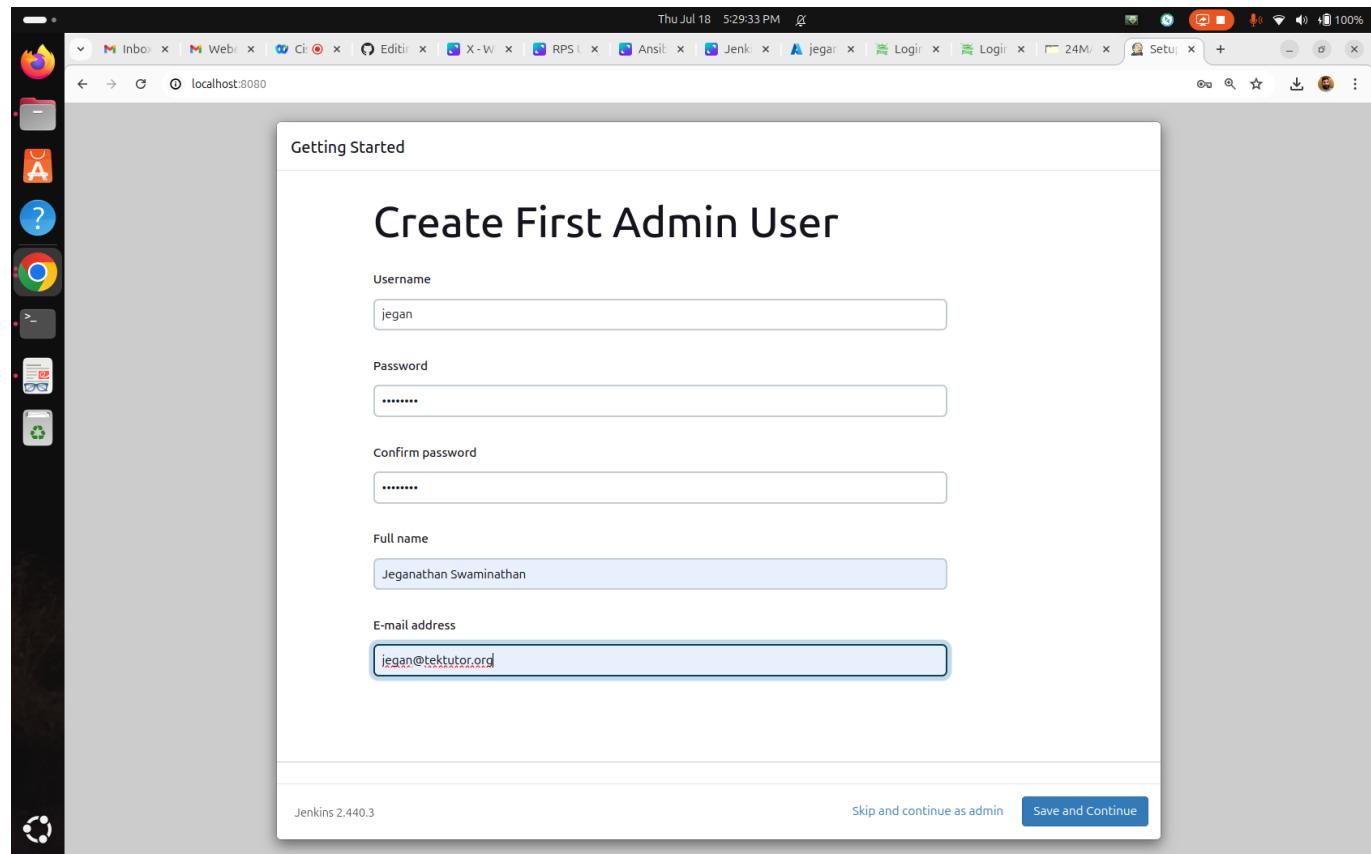
Password

Confirm password

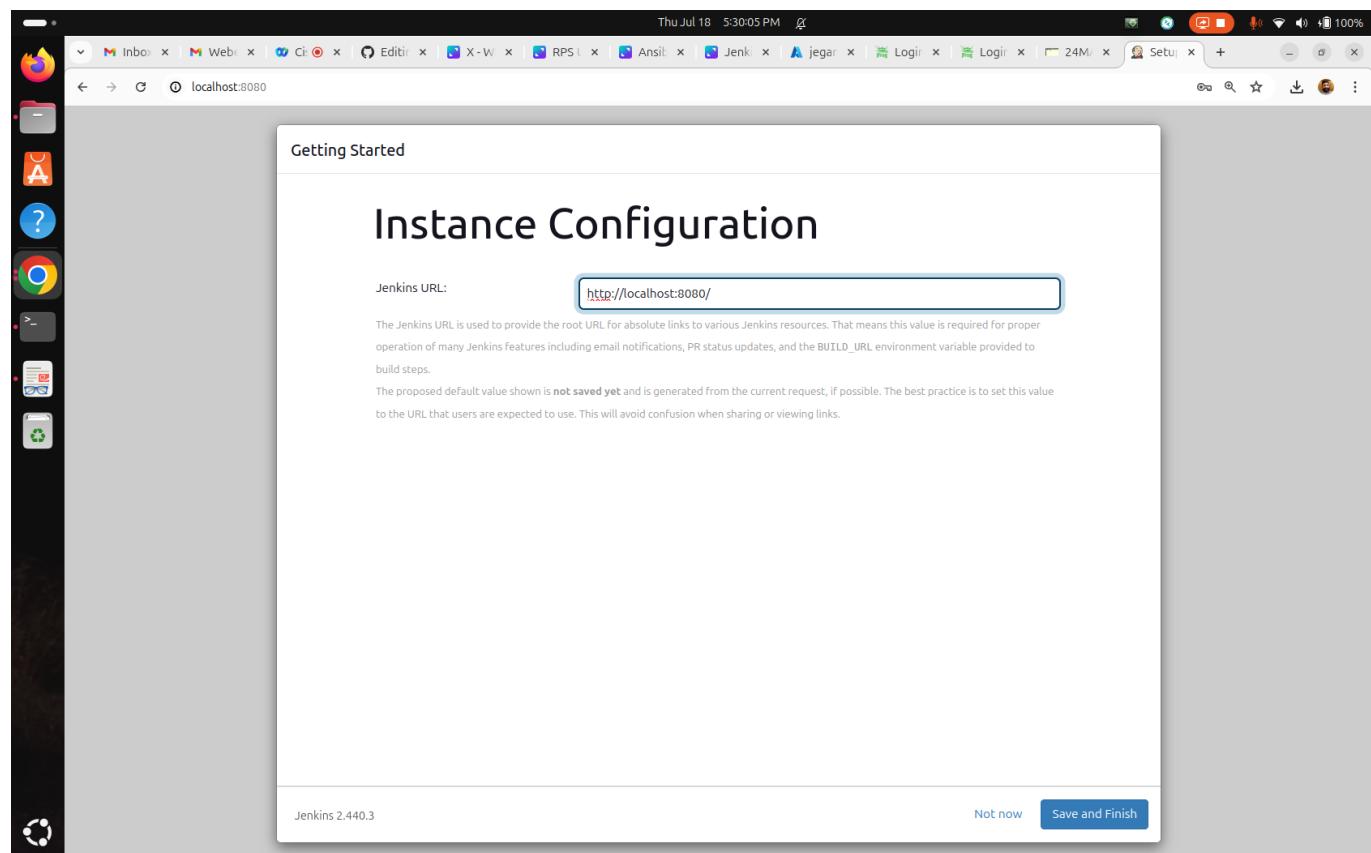
Full name

E-mail address

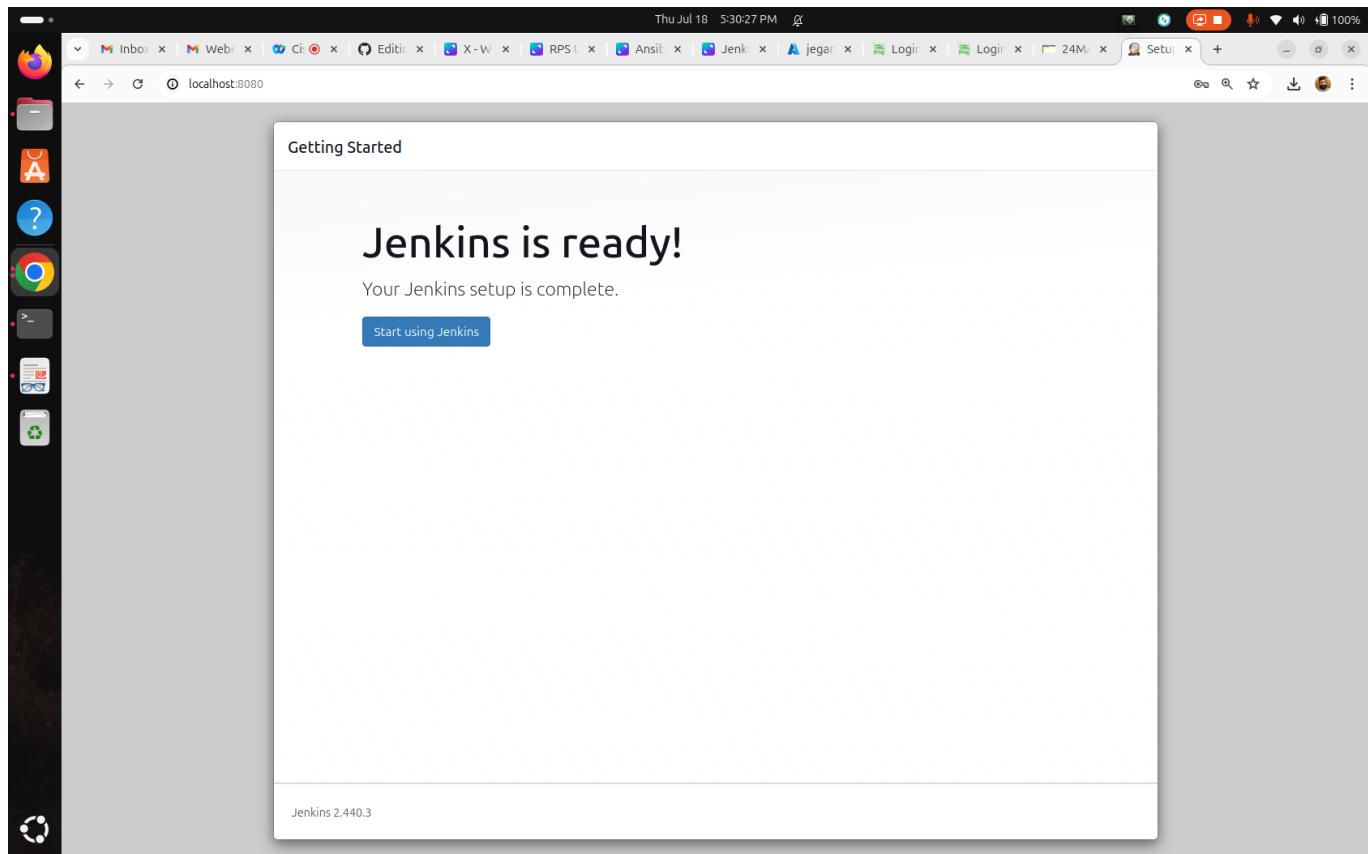
Jenkins 2.440.3 Skip and continue as admin Save and Continue



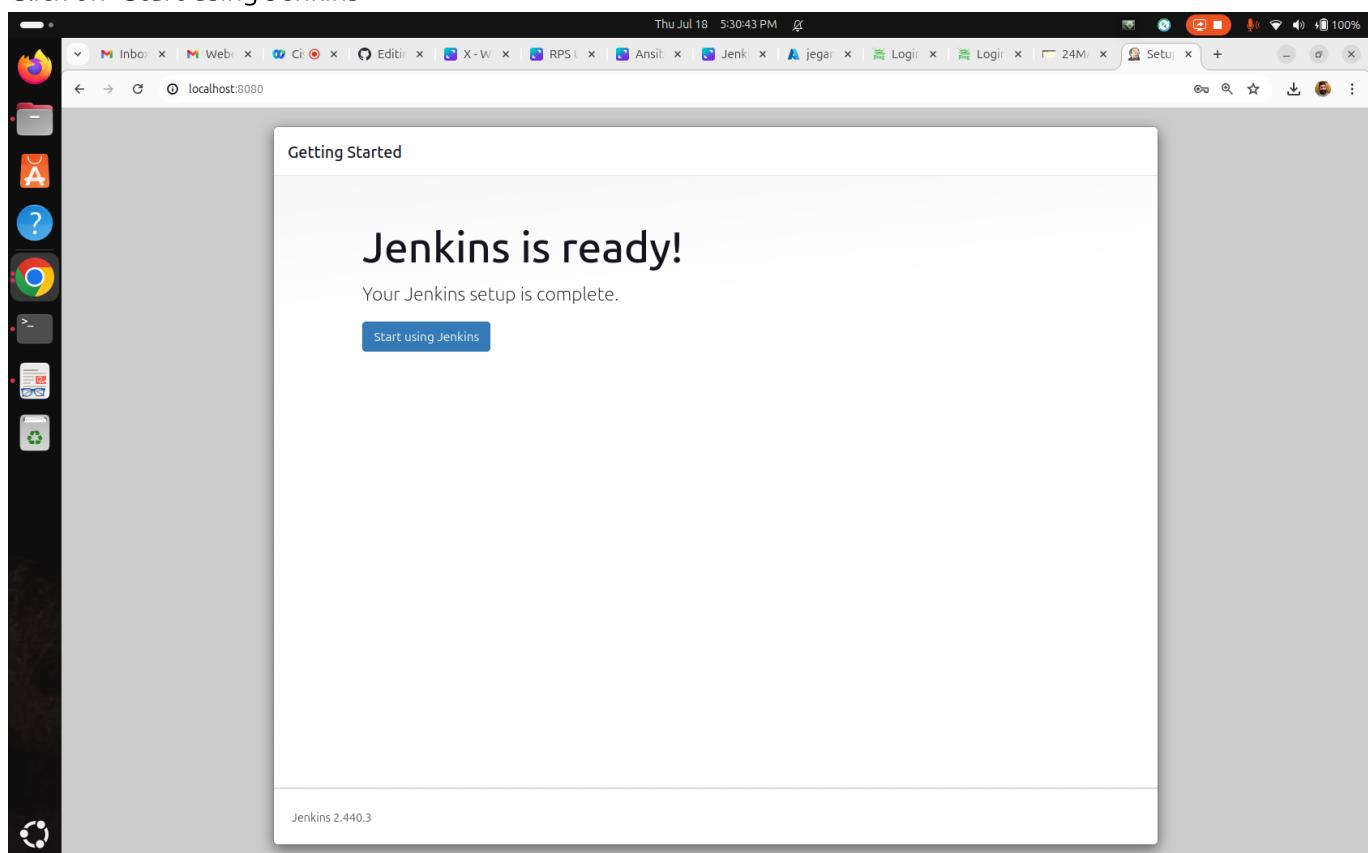
Click on "Save and Continue"



Click on "Save and Finish"



Click on "Start using Jenkins"



The screenshot shows the Jenkins dashboard at localhost:8080. The top navigation bar includes links for various Jenkins instances and system status. The main content area features a "Welcome to Jenkins!" message, a "Start building your software project" call-to-action, and several management options like "Create a job", "Set up a distributed build", "Set up an agent", "Configure a cloud", and "Learn more about distributed builds". On the left, there are sections for "Build Queue" (empty) and "Build Executor Status" (2 Idle). A sidebar on the far left contains icons for various Jenkins features.

Thu Jul 18 5:30:53 PM

Inbox x Web x C: x Edit x X-W x RPS x Ansit x Jenk x jegar x Logix x Logix x 24M x Dash x

localhost:8080

Jenkins

Search (CTRL+K)

Jeganathan Swaminathan log out

Dashboard >

+ New Item

People

Build History

Manage Jenkins

My Views

Welcome to Jenkins!

Add description

This page is where your Jenkins jobs will be displayed. To get started, you can set up distributed builds or start building a software project.

Start building your software project

Create a job +

Set up a distributed build

Set up an agent

Configure a cloud

Learn more about distributed builds ?

Build Queue

No builds in the queue.

Build Executor Status

1 Idle

2 Idle

REST API | Jenkins 3.440.3

```
Thu Jul 18 15:32:04 PM AZ
jegan@tektutor.org: ~/Downloads
jegan@tektutor.org: ~/Downloads
jegan@tektutor:~
```

of admin
2024-07-18 11:58:52.388+0000 [id=170] INFO h.m.UpdateCenter\$UpdateCenterConfiguration#download: Downloading email-ext
2024-07-18 11:58:54.005+0000 [id=170] INFO h.model.UpdateCenter\$DownloadJob#run: Installation successful: email-ext
2024-07-18 11:58:54.005+0000 [id=170] INFO h.model.UpdateCenter\$DownloadJob#run: Starting the installation of mailer on behalf of admin
2024-07-18 11:58:54.005+0000 [id=170] INFO h.m.UpdateCenter\$InstallationJob#_run: Skipping duplicate install of: Mailer@472.vf7c289a_4b_420
2024-07-18 11:58:54.005+0000 [id=170] INFO h.model.UpdateCenter\$DownloadJob#run: Installation successful: mailer
2024-07-18 11:58:54.005+0000 [id=170] INFO h.model.UpdateCenter\$DownloadJob#run: Starting the installation of theme-manager on behalf of admin
2024-07-18 11:58:54.005+0000 [id=170] INFO h.m.UpdateCenter\$UpdateCenterConfiguration#download: Downloading theme-manager
2024-07-18 11:58:55.525+0000 [id=170] INFO h.model.UpdateCenter\$DownloadJob#run: Installation successful: theme-manager
2024-07-18 11:58:55.525+0000 [id=170] INFO h.model.UpdateCenter\$DownloadJob#run: Starting the installation of dark-theme on behalf of admin
2024-07-18 11:58:55.525+0000 [id=170] INFO h.m.UpdateCenter\$UpdateCenterConfiguration#download: Downloading dark-theme
2024-07-18 11:58:56.977+0000 [id=170] INFO h.model.UpdateCenter\$DownloadJob#run: Installation successful: dark-theme
2024-07-18 11:58:56.978+0000 [id=170] INFO h.m.UpdateCenter\$CompleteBatchJob#run: Completing installing of plugin batch...
2024-07-18 11:58:57.563+0000 [id=170] INFO h.p.b.g.GlobalTimeOutConfiguration#load: global timeout not set
2024-07-18 11:58:58.528+0000 [id=170] INFO h.m.DownloadService\$Downloadable#load: Obtained the updated data file for hudson.tasks.Ant.AntInstaller
2024-07-18 11:58:59.800+0000 [id=170] INFO h.m.DownloadService\$Downloadable#load: Obtained the updated data file for hudson.plugins.gradle.GradleInstaller
2024-07-18 11:58:59.925+0000 [id=354] INFO jenkins.InitReactorRunner\$1#onAttained: Started initialization
2024-07-18 11:58:59.926+0000 [id=366] INFO jenkins.InitReactorRunner\$1#onAttained: Listed all plugins
2024-07-18 11:58:59.927+0000 [id=372] INFO jenkins.InitReactorRunner\$1#onAttained: Prepared all plugins
2024-07-18 11:58:59.932+0000 [id=382] INFO jenkins.InitReactorRunner\$1#onAttained: Started all plugins
2024-07-18 11:58:59.934+0000 [id=389] INFO jenkins.InitReactorRunner\$1#onAttained: Augmented all extensions
2024-07-18 11:58:59.935+0000 [id=400] INFO jenkins.InitReactorRunner\$1#onAttained: System config loaded
2024-07-18 11:58:59.935+0000 [id=354] INFO jenkins.InitReactorRunner\$1#onAttained: System config adapted
2024-07-18 11:58:59.942+0000 [id=394] INFO jenkins.InitReactorRunner\$1#onAttained: Loaded all jobs
2024-07-18 11:58:59.943+0000 [id=394] INFO jenkins.InitReactorRunner\$1#onAttained: Configuration for all jobs updated
2024-07-18 11:59:00.056+0000 [id=374] INFO jenkins.InitReactorRunner\$1#onAttained: Completed initialization
2024-07-18 11:59:00.057+0000 [id=170] INFO h.m.UpdateCenter\$CompleteBatchJob#run: Completed installation of 89 plugins in 2 min 34 sec

Click on "Manage Jenkins"

The screenshot shows the Jenkins Manage Jenkins interface. On the left, there's a sidebar with links for New Item, People, Build History, Manage Jenkins (which is selected), and My Views. The main area has a header "Manage Jenkins". A message at the top says "New version of Jenkins (2.452.3) is available for download (changelog)." with a "Or Upgrade Automatically" button. Below this, a warning message states "Building on the built-in node can be a security issue. You should set up distributed builds. See [the documentation](#)." with buttons for "Set up agent", "Set up cloud", and "Dismiss". The "System Configuration" section contains icons for System, Tools, Plugins, Nodes, Clouds, and Appearance, each with a brief description. On the far left, there are sections for Build Queue (No builds in the queue) and Build Executor Status (1 Idle, 2 Idle).

Click on "Tools"

The screenshot shows the Jenkins Tools configuration page. The left sidebar shows "Dashboard > Manage Jenkins > Tools". The main content area has a header "Tools". It includes sections for "Maven Configuration" (Default settings provider: Use default maven settings), "JDK installations" (Add JDK button), and "Git installations" (a table with one entry for "Git" with Name "Default"). Buttons for "Save" and "Apply" are at the bottom.

Under "JDK Installation", select "Add JDK"

The screenshot shows the Jenkins 'Tools' configuration page. Under the 'JDK installations' section, there is a form to add a new JDK. The 'Name' field is required and highlighted with a red border. The 'JAVA_HOME' field is also present. There is an option to 'Install automatically'. A 'Save' button is visible at the bottom of the section.

Go to your Linux Terminal and type "mvn --version", copy the Java home and return back to Jenkins browser tab

The screenshot shows a Linux terminal window with three tabs. The active tab shows the command 'mvn --version' being run, and the output is displayed:

```
Apache Maven 3.8.7
Maven home: /usr/share/maven
Java version: 21.0.3, vendor: Ubuntu, runtime: /usr/lib/jvm/java-21-openjdk-amd64
Default locale: en_US, platform encoding: UTF-8
OS name: "linux", version: "6.8.0-38-generic", arch: "amd64", family: "unix"
```

Paste the JDK path you copied from mvn terminal into the `JDK_PATH` in the Jenkins page

The screenshot shows the Jenkins 'Tools' configuration page. At the top, there is a link to 'Use default maven global settings'. Below it, under 'JDK installations', a new entry named 'JAVA21' is being added. The 'Name' field contains 'JAVA21', which is highlighted in red with a validation error message: 'Required'. The 'JAVA_HOME' field is set to '/usr/lib/jvm/java-21-openjdk-amd64'. A checkbox for 'Install automatically' is unchecked. Below this section is another 'Add JDK' button. Under 'Git installations', a new entry named 'Git' is being added. The 'Name' field contains 'Git', and the 'Save' button is highlighted in blue. There is also an 'Apply' button.

Scroll to the bottom and you will see "Maven Installations"

The screenshot shows the Jenkins interface for managing tools. In the top navigation bar, the URL is `localhost:8080/manage/configureTools/`. The left sidebar has icons for various Jenkins features like Pipeline, Ansible, and Jenkins itself. The main content area is titled "Tools". Under "git", there is a search bar with "git" typed in and an unchecked checkbox labeled "Install automatically". Below this is a button "Add Git". There are also sections for "Gradle installations", "Ant installations", and "Maven installations", each with a "Add [Tool]" button. At the bottom are "Save" and "Apply" buttons.

Click on "Add Maven"

This screenshot shows the same Jenkins interface as above, but with a new "Maven" entry being added. The "Name" field is empty and has a red exclamation mark indicating it is required. The "Install automatically" checkbox is checked. Under "Install from Apache", the "Version" dropdown is set to "3.9.8". A "Save" and "Apply" button are at the bottom.

Under Name, you can type MAVEN387 to indicate maven 3.8.7 or give an appropriate string as per your

maven version. Uncheck, "Install automatically"

The screenshot shows the Jenkins 'Tools' configuration page under 'Manage Jenkins'. The 'Maven installations' section is active. A new Maven installation is being configured with the name 'MAVEN387'. The 'Install automatically' checkbox is unchecked. Buttons for 'Save' and 'Apply' are visible at the bottom.

Go to terminal, copy the Maven home

The screenshot shows a terminal window with three tabs. The active tab displays the output of the 'mvn --version' command, which shows Apache Maven 3.8.7 is installed. The Java version is 21.0.3, vendor is Ubuntu, runtime is /usr/lib/jvm/java-21-openjdk-amd64, and the default locale is en_US.

```
jegan@tektutor.org ~$ mvn --version
Apache Maven 3.8.7
Maven home: /usr/share/maven
Java version: 21.0.3, vendor: Ubuntu, runtime: /usr/lib/jvm/java-21-openjdk-amd64
Default locale: en_US, platform encoding: UTF-8
OS name: "linux", version: "6.8.0-38-generic", arch: "amd64", family: "unix"
```

Return back to your Jenkins configuration page and paste the Maven Path as shown below

The screenshot shows the Jenkins 'Configure Tools' page. Under the 'Maven installations' section, a new Maven configuration is being added. The 'Name' field contains 'MAVEN387'. The 'MAVEN_HOME' field contains '/usr/share/maven'. A checkbox for 'Install automatically' is unchecked. At the bottom, there are 'Save' and 'Apply' buttons, and a status message 'Jenkins 2.440.3'.

Click "Save" button

The screenshot shows the Jenkins 'Manage Jenkins' page. The left sidebar has 'Manage Jenkins' selected. The main area displays system configuration options like System, Tools, Nodes, Clouds, Plugins, Appearance, Security, Credentials, and Credential Providers. A message at the top indicates a new Jenkins version (2.452.3) is available for download. On the right, there are buttons for 'Or Upgrade Automatically', 'Set up agent', 'Set up cloud', and 'Dismiss'.