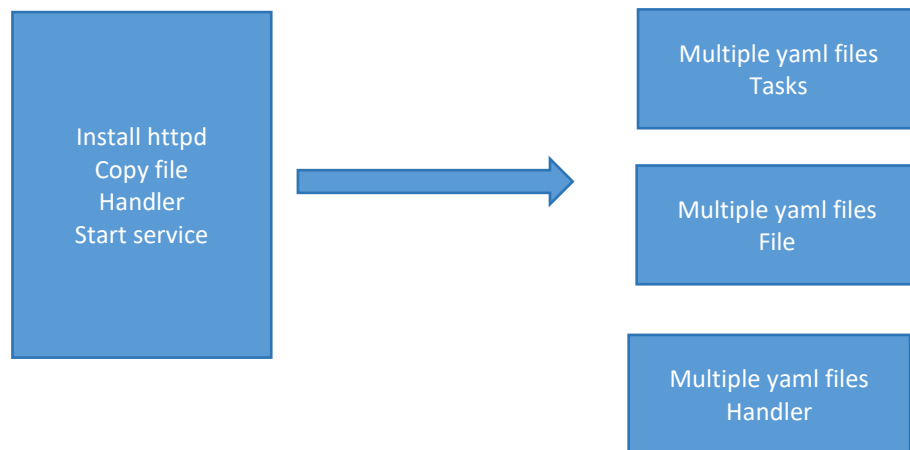Ansible 4:

Ansible roles: --> Webservers, Apache Tomcat

If you want to divide tasks from one playbook into smaller chunks or if you want to organize playbook better then we need Ansible roles. Tasks, Handlers all separate

When I breakdown one playbook into multiple chunks, then there is less ambiguity and issues. This is all possible with the concept of Ansible Roles
Whenever there are more tasks or more functionalities in our playbook, it becomes difficult to manage and maintain. Here Ansible Roles, come into picture, allow us to breakdown complex playbook into smaller chuncks that could be co-ordinated by a central entry point. To create Roles in Ansible, we have a concept called as "Ansible Galaxy".

```
[ec2-user@ip-172-31-14-91 ~]$
[ec2-user@ip-172-31-14-91 ~]$ sudo su ansible
[ansible@ip-172-31-14-91 ec2-user]$
[ansible@ip-172-31-14-91 ec2-user]$
[ansible@ip-172-31-14-91 ec2-user]$ cd ~
[ansible@ip-172-31-14-91 ~]$
[ansible@ip-172-31-14-91 ~]$
[ansible@ip-172-31-14-91 ~]$ pwd
/home/ansible
[ansible@ip-172-31-14-91 ~]$
[ansible@ip-172-31-14-91 ~]$
[ansible@ip-172-31-14-91 ~]$ mkdir roles
[ansible@ip-172-31-14-91 ~]$ cd roles
[ansible@ip-172-31-14-91 roles]$ ansible-galaxy init apache
- Role apache was created successfully
[ansible@ip-172-31-14-91 roles]$ ls -l
total 0
drwxrwxr-x. 10 ansible ansible 135 May  1 02:12 apache
[ansible@ip-172-31-14-91 roles]$
```

Practicals with Ansible roles:

sudo su ansible
cd ~
pwd ---> /home/ansible
mkdir roles
cd roles
ansible-galaxy init apache ---> apache role is created

```
(install tree)
tree apache

cd apache
vi tasks/main.yml
---
# tasks file for apache
- name: Install httpd in only Managed node1
  yum:
    name: httpd
    state: present

- name: Copy index.html file
  copy:
    src: index.html
    dest: /var/www/html/index.html
  notify:
    - restart apache
…

vi files/index.html
<h1> Ansible Role Learning </h1>

vi handlers/main.yml
---
# handlers file for apache
- name: restart apache
  service:
      name: httpd
      state: restarted
…

vi runsetup
---
- hosts: webservers
  become: true
  roles:
    - apache
…
```

```
[ansible@ip-172-31-14-91 roles]$
[ansible@ip-172-31-14-91 roles]$ tree apache
apache
├── README.md
├── defaults
│   └── main.yml
├── files
├── handlers
│   └── main.yml
├── meta
│   └── main.yml
├── tasks
│   └── main.yml
├── templates
├── tests
│   ├── inventory
│   └── test.yml
└── vars
    └── main.yml

8 directories, 8 files
[ansible@ip-172-31-14-91 roles]$
```

One playbook is divided into three parts

[ansible@ip-172-31-14-91 roles]$ cd apache/
[ansible@ip-172-31-14-91 apache]$ vi tasks/main.yml
[ansible@ip-172-31-14-91 apache]$
[ansible@ip-172-31-14-91 apache]$
[ansible@ip-172-31-14-91 apache]$ cat tasks/main.yml
---
tasks file for apache
- name: Install httpd in only Managed node1
  yum:
    name: httpd
    state: present

- name: Copy index.html file
  copy:
    src: index.html
    dest: /var/www/html/index.html
  notify:
    - restart apache
…

Step 2:
[ansible@ip-172-31-14-91 apache]$ vi files/index.html
[ansible@ip-172-31-14-91 apache]$ cat files/index.html
<h1> Ansible Role Learning </h1>

```
8 directories, 8 files
[ansible@ip-172-31-14-91 roles]$ cd apache/
[ansible@ip-172-31-14-91 apache]$ vi tasks/main.yml
[ansible@ip-172-31-14-91 apache]$
[ansible@ip-172-31-14-91 apache]$
[ansible@ip-172-31-14-91 apache]$ cat tasks/main.yml
---
# tasks file for apache
- name: Install httpd in only Managed node1
  yum:
    name: httpd
    state: present

- name: Copy index.html file
  copy:
    src: index.html
    dest: /var/www/html/index.html
  notify:
    - restart apache
...
[ansible@ip-172-31-14-91 apache]$ vi files/index.html
[ansible@ip-172-31-14-91 apache]$ cat files/index.html
<h1> Ansible Role Learning </h1>

[ansible@ip-172-31-14-91 apache]$
```

[ansible@ip-172-31-14-91 apache]$ cat handlers/main.yml

---

handlers file for apache

- name: restart apache

 service:

    name: httpd

    state: restarted

…

[ansible@ip-172-31-14-91 apache]$ cd ..

[ansible@ip-172-31-14-91 roles]$ tree apache

apache

├── README.md

├── defaults

| └── main.yml

├── files

| └── index.html

├── handlers

| └── main.yml

├── meta

| └── main.yml

├── tasks

| └── main.yml

├── templates

├── tests

| ├── inventory

| └── test.yml

└── vars

    └── main.yml

8 directories, 9 files

```
[ansible@ip-172-31-14-91 apache]$ cd ..
[ansible@ip-172-31-14-91 roles]$ tree apache
apache
├── README.md
├── defaults
│   └── main.yml
├── files
│   └── index.html
├── handlers
│   └── main.yml
├── meta
│   └── main.yml
├── tasks
│   └── main.yml
├── templates
├── tests
│   ├── inventory
│   └── test.yml
└── vars
    └── main.yml

8 directories, 9 files
```

[ansible@ip-172-31-14-91 ~]$ vi runsetup
[ansible@ip-172-31-14-91 ~]$ cat runsetup

- hosts: webservers
  become: true
  roles:
    - apache
...

```
[ansible@ip-172-31-14-91 roles]$ cd ..
[ansible@ip-172-31-14-91 ~]$ ls -l
total 32
-rw-rw-r--. 1 ansible ansible  108 Apr 14 01:10 01-ping.yml
-rw-------. 1 root    root     154 Apr 29 03:15 02-create-file.yml
-rw-rw-r--. 1 ansible ansible  146 Apr 27 22:58 03-install-git.yml
-rw-rw-r--. 1 ansible ansible  369 Apr 27 23:24 04-website-hosting.yml
-rw-rw-r--. 1 ansible ansible  374 Apr 28 02:55 05-website-hosting-variable.yml
-rw-rw-r--. 1 ansible ansible  408 Apr 28 03:19 06-playbook-variables.yml
-rw-------. 1 root    root    2493 Apr 29 02:50 07-handlers.yml
-rw-rw-r--. 1 ansible ansible   72 Apr 28 03:11 index.html
drwxrwxr-x. 3 ansible ansible   20 May  1 02:12 roles
[ansible@ip-172-31-14-91 ~]$ vi runsetup
[ansible@ip-172-31-14-91 ~]$ cat runsetup
---
- hosts: webservers
  become: true
  roles:
    - apache
...
[ansible@ip-172-31-14-91 ~]$
```

```
[ansible@ip-172-31-14-91 ~]$ vi runsetup.yml
[ansible@ip-172-31-14-91 ~]$ ansible-playbook runsetup.yml

PLAY [webservers] ***************************************

TASK [Gathering Facts] *********************************
[WARNING]: Platform linux on host 172.31.5.77 is using the disco
interpreter could change the meaning of that path. See https://d
information.
ok: [172.31.5.77]

TASK [apache : Install httpd in only Managed node1] ************
ok: [172.31.5.77]

TASK [apache : Copy index.html file] *************************
changed: [172.31.5.77]

RUNNING HANDLER [apache : restart apache] ********************
changed: [172.31.5.77]

PLAY RECAP *****************************************************
172.31.5.77                  : ok=4    changed=2    unreachable=0
```
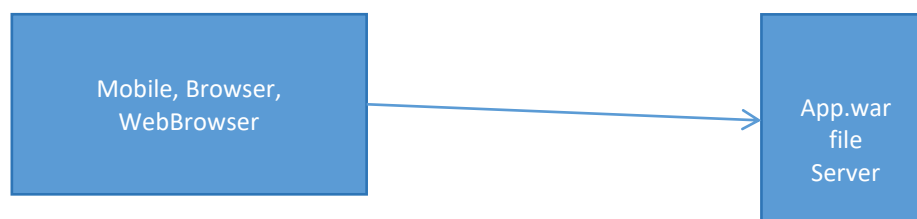
Ansible Tower:

Ansible Tower (now part of **Red Hat Ansible Automation Platform**) is a
web-based UI and REST API that makes it easier to manage and scale
Ansible across an enterprise.

Ansible Tower is a paid software managed by Red Hat which provides UI to
create/schedule/manage/execute playbook

Webservers:
Web service over internet --->

App is packaged as a .war file. Keeping application file inside Server software is called as Deployment

Webservers --->
Server software is used to run our web applications and users across the globe with internet can
access our web applications by sending requests to server

Users will use Client softare or web browsers to send requests to server

Server is responsible to handle requests and response

There are many server softwares in the market to run our web apps
Example: Tomcat, Glassfish, WebLogic, JBoss, IIS, Websphere

Is it compulsory to have a server software to run our application? The answer is Yes

Every web application must be present inside the Server software. To run our webapp server is mandatory

The process of executing web applications by using server is called as Deployment

Tomcat server:
It is free and open-source sofware, which is developed by Apache organization
Tomcat server is developed using Java language. Hence, to run Tomcat, Java also has to be installed
Tomcat supports multiple operating system
By default, Tomcat runs on 8080 port number (if needed we can change the port number of Tomcat)