

# Structuring Ansible Playbook

---

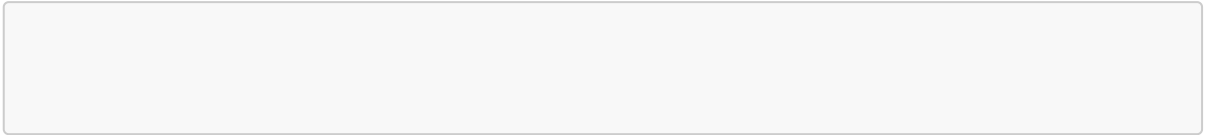
- here in this section we will learn about using the `includes` and `imports`, the difference between `Static` and `Dynamic` inclusion
- here also we will learn how to use `Tags` for `playbook` component reuse
- we will take the knowledge that we learn in other section and learn how can we actually translate the existing effort into the Ansible Roles
- creating a Role of components and task i.e the `webserver` role

## Usage of imports and include to Structure Your Playbook

---

- in this one we will see how many way we can use the `includes` and `imports`
  - `include_tasks`
  - `import_tasks`
  - `Static` vs `Dynamic`
  - `import_playbook`
- **Case01**
  - the `ansible` engine provide directive that we can use for bundling task and even playbooks into separate files
  - there might be verity of reason for that
    - lets suppose we need the packages to be installed prior to the execution of the `playbook` as the pre-requisite, hence we can put that to a separate file and include that in your main `playbook`
    - we can define the - `include_tasks` derivative in the main `playbook` referenceing to the `playbook` from where we want the task to be included
    - when we execute the main `playbook` then we can see that its been including and executing task listed in the `yaml` file which been mentioned against `include_tasks` directives
    - when we look into the output then we can see that its included the another `playbook` with the name as `include_tasks`
    - then it execute the task reside in the `playbook` which is under the `playbook` which is being included

- while defining the `playbook` which we want to be included we don't need the `- hosts:` or `- task:` derivative we can simply mentioned with the `- name:` and `packages` that we want to use , it will autouse the `hosts` and `task` from the main `playbook` from where its been imported



- **Case02**

- the `import_tasks` statements were same as the `include_tasks` statement , if we execute we can see the same output
- but here the `task included` statement that we saw in the `include_tasks` will not be displayed
- but there is a major difference there as well

<!-- | `import_tasks` | `include_tasks` |

|||-->

