Variables

In Ansible, variables are used to store values that can be referenced in playbooks or roles. Variables can be defined at various levels, including global, play, role, or host level. Here are the main types of variables in Ansible:

1. **Global Variables:**
    - Global variables are defined in the **group_vars** or **host_vars** directories in the inventory. These variables apply to all hosts in a particular group or a specific host, respectively.
2. **Play Variables:**
    - Variables can be defined within a specific play in a playbook. These play-specific variables only apply to that particular play.
3. **Facts:**
    - Ansible gathers facts about remote systems before running tasks. These facts are stored in variables that you can reference in your playbooks. For example, the **ansible_hostname** fact contains the hostname of the remote system.
4. **Role Variables:**
    - Roles in Ansible can have their own variable files. Variables defined in a role are specific to that role and can be used to customize the behavior of the role.
5. **User-Defined Variables:**
    - You can define your own variables directly in the playbook. These variables are often used to parameterize tasks and make playbooks more flexible.

Here's an example of how variables are defined in an Ansible playbook:

```yaml
---
- name: Example Playbook
  hosts: web_servers
  vars:
    http_port: 80
    app_name: "my_web_app"
  tasks:
    - name: Ensure Apache is installed
      package:
        name: apache2
        state: present

    - name: Start Apache service
      service:
        name: apache2
        state: started
    - name: Deploy the web application
      copy:
        src: /path/to/{{ app_name }}.tar.gz
        dest: /var/www/html/
      notify:
        - Restart Apache
  handlers:
    - name: Restart Apache
      service:
        name: apache2
        state: restarted
```

In this example:

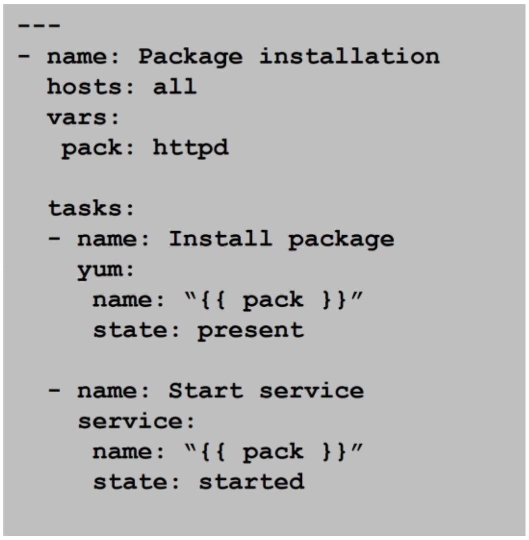
- **http_port** and **app_name** are play-level variables.
- **ansible_hostname** is an example of a fact.
- The role can also have its own variables in a separate **vars** directory.

Variables can be accessed using Jinja2 templating syntax, allowing for dynamic and flexible configurations in Ansible playbooks.

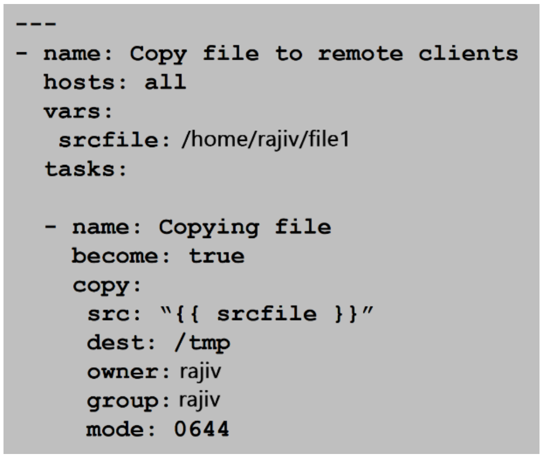
Variables are like containers that hold the defined value which can be used repetitively.

- Name can include letters, numbers, and underscore
- Name should always start with a letter
- Cannot have a space, dots (.) or hyphen (-) in variable name
- Variables can be defined inside of inventory files as well

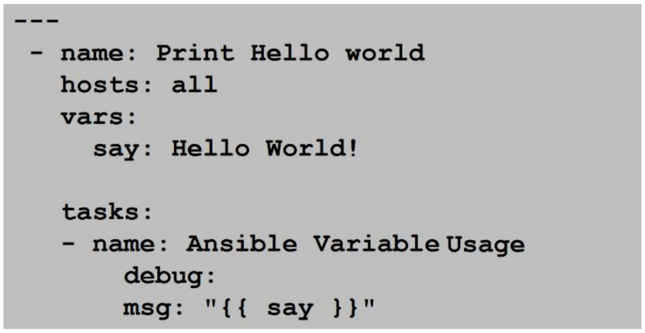
Example



```
---
- name: Package installation
  hosts: all
  vars:
   pack: httpd

  tasks:
  - name: Install package
    yum:
     name: “{{ pack }}”
     state: present

  - name: Start service
    service:
     name: “{{ pack }}”
     state: started
```

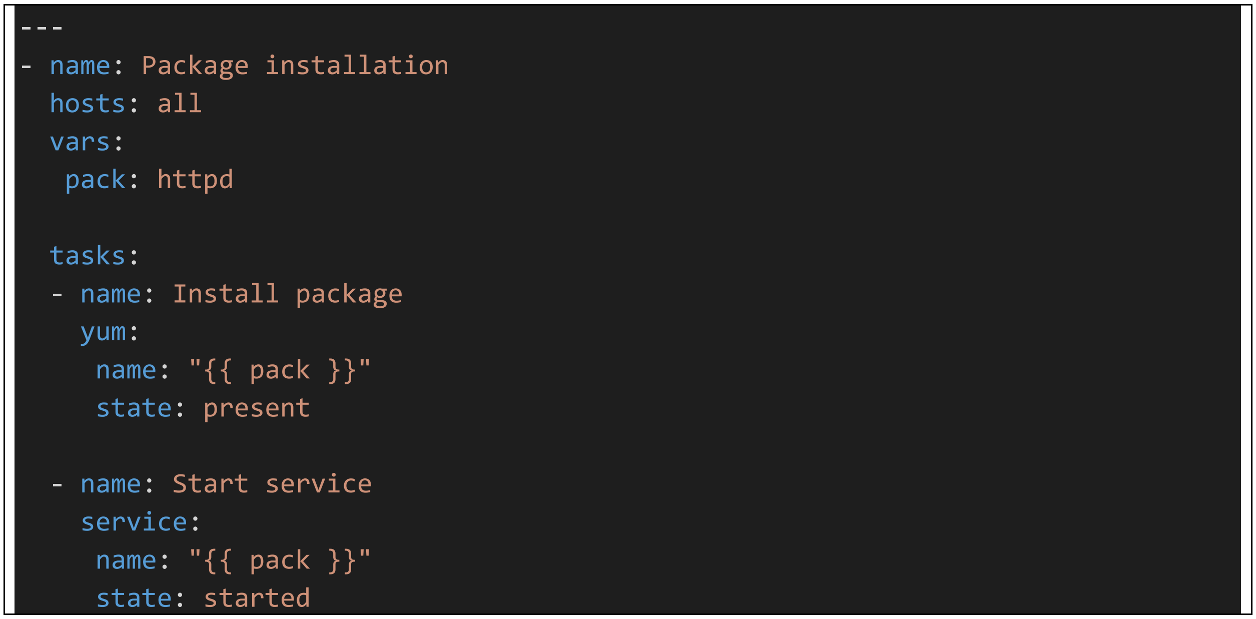


```
---
- name: Copy file to remote clients
  hosts: all
  vars:
   srcfile: /home/rajiv/file1
  tasks:

  - name: Copying file
    become: true
    copy:
     src: “{{ srcfile }}”
     dest: /tmp
     owner: rajiv
     group: rajiv
     mode: 0644
```



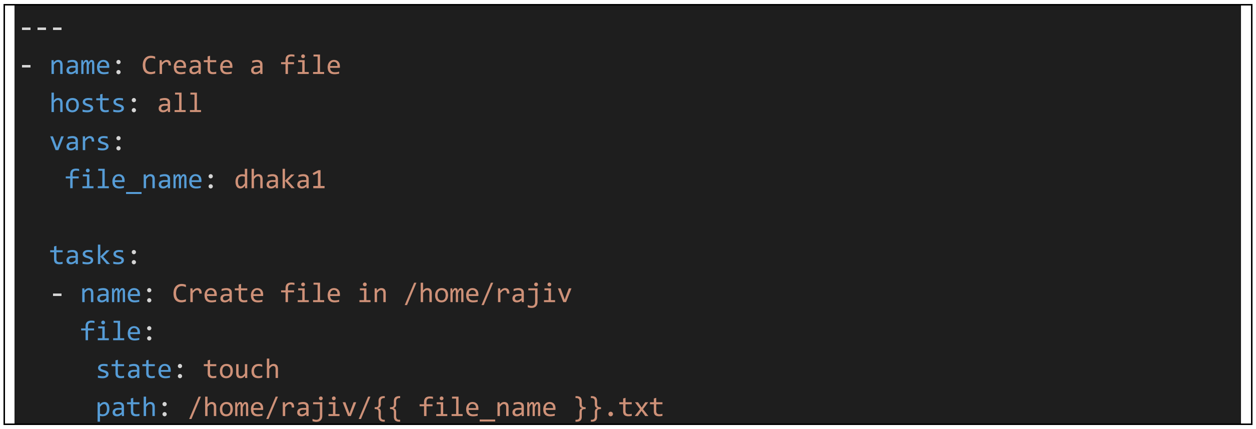
```
---
 - name: Print Hello world
   hosts: all
   vars:
     say: Hello World!

   tasks:
   - name: Ansible Variable Usage
       debug:
       msg: "{{ say }}"
```

Example-1: install and start a package.



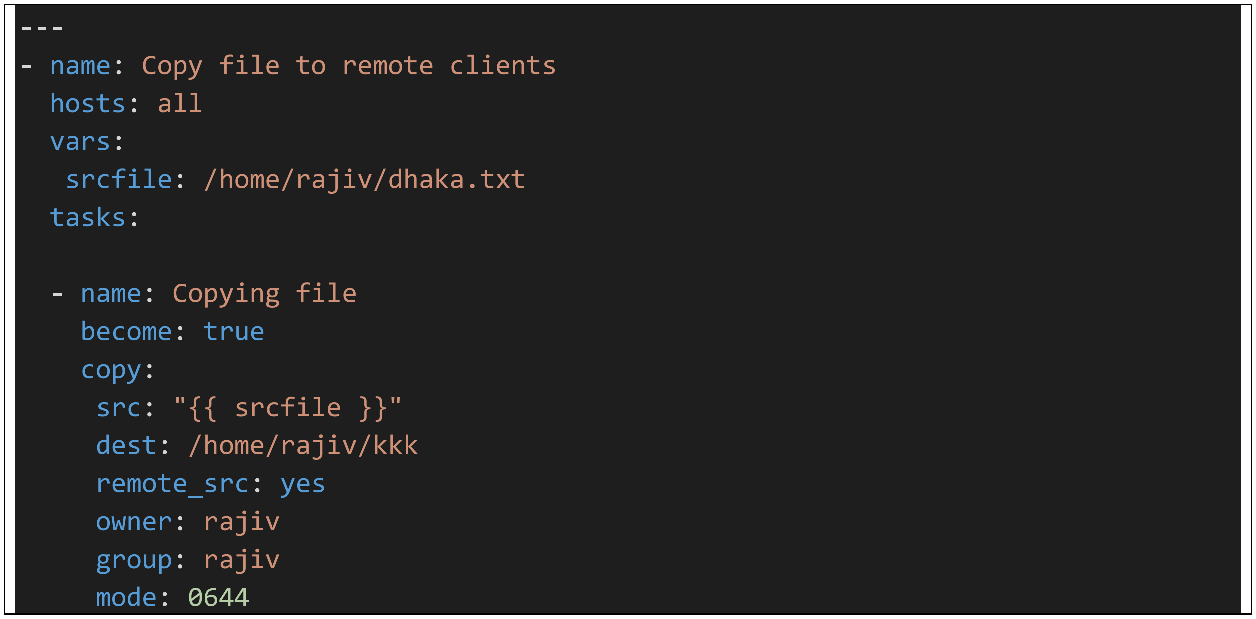
```
---
- name: Package installation
  hosts: all
  vars:
   pack: httpd

  tasks:
  - name: Install package
    yum:
     name: "{{ pack }}"
     state: present

  - name: Start service
    service:
     name: "{{ pack }}"
     state: started
```

Example-2: Create a file to remote host



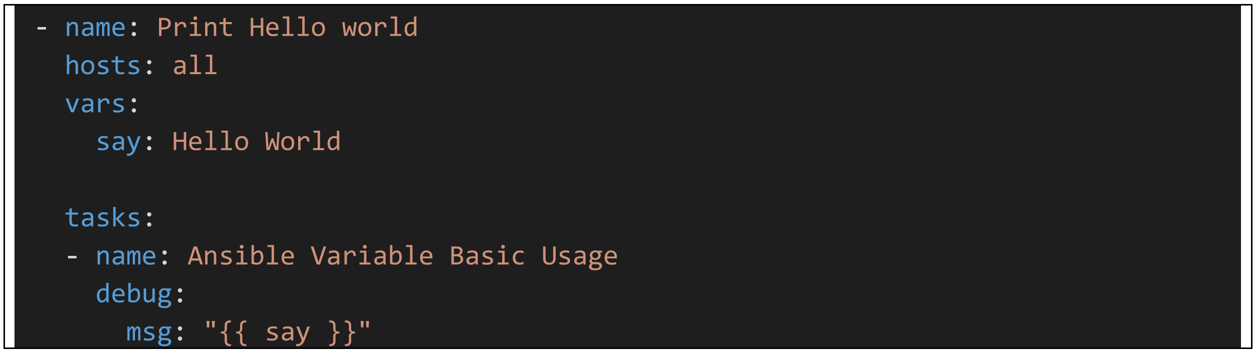
```
---
- name: Create a file
  hosts: all
  vars:
   file_name: dhaka1

  tasks:
  - name: Create file in /home/rajiv
    file:
     state: touch
     path: /home/rajiv/{{ file_name }}.txt
```

Example-3: Copy file to remote host



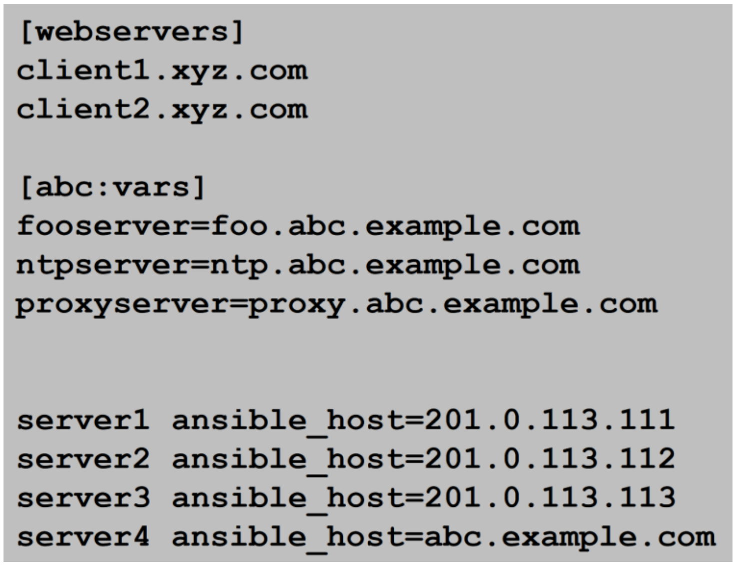
```
---
- name: Copy file to remote clients
  hosts: all
  vars:
   srcfile: /home/rajiv/dhaka.txt
  tasks:

  - name: Copying file
    become: true
    copy:
     src: "{{ srcfile }}"
     dest: /home/rajiv/kkk
     remote_src: yes
     owner: rajiv
     group: rajiv
     mode: 0644
```

Example-4: Prine a message



```
 - name: Print Hello world
   hosts: all
   vars:
     say: Hello World

   tasks:
   - name: Ansible Variable Basic Usage
     debug:
       msg: "{{ say }}"
```

Variables use in Inventory File



```
[webservers]
client1.xyz.com
client2.xyz.com

[abc:vars]
fooserver=foo.abc.example.com
ntpserver=ntp.abc.example.com
proxyserver=proxy.abc.example.com


server1 ansible_host=201.0.113.111
server2 ansible_host=201.0.113.112
server3 ansible_host=201.0.113.113
server4 ansible_host=abc.example.com
```