

## Ansible Variables – Complete Guide

Ansible variables are **key-value pairs** that allow you to **parameterize** playbooks and templates for flexibility, reusability, and control.

---

### Why Use Variables?

Variables let you:

- Avoid hardcoding values.
  - Write dynamic, reusable code.
  - Apply configuration to different environments or hosts.
- 

### Declaring Variables

#### 1. Inline in a Playbook

```
- hosts: web
  vars:
    app_port: 8080
    app_name: myapp
```

#### 2. Host or Group Vars

- ♦ **Host-specific (in `host_vars/hostname.yml`)**

```
ansible_host: 192.168.1.10
env: production
```

- ♦ **Group-specific (in `group_vars/web.yml`)**

```
app_user: nginx
timezone: UTC
```

### 3. Command Line

```
ansible-playbook playbook.yml -e "env=dev region=us-east-1"
```

### 4. Inventory File

```
[web]  
web1 ansible_host=192.168.1.10 app_port=80
```

### 5. Vars Files

```
# vars/common.yml  
package_name: nginx
```

In Playbook:

```
vars_files:  
  - vars/common.yml
```

---



## Variable Precedence (High to Low)

Precedence (High → Low)	Example
Extra vars via CLI	<code>-e "foo=bar"</code>
Task-level vars	Inside a task
Block-level vars	Inside a block
Role defaults	<code>defaults/main.yml</code> in roles
Inventory vars	<code>host_vars / group_vars</code>
Playbook vars	<code>vars:</code> section
Vars files	<code>vars_files:</code> section
Set facts	Using <code>set_fact</code>

Environment variables      `lookup( 'env' ,  
'HOME' )`

Default in Jinja2      `{{ var`

---

## Types of Variables

### 1. Scalar Variable

env: production

### 2. List Variable

users:

- alice
- bob
- charlie

### 3. Dictionary Variable

db\_config:

user: admin  
pass: secret

---

## Using Variables

### ♦ In Playbooks:

- name: Install package  
apt:  
  name: "{{ package\_name }}"

### ♦ In Templates ( **.j2** ):

Welcome to {{ env }} environment

---

## **set\_fact Module**

To define variables **dynamically at runtime**:

- name: Set dynamic variable

set\_fact:

full\_name: "{{ first\_name }}" "{{ last\_name }}"

---

## **Best Practices**

- Use descriptive names (e.g., `nginx_port`, `app_path`)
  - Use `group_vars/` and `host_vars/` for clean separation
  - Store secrets in Ansible Vault
  - Avoid hardcoding in tasks
-