

PRACTICAL QUESTION 1 – STEP-BY-STEP ANSIBLE PLAYBOOK GUIDE

INTRODUCTION

This guide explains how to complete Practical Question 1. You will learn how to install Apache and MariaDB using Ansible, start and enable the services, copy a custom HTML file, and use variables. A bonus section explains how to organise your playbook using roles.

STEP 1 — Create Your Working Directory

Run the following:

```
mkdir ansible_q1
cd ansible_q1
```

STEP 2 — Create the Inventory File

Create inventory.ini:

```
[web]
localhost ansible_connection=local
```

STEP 3 — Create the Ansible Playbook (site.yml)

Example content:

```
- name: Install Apache and MariaDB hosts: web become: true vars: apache_pkg: apache2 mariadb_pkg: mariadb-server
tasks:
  - name: Install packages package: name: "{{ item }}" state: present loop: - "{{ apache_pkg }}" - "{{ mariadb_pkg }}"
  - name: Start and enable Apache service: name: apache2 state: started enabled: yes
  - name: Start and enable MariaDB service: name: mariadb state: started enabled: yes
  - name: Copy index.html
    copy: src: index.html dest: /var/www/html/index.html
```

STEP 4 — Create index.html

Example:

```
<html>
<body>
<h1>Ansible Deployment Successful</h1>
</body>
</html>
```

STEP 5 — Bonus: Using Roles

Run:

```
ansible-galaxy init roles/webserver
ansible-galaxy init roles/database
```

Move task files and variables into appropriate folders to modularize your playbook.

STEP 6 — Run the Playbook

Run:

```
ansible-playbook -i inventory.ini site.yml
```

EXPECTED OUTPUT

You should see tasks such as:

TASK [Install packages] – ok/changed

TASK [Start and enable services] – ok

PLAY RECAP – failed=0

CONCLUSION

You now have a complete and fully working Ansible setup that satisfies all requirements of Practical Question 1.