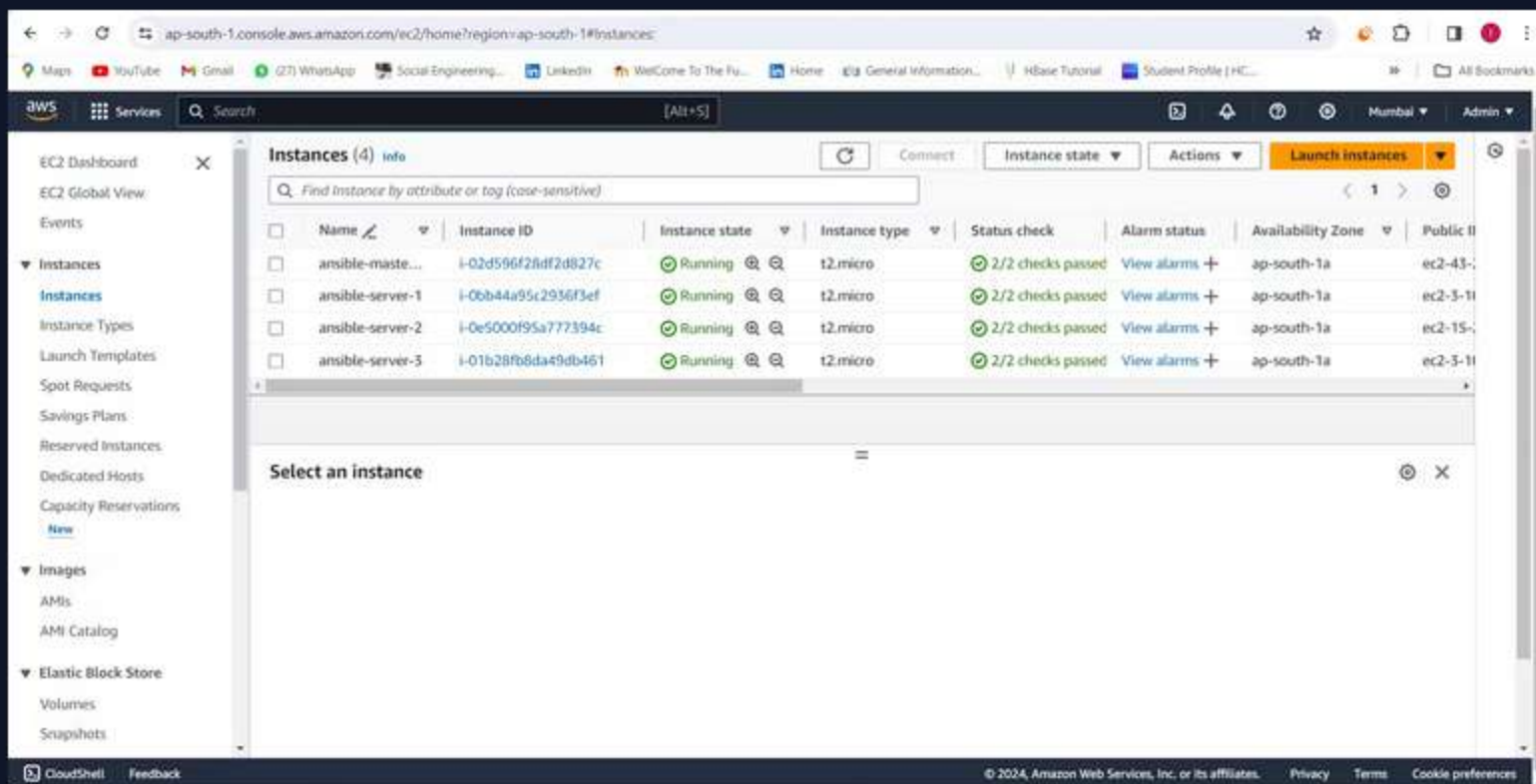


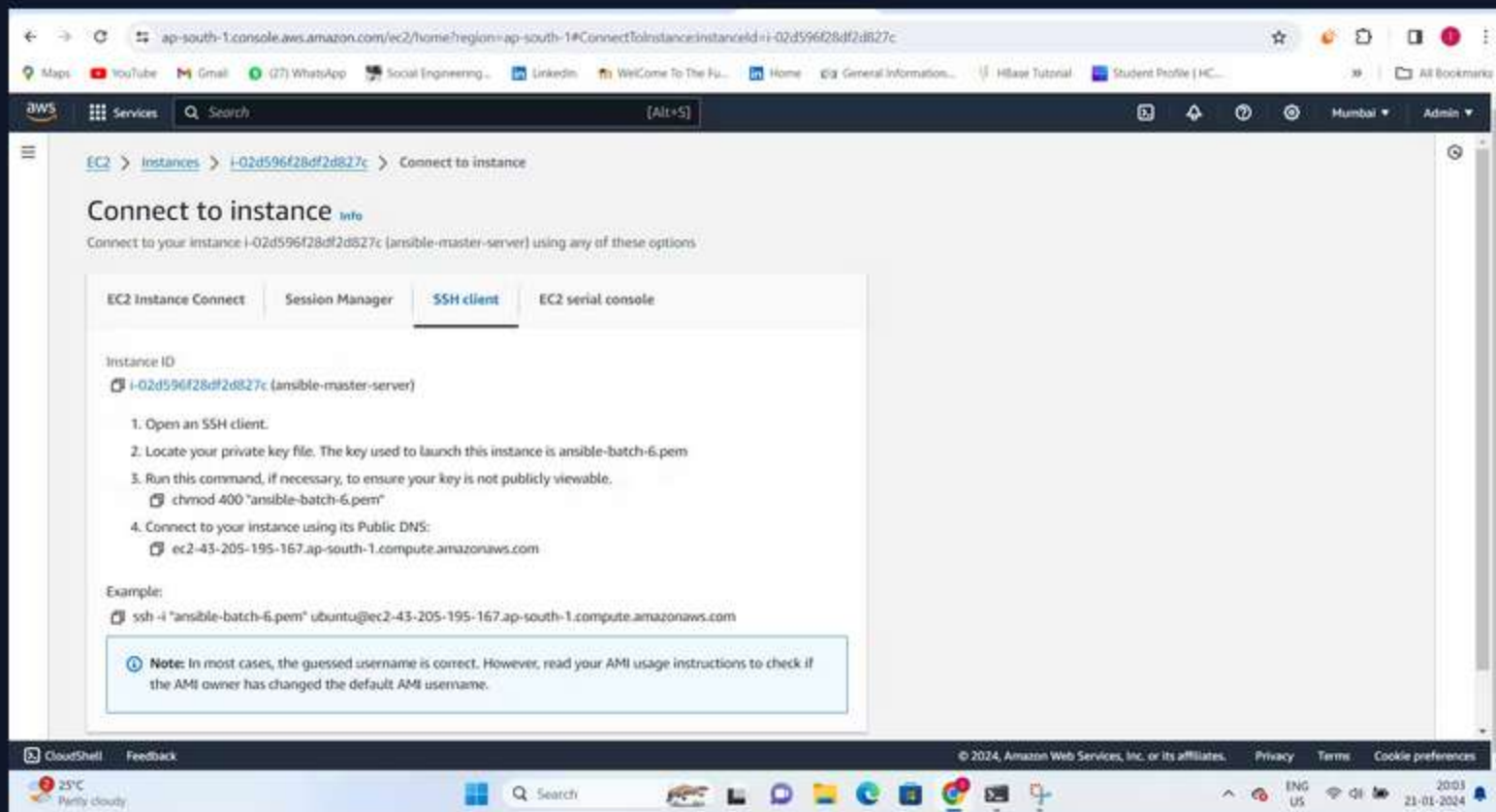
## Task-01: Create 3 EC2 instances with the same key pair

1. Use the AWS Management Console or AWS CLI to create three EC2 instances.
2. Make sure to use the same key pair for all three instances.
3. Note down the public IP addresses of the instances for later use.



The screenshot displays the AWS Management Console interface for the 'Instances' page in the 'ap-south-1' region. The left-hand navigation pane shows various AWS services, with 'Instances' selected. The main content area, titled 'Instances (4) info', contains a search bar and a table listing four EC2 instances. All instances are in a 'Running' state and are of the 't2.micro' type. The instances are named 'ansible-master', 'ansible-server-1', 'ansible-server-2', and 'ansible-server-3'. Each instance has a 'Status check' of '2/2 checks passed' and an 'Alarm status' of 'View alarms'. The 'Availability Zone' for all instances is 'ap-south-1a'. The 'Public IP' addresses are partially visible as 'ec2-43...', 'ec2-3-11', 'ec2-15...', and 'ec2-3-11'. Below the table, there is a 'Select an instance' section. The bottom of the console shows the footer with copyright information for Amazon Web Services, Inc.


Name	Instance ID	Instance state	Instance type	Status check	Alarm status	Availability Zone	Public IP
ansible-master	i-02d596f28df2d827c	Running	t2.micro	2/2 checks passed	View alarms	ap-south-1a	ec2-43...
ansible-server-1	i-06b44a95c2936f3ef	Running	t2.micro	2/2 checks passed	View alarms	ap-south-1a	ec2-3-11
ansible-server-2	i-0e5000f95a777394c	Running	t2.micro	2/2 checks passed	View alarms	ap-south-1a	ec2-15...
ansible-server-3	i-01b28fb8da49db461	Running	t2.micro	2/2 checks passed	View alarms	ap-south-1a	ec2-3-11



## Task-02: Install Ansible in the host server

Connect to your Ansible host server and install Ansible. Assuming you're using Ubuntu, you can use the following commands:

```
sudo apt update  
sudo apt install ansible
```

COPY 

```
ubuntu@ip-172-31-32-232: ~$ sudo apt update
compliance features.
https://ubuntu.com/aws/pro

Expanded Security Maintenance for Applications is not enabled.


18 updates can be applied immediately.
To see these additional updates run: apt list --upgradable

1 additional security update can be applied with ESM Apps.
Learn more about enabling ESM Apps service at https://ubuntu.com/esm

Last login: Sat Jan 20 13:09:52 2024 from 103.150.138.64
ubuntu@ip-172-31-32-232:~$ sudo apt update
sudo apt install ansible
Hit:1 http://ap-south-1.ec2.archive.ubuntu.com/ubuntu jammy InRelease
Get:2 http://ap-south-1.ec2.archive.ubuntu.com/ubuntu jammy-updates InRelease [119 kB]
Hit:3 http://ap-south-1.ec2.archive.ubuntu.com/ubuntu jammy-backports InRelease
Get:4 http://security.ubuntu.com/ubuntu jammy-security InRelease [110 kB]
Get:5 http://ap-south-1.ec2.archive.ubuntu.com/ubuntu jammy-updates/main amd64 Packages [1282 kB]
Get:6 http://ap-south-1.ec2.archive.ubuntu.com/ubuntu jammy-updates/universe amd64 Packages [1031 kB]
Hit:7 https://ppa.launchpadcontent.net/ansible/ansible/ubuntu jammy InRelease
Fetched 2543 kB in 1s (1706 kB/s)
Reading package lists... Done
Building dependency tree... Done
Reading state information... Done
18 packages can be upgraded. Run 'apt list --upgradable' to see them.
Reading package lists... Done
Building dependency tree... Done
Reading state information... Done
ansible is already the newest version (8.7.0-1ppa-jammy).
0 upgraded, 0 newly installed, 0 to remove and 18 not upgraded.
ubuntu@ip-172-31-32-232:~$ sudo apt install ansible
Reading package lists... Done
Building dependency tree... Done
Reading state information... Done
ansible is already the newest version (8.7.0-1ppa-jammy).
0 upgraded, 0 newly installed, 0 to remove and 18 not upgraded.
```

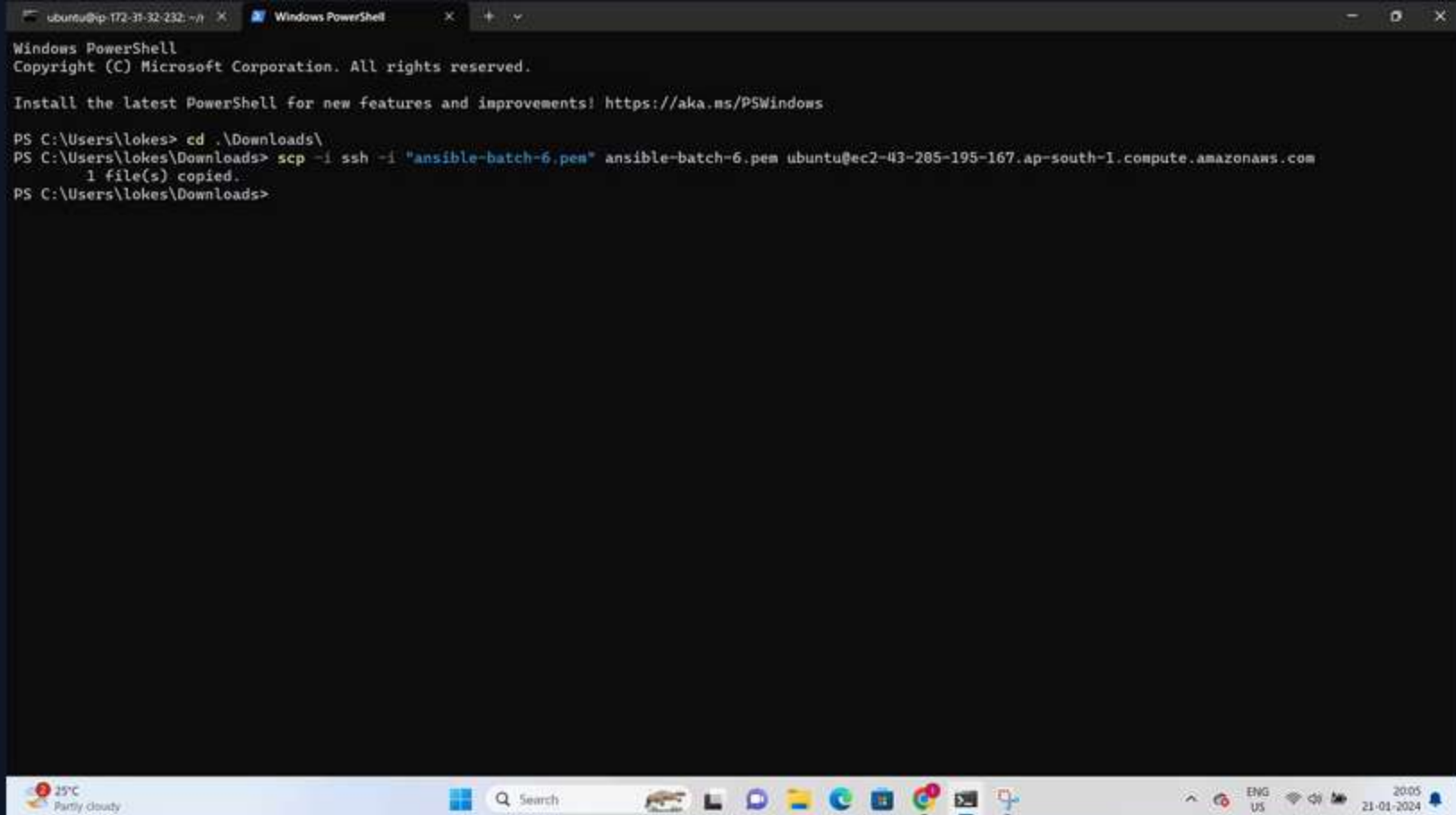
### Task-03: Copy the private key from local to Ansible host

Assuming you have the private key on your local machine, use `scp` to copy it to your Ansible host:

COPY 

```
scp /path/to/your/private-key.pem user@ansible_host_ip:/home/ubuntu/.ssh/
```

```
ubuntu@ip-172-31-32-232: ~$  
Windows PowerShell  
Copyright (C) Microsoft Corporation. All rights reserved.  
  
Install the latest PowerShell for new features and improvements! https://aka.ms/PSWindows  
  
PS C:\Users\lokes> cd .\Downloads\  
PS C:\Users\lokes\Downloads> scp -i ssh -i "ansible-batch-6.pem" ansible-batch-6.pem ubuntu@ec2-43-205-195-167.ap-south-1.compute.amazonaws.com  
1 file(s) copied.  
PS C:\Users\lokes\Downloads>
```



Replace `/path/to/your/private-key.pem`, `user`, `ansible_host_ip` with the actual path to your private key, your server's username, and the IP address of your Ansible host.

## Task-04: Access the inventory file using sudo vim

Edit the Ansible inventory file to include the details of your EC2 instances. Open the inventory file using `sudo vim /etc/ansible/hosts` and add the following:

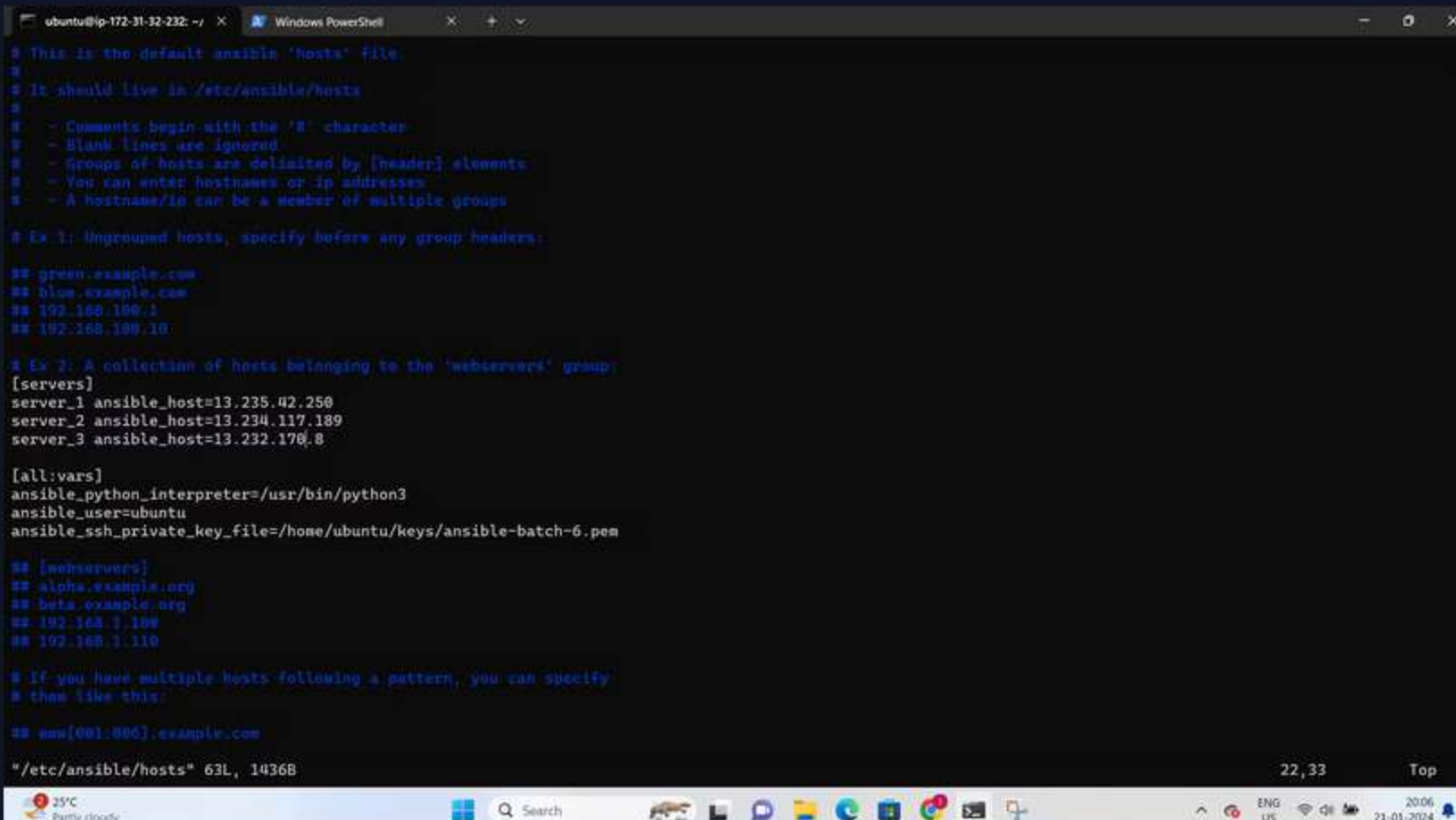


```
[web_servers]
```

```
ec2_instance_ip1
```

```
ec2_instance_ip2
```

```
ec2_instance_ip3
```



```
# This is the default ansible 'hosts' file.
#
# It should live in /etc/ansible/hosts
#
# - Comments begin with the '#' character
# - Blank lines are ignored
# - Groups of hosts are delimited by [header]: elements
# - You can enter hostnames or ip addresses
# - A hostname/ip can be a member of multiple groups
#
# Ex 1: Ungrouped hosts, specify before any group headers:
## green.example.com
## blue.example.com
## 192.168.100.1
## 192.168.100.10

# Ex 2: A collection of hosts belonging to the 'servers' group:
[servers]
server_1 ansible_host=13.235.42.250
server_2 ansible_host=13.234.117.189
server_3 ansible_host=13.232.170.8

[all:vars]
ansible_python_interpreter=/usr/bin/python3
ansible_user=ubuntu
ansible_ssh_private_key_file=/home/ubuntu/keys/ansible-batch-6.pem

## [web_servers]
## alpha.example.org
## beta.example.org
## 192.168.1.100
## 192.168.1.110

# If you have multiple hosts following a pattern, you can specify
# them like this:
## www[001:006].example.com

"/etc/ansible/hosts" 63L, 1436B
```

Replace `ec2_instance_ip1`, `ec2_instance_ip2`, and `ec2_instance_ip3` with the public IP addresses of your EC2 instances.

## Task-05: Create a playbook to install Nginx

Create an Ansible playbook, let's say `nginx_install.yml`, with the following content:

COPY 

```
---  
  
- name: Install Nginx  
  hosts: web_servers  
  become: yes  
  
  tasks:  
    - name: Update apt cache  
      apt:  
        update_cache: yes  
  
    - name: Install Nginx  
      apt:  
        name: nginx  
        state: present  
  
    - name: Start Nginx service  
      service:  
        name: nginx  
        state: started  
        enabled: yes
```

```
name: nginx_install
hosts: servers
become: yes

tasks:
  - name: nginx
    apt:
      name: nginx
      state: latest
```

## Task-06: Deploy a sample webpage using the Ansible playbook

Extend the playbook ( `nginx_install.yml` ) to deploy a sample webpage:

```
---
```

- **name:** Install Nginx and deploy sample webpage

**hosts:** web\_servers

**become:** yes

**tasks:**

# ... (previous tasks)

- **name:** Create sample index.html

**copy:**

**content:** "<html><body><h1>Hello from Ansible!</h1></body></html>"

**dest:** /var/www/html/index.html

- **name:** Restart Nginx service

**service:**

**name:** nginx

**state:** restarted



```
ubuntu@ip-172-31-32-232:~/my_playbook$ cat docker_play.yml
-
  name: Install Docker
  hosts: servers
  become: yes

  tasks:
    - name: Install docker
      apt:
        name: docker.io
        state: latest
ubuntu@ip-172-31-32-232:~/my_playbook$ |
```

