1) Install httpd using ansible playbook, use handlers, notifiers.

Purpose: Install and start the Apache HTTP server (apache2 on Ubuntu) with a handler for service control.

Explanation:

Task Overview: This playbook installs the apache2 package, which provides the httpd web server on Ubuntu.

1. **Using Handlers and Notifiers**: If apache2 is installed or updated, the notify triggers a handler to start the service. This method is efficient because the handler only runs if there's a change, which saves resources and ensures the service is started only when needed.
2. **Modules**:
    - **apt**: Installs apache2.
    - **service**: Starts the service if notified.

Playbook Breakdown:

- **- name: Install apache2 and start service**: This step installs Apache2 and starts the service.
- **hosts: all**: The play will run on all specified hosts in the inventory.

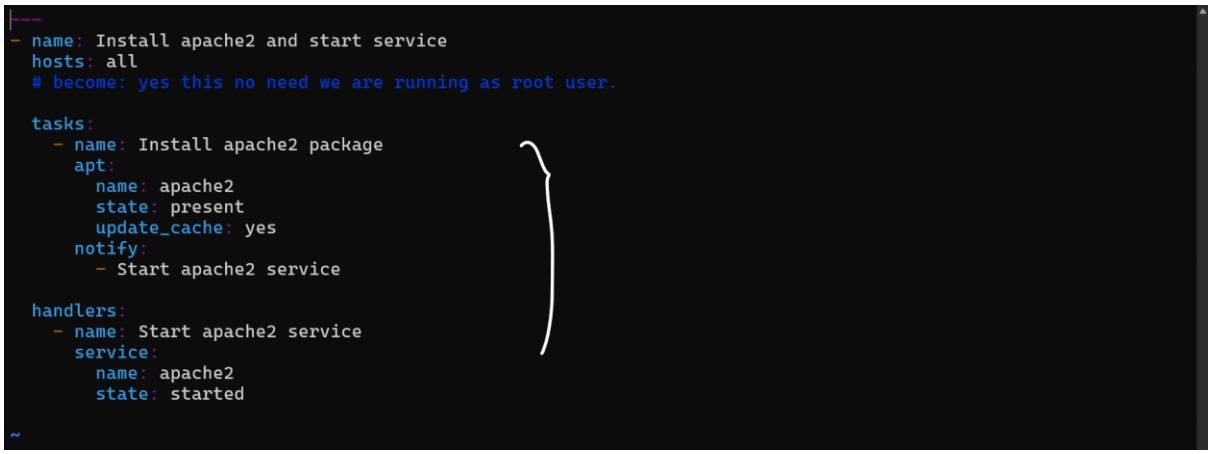
Tasks:

- **name: Install apache2 package**: This task installs the Apache2 package.
- **apt**: Uses the apt module, which is for managing packages on Debian-based systems.
- **name: apache2**: Specifies the apache2 package to be installed.
- **state: present**: Ensures the Apache2 package is installed.
- **update_cache: yes**: Updates the package cache before installation.
- **notify: Start apache2 service**: Notifies the handler to start the service after installation.

Handlers:

- **name: Start apache2 service**: This handler starts the Apache2 service.
- **service**: Uses the service module to manage services.
- **name: apache2**: Specifies the Apache2 service to start.
- **state: started**: Ensures the Apache2 service is started.

The below one is my playbook script to Install and start the Apache HTTP server (apache2 on Ubuntu) with a handler for service control



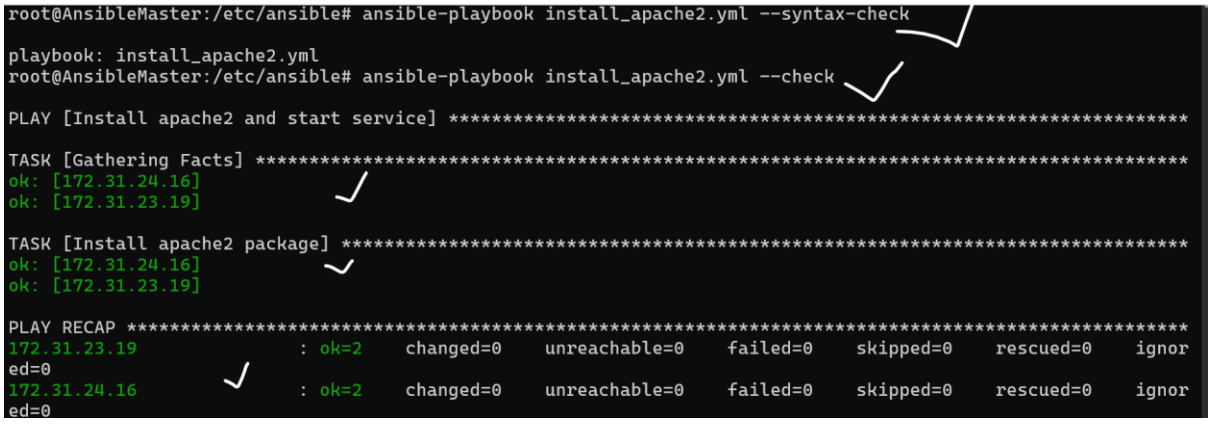
```yaml
---
- name: Install apache2 and start service
  hosts: all
  # become: yes this no need we are running as root user.

  tasks:
    - name: Install apache2 package
      apt:
        name: apache2
        state: present
        update_cache: yes
      notify:
        - Start apache2 service

  handlers:
    - name: Start apache2 service
      service:
        name: apache2
        state: started

~
```

CMD: ansible-playbook install_apache2.yml --syntax-check

The command `ansible-playbook install_apache2.yml --syntax-check` is used to check the syntax of the `install_apache2.yml` playbook for any errors without actually running it.

CMD: ansible-playbook install_apache2.yml --check

The command `ansible-playbook install_apache2.yml --check` shows what changes the playbook will make without actually applying them.



```
root@AnsibleMaster:/etc/ansible# ansible-playbook install_apache2.yml --syntax-check

playbook: install_apache2.yml
root@AnsibleMaster:/etc/ansible# ansible-playbook install_apache2.yml --check

PLAY [Install apache2 and start service] ***************************************************************

TASK [Gathering Facts] *********************************************************************************
ok: [172.31.24.16]
ok: [172.31.23.19]

TASK [Install apache2 package] *************************************************************************
ok: [172.31.24.16]
ok: [172.31.23.19]

PLAY RECAP *********************************************************************************************
172.31.23.19               : ok=2    changed=0    unreachable=0    failed=0    skipped=0    rescued=0    ignor
ed=0
172.31.24.16               : ok=2    changed=0    unreachable=0    failed=0    skipped=0    rescued=0    ignor
ed=0
```

CMD: ansible-playbook install_a

```
root@AnsibleMaster:/etc/ansible# ansible-playbook install_apache2.yml

PLAY [Install apache2 and start service] ***************************************************************

TASK [Gathering Facts] *********************************************************************************
ok: [172.31.23.19]
ok: [172.31.24.16]

TASK [Install apache2 package] *************************************************************************
changed: [172.31.24.16]
changed: [172.31.23.19]

RUNNING HANDLER [Start apache2 service] ****************************************************************
ok: [172.31.24.16]
ok: [172.31.23.19]

PLAY RECAP *********************************************************************************************
172.31.23.19               : ok=3    changed=1    unreachable=0    failed=0    skipped=0    rescued=0    ignor
ed=0
172.31.24.16               : ok=3    changed=1    unreachable=0    failed=0    skipped=0    rescued=0    ignor
ed=0

root@AnsibleMaster:/etc/ansible#
```

pache2.yml

The command `ansible-playbook install_apache2.yml` runs the playbook and applies the changes defined in the `install_apache2.yml` file to the target systems.

Now we see the status of the servers.

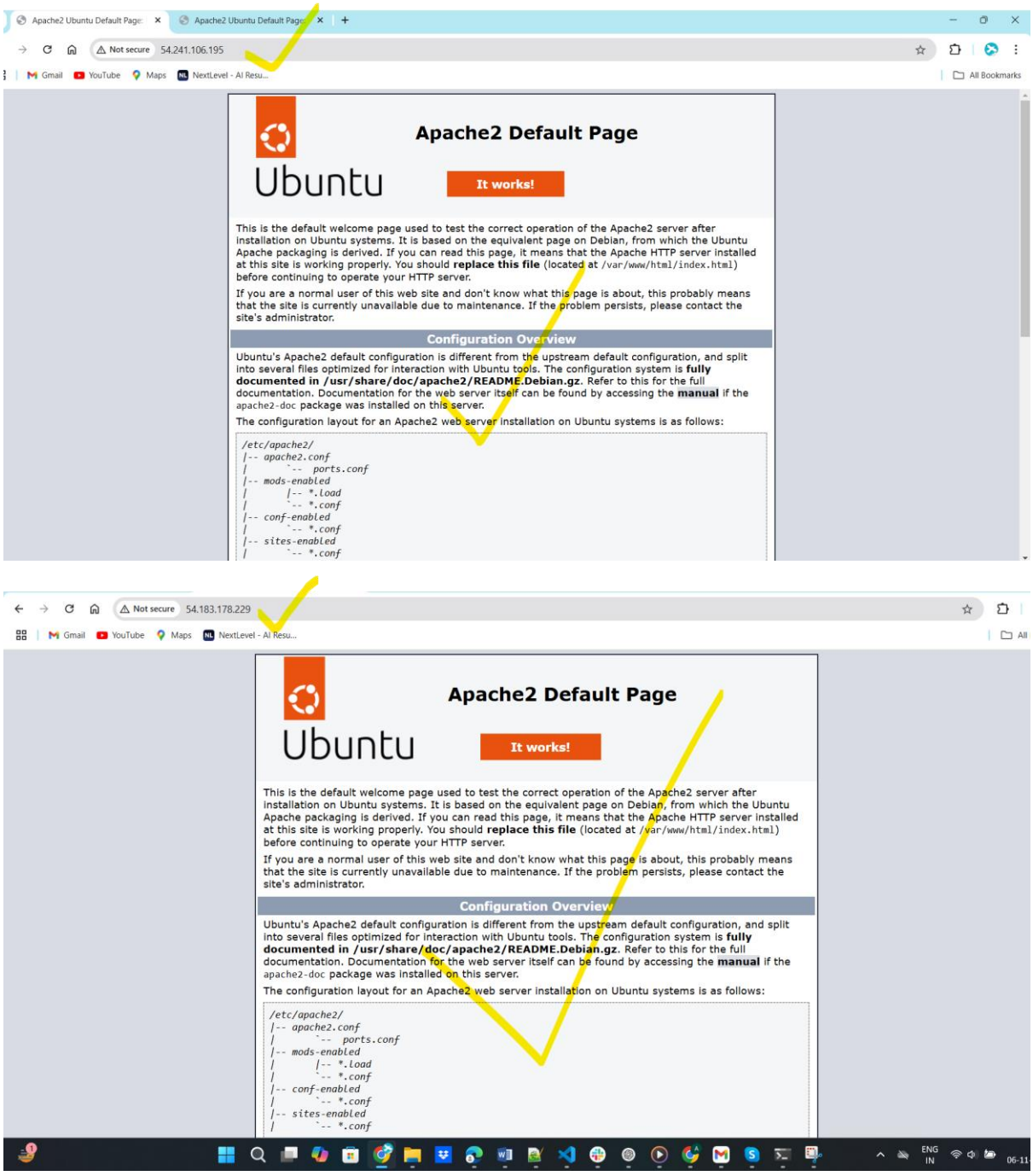
This is AnsibleNode01 server apache 2 is running.

```
ubuntu@AnsibleNode01:~$ sudo systemctl status apache2
● apache2.service - The Apache HTTP Server
     Loaded: loaded (/usr/lib/systemd/system/apache2.service; enabled; preset: enabled)
     Active: active (running) since Wed 2024-11-06 13:03:45 UTC; 35min ago
       Docs: https://httpd.apache.org/docs/2.4/
   Main PID: 13051 (apache2)
      Tasks: 55 (limit: 1130)
     Memory: 5.4M (peak: 5.6M)
        CPU: 116ms
     CGroup: /system.slice/apache2.service
             ├─13051 /usr/sbin/apache2 -k start
             ├─13054 /usr/sbin/apache2 -k start
             └─13055 /usr/sbin/apache2 -k start

Nov 06 13:03:45 AnsibleNode01 systemd[1]: Starting apache2.service - The Apache HTTP Server...
Nov 06 13:03:45 AnsibleNode01 systemd[1]: Started apache2.service - The Apache HTTP Server.
ubuntu@AnsibleNode01:~$
```

This is AnsibleNode02 server apache 2 is running.

```
ubuntu@AnsibleNode02:~$ sudo systemctl status apache2
● apache2.service - The Apache HTTP Server
     Loaded: loaded (/usr/lib/systemd/system/apache2.service; enabled; preset: enabled)
     Active: active (running) since Wed 2024-11-06 13:03:45 UTC; 35min ago
       Docs: https://httpd.apache.org/docs/2.4/
   Main PID: 13132 (apache2)
      Tasks: 55 (limit: 1130)
     Memory: 5.4M (peak: 5.6M)
        CPU: 120ms
     CGroup: /system.slice/apache2.service
             ├─13132 /usr/sbin/apache2 -k start
             ├─13135 /usr/sbin/apache2 -k start
             └─13136 /usr/sbin/apache2 -k start

Nov 06 13:03:45 AnsibleNode02 systemd[1]: Starting apache2.service - The Apache HTTP Server...
Nov 06 13:03:45 AnsibleNode02 systemd[1]: Started apache2.service - The Apache HTTP Server.
ubuntu@AnsibleNode02:~$
```

And also we check the it’s accessible on browsers.

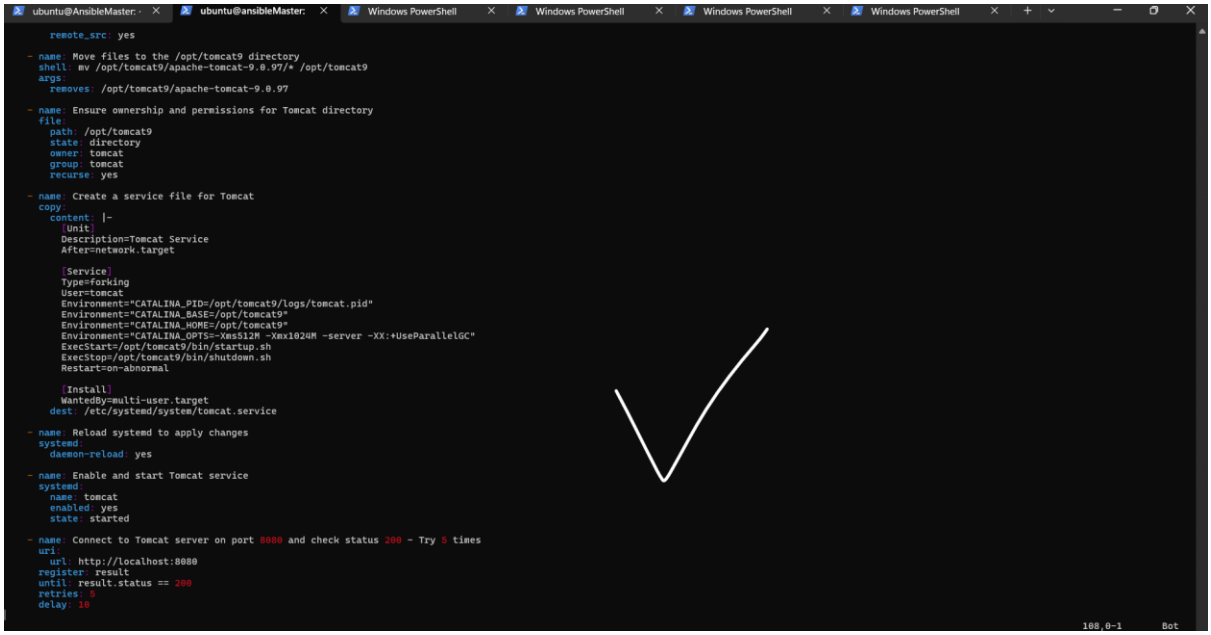


It’s accessible.

3) Write a ansible playbook to install apache tomcat.

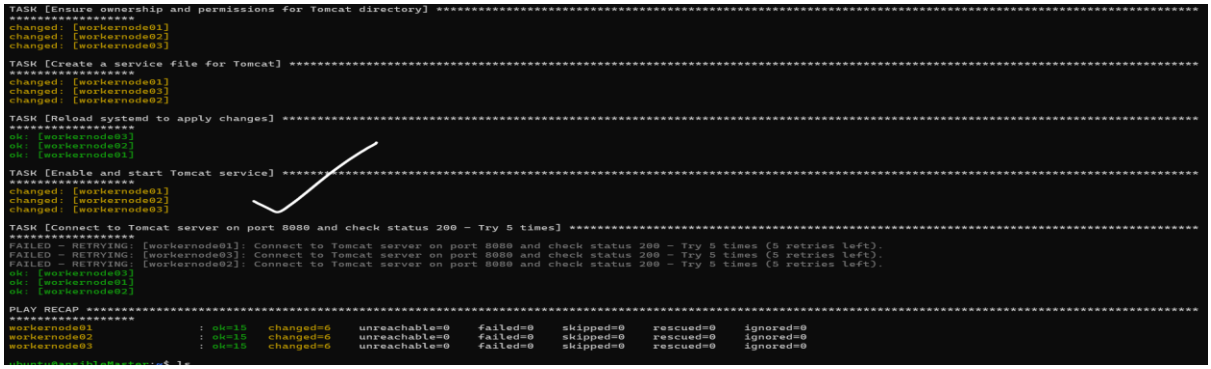
Here I am written a playbook to install tomcat.

```yaml
- name: Download and Install Tomcat9 from tomcat.apache.org
  hosts: all
  become: true
  vars:
    download_url: https://dlcdn.apache.org/tomcat/tomcat-9/v9.0.97/bin/apache-tomcat-9.0.97.tar.gz
    local_tomcat_path: /tmp/apache-tomcat-9.0.97.tar.gz
  tasks:
    - name: Install OpenJDK 17
      apt:
        name: openjdk-17-jre-headless
        update_cache: yes
        state: present

    - name: Validate if Java is available
      shell: java -version

    - name: Create Tomcat group
      group:
        name: tomcat
        state: present

    - name: Create Tomcat user
      user:
        name: tomcat
        group: tomcat
        state: present

    - name: Create a Directory /opt/tomcat9
      file:
        path: /opt/tomcat9
        state: directory
        mode: 0755
        owner: tomcat
        group: tomcat

    - name: Download Tomcat archive locally
      local_action:
        module: get_url
        url: "{{ download_url }}"
        dest: "{{ local_tomcat_path }}"
      delegate_to: localhost

    - name: Copy Tomcat archive to remote hosts
      copy:
        src: "{{ local_tomcat_path }}"
        dest: /tmp/apache-tomcat-9.0.97.tar.gz
        mode: 0755

    - name: Extract Tomcat archive
      unarchive:
        src: /tmp/apache-tomcat-9.0.97.tar.gz
        dest: /opt/tomcat9
        mode: 0755
        remote_src: yes
```



```yaml
      remote_src: yes

    - name: Move files to the /opt/tomcat9 directory
      shell: mv /opt/tomcat9/apache-tomcat-9.0.97/* /opt/tomcat9
      args:
        removes: /opt/tomcat9/apache-tomcat-9.0.97

    - name: Ensure ownership and permissions for Tomcat directory
      file:
        path: /opt/tomcat9
        state: directory
        owner: tomcat
        group: tomcat
        recurse: yes

    - name: Create a service file for Tomcat
      copy:
        content: |-
          [Unit]
          Description=Tomcat Service
          After=network.target

          [Service]
          Type=forking
          User=tomcat
          Environment="CATALINA_PID=/opt/tomcat9/logs/tomcat.pid"
          Environment="CATALINA_BASE=/opt/tomcat9"
          Environment="CATALINA_HOME=/opt/tomcat9"
          Environment="CATALINA_OPTS=-Xms512M -Xmx1024M -server -XX:+UseParallelGC"
          ExecStart=/opt/tomcat9/bin/startup.sh
          ExecStop=/opt/tomcat9/bin/shutdown.sh
          Restart=on-abnormal

          [Install]
          WantedBy=multi-user.target
        dest: /etc/systemd/system/tomcat.service

    - name: Reload systemd to apply changes
      systemd:
        daemon-reload: yes

    - name: Enable and start Tomcat service
      systemd:
        name: tomcat
        enabled: yes
        state: started

    - name: Connect to Tomcat server on port 8080 and check status 200 - Try 5 times
      uri:
        url: http://localhost:8080
      register: result
      until: result.status == 200
      retries: 5
      delay: 10
```

I run the play book that should executed.



```
TASK [Ensure ownership and permissions for Tomcat directory] ********************************************************************************************************************
changed: [workernode01]
changed: [workernode02]
changed: [workernode03]

TASK [Create a service file for Tomcat] *****************************************************************************************************************************************
changed: [workernode01]
changed: [workernode03]
changed: [workernode02]

TASK [Reload systemd to apply changes] ******************************************************************************************************************************************
ok: [workernode03]
ok: [workernode02]
ok: [workernode01]

TASK [Enable and start Tomcat service] ******************************************************************************************************************************************
changed: [workernode01]
changed: [workernode02]
changed: [workernode03]

TASK [Connect to Tomcat server on port 8080 and check status 200 - Try 5 times] *************************************************************************************************
FAILED - RETRYING: [workernode01]: Connect to Tomcat server on port 8080 and check status 200 - Try 5 times (5 retries left).
FAILED - RETRYING: [workernode03]: Connect to Tomcat server on port 8080 and check status 200 - Try 5 times (5 retries left).
FAILED - RETRYING: [workernode02]: Connect to Tomcat server on port 8080 and check status 200 - Try 5 times (5 retries left).
ok: [workernode01]
ok: [workernode03]
ok: [workernode02]

PLAY RECAP **********************************************************************************************************************************************************************
workernode01               : ok=15   changed=6    unreachable=0    failed=0    skipped=0    rescued=0    ignored=0
workernode02               : ok=15   changed=6    unreachable=0    failed=0    skipped=0    rescued=0    ignored=0
workernode03               : ok=15   changed=6    unreachable=0    failed=0    skipped=0    rescued=0    ignored=0
```

Now I want to check the all worker nodes the tomcat service is running or not and that should available on browser.

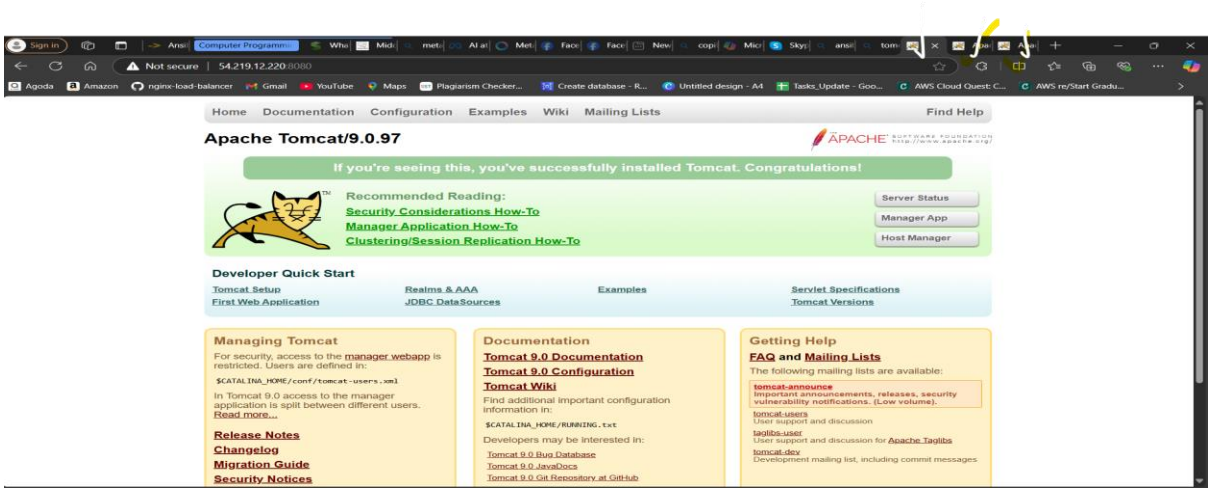
The below in three worker nodes tomcat is active.

```
ubuntu@workernode01:~$ sudo systemctl status tomcat
● tomcat.service - Tomcat Service
     Loaded: loaded (/etc/systemd/system/tomcat.service; enabled; preset: enabled)
     Active: active (running) since Wed 2024-11-13 18:15:42 UTC; 17min ago
    Process: 29707 ExecStart=/opt/tomcat9/bin/startup.sh (code=exited, status=0/SUCCESS)
   Main PID: 29717 (java)
      Tasks: 31 (limit: 1130)
     Memory: 222.6M (peak: 241.7M)
        CPU: 6.199s
     CGroup: /system.slice/tomcat.service
             └─29717 /usr/bin/java -Djava.util.logging.config.file=/opt/tomcat9/conf/logging.properties -Djava.util.logging.manager=o>

Nov 13 18:15:42 workernode01 systemd[1]: Starting tomcat.service - Tomcat Service...
Nov 13 18:15:42 workernode01 startup.sh[29707]: Tomcat started.
Nov 13 18:15:42 workernode01 systemd[1]: Started tomcat.service - Tomcat Service.
lines 1-14/14 (END)
```

```
ubuntu@workernode02:~$ sudo systemctl status tomcat
● tomcat.service - Tomcat Service
     Loaded: loaded (/etc/systemd/system/tomcat.service; enabled; preset: enabled)
     Active: active (running) since Wed 2024-11-13 18:15:42 UTC; 15min ago
    Process: 29495 ExecStart=/opt/tomcat9/bin/startup.sh (code=exited, status=0/SUCCESS)
   Main PID: 29505 (java)
      Tasks: 31 (limit: 1130)
     Memory: 216.7M (peak: 216.9M)
        CPU: 6.125s
     CGroup: /system.slice/tomcat.service
             └─29505 /usr/bin/java -Djava.util.logging.config.file=/opt/tomcat9/conf/logging.properties -Djava.util.lo

Nov 13 18:15:42 workernode02 systemd[1]: Starting tomcat.service - Tomcat Service...
Nov 13 18:15:42 workernode02 startup.sh[29495]: Tomcat started.
Nov 13 18:15:42 workernode02 systemd[1]: Started tomcat.service - Tomcat Service.
lines 1-14/14 (END)
```

```
ubuntu@workernode03:~$ sudo systemctl status tomcat
● tomcat.service - Tomcat Service
     Loaded: loaded (/etc/systemd/system/tomcat.service; enabled; preset: enabled)
     Active: active (running) since Wed 2024-11-13 18:15:42 UTC; 14min ago
    Process: 28390 ExecStart=/opt/tomcat9/bin/startup.sh (code=exited, status=0/SUCCESS)
   Main PID: 28400 (java)
      Tasks: 31 (limit: 1130)
     Memory: 221.0M (peak: 222.8M)
        CPU: 6.175s
     CGroup: /system.slice/tomcat.service
             └─28400 /usr/bin/java -Djava.util.logging.config.file=/opt/tomcat9/conf/logging.properties -Djava.util.logg

Nov 13 18:15:42 workernode03 systemd[1]: Starting tomcat.service - Tomcat Service...
Nov 13 18:15:42 workernode03 startup.sh[28390]: Tomcat started.
Nov 13 18:15:42 workernode03 systemd[1]: Started tomcat.service - Tomcat Service.
lines 1-14/14 (END)
```

Now I want to browsing.



Three servers also accessing the tomcat.

4) Write a ansible playbook to provision one ec2 on aws.

To crate ec2 instance through playbook in ansible.

Follow the below steps:

First you need to update the server.

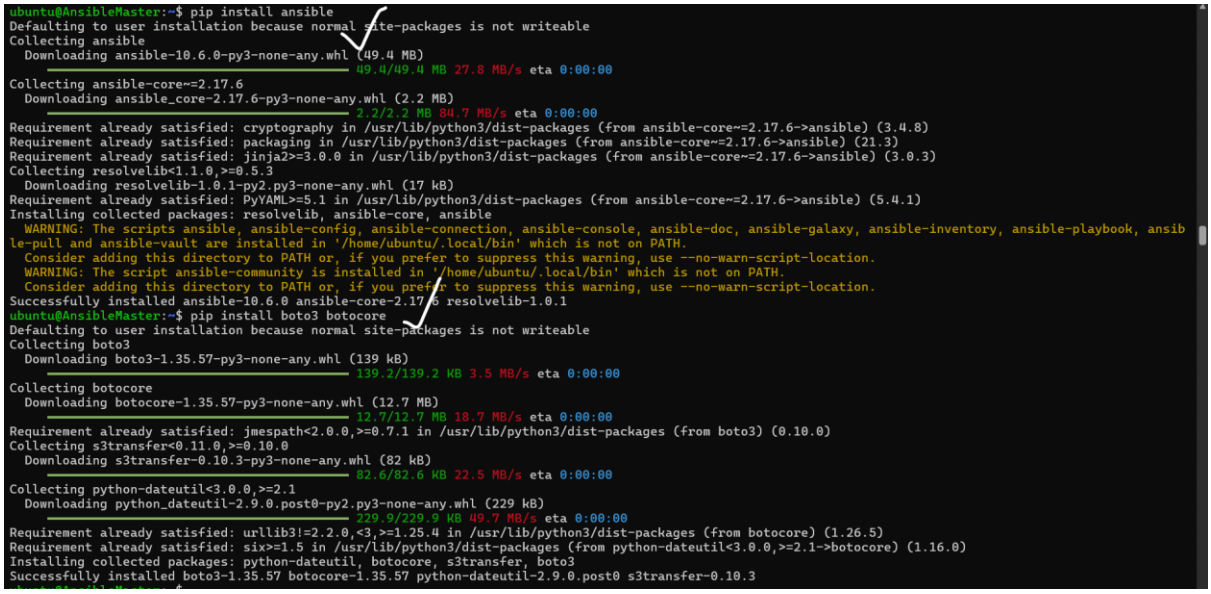
```
ubuntu@AnsibleMaster:~$ sudo apt update
Hit:1 http://us-west-1.ec2.archive.ubuntu.com/ubuntu jammy InRelease
Get:2 http://us-west-1.ec2.archive.ubuntu.com/ubuntu jammy-updates InRelease [128 kB]
Get:3 http://us-west-1.ec2.archive.ubuntu.com/ubuntu jammy-backports InRelease [127 kB]
Get:4 http://security.ubuntu.com/ubuntu jammy-security InRelease [129 kB]
Get:5 http://us-west-1.ec2.archive.ubuntu.com/ubuntu jammy/universe amd64 Packages [14.1 MB]
Get:6 http://us-west-1.ec2.archive.ubuntu.com/ubuntu jammy/universe Translation-en [5652 kB]
Get:7 http://us-west-1.ec2.archive.ubuntu.com/ubuntu jammy/universe amd64 c-n-f Metadata [286 kB]
Get:8 http://us-west-1.ec2.archive.ubuntu.com/ubuntu jammy/multiverse amd64 Packages [217 kB]
Get:9 http://us-west-1.ec2.archive.ubuntu.com/ubuntu jammy/multiverse Translation-en [112 kB]
Get:10 http://us-west-1.ec2.archive.ubuntu.com/ubuntu jammy/multiverse amd64 c-n-f Metadata [8372 B]
Get:11 http://us-west-1.ec2.archive.ubuntu.com/ubuntu jammy-updates/main amd64 Packages [2146 kB]
Get:12 http://security.ubuntu.com/ubuntu jammy-security/main amd64 Packages [1911 kB]
Get:13 http://us-west-1.ec2.archive.ubuntu.com/ubuntu jammy-updates/main Translation-en [367 kB]
Get:14 http://us-west-1.ec2.archive.ubuntu.com/ubuntu jammy-updates/main amd64 c-n-f Metadata [17.9 kB]
Get:15 http://us-west-1.ec2.archive.ubuntu.com/ubuntu jammy-updates/restricted amd64 Packages [2633 kB]
Get:16 http://us-west-1.ec2.archive.ubuntu.com/ubuntu jammy-updates/restricted Translation-en [455 kB]
Get:17 http://us-west-1.ec2.archive.ubuntu.com/ubuntu jammy-updates/restricted amd64 c-n-f Metadata [612 B]
Get:18 http://us-west-1.ec2.archive.ubuntu.com/ubuntu jammy-updates/universe amd64 Packages [1134 kB]
Get:19 http://us-west-1.ec2.archive.ubuntu.com/ubuntu jammy-updates/universe Translation-en [265 kB]
Get:20 http://us-west-1.ec2.archive.ubuntu.com/ubuntu jammy-updates/universe amd64 c-n-f Metadata [26.4 kB]
Get:21 http://us-west-1.ec2.archive.ubuntu.com/ubuntu jammy-updates/multiverse amd64 Packages [43.3 kB]
Get:22 http://us-west-1.ec2.archive.ubuntu.com/ubuntu jammy-updates/multiverse Translation-en [10.8 kB]
Get:23 http://us-west-1.ec2.archive.ubuntu.com/ubuntu jammy-updates/multiverse amd64 c-n-f Metadata [440 B]
Get:24 http://us-west-1.ec2.archive.ubuntu.com/ubuntu jammy-backports/main amd64 Packages [67.7 kB]
Get:25 http://us-west-1.ec2.archive.ubuntu.com/ubuntu jammy-backports/main Translation-en [11.1 kB]
Get:26 http://us-west-1.ec2.archive.ubuntu.com/ubuntu jammy-backports/main amd64 c-n-f Metadata [388 B]
Get:27 http://us-west-1.ec2.archive.ubuntu.com/ubuntu jammy-backports/restricted amd64 c-n-f Metadata [116 B]
Get:28 http://us-west-1.ec2.archive.ubuntu.com/ubuntu jammy-backports/universe amd64 Packages [28.8 kB]
Get:29 http://us-west-1.ec2.archive.ubuntu.com/ubuntu jammy-backports/universe Translation-en [16.5 kB]
Get:30 http://us-west-1.ec2.archive.ubuntu.com/ubuntu jammy-backports/universe amd64 c-n-f Metadata [672 B]
Get:31 http://us-west-1.ec2.archive.ubuntu.com/ubuntu jammy-backports/multiverse amd64 c-n-f Metadata [116 B]
Get:32 http://security.ubuntu.com/ubuntu jammy-security/main Translation-en [307 kB]
Get:33 http://security.ubuntu.com/ubuntu jammy-security/main amd64 c-n-f Metadata [13.3 kB]
Get:34 http://security.ubuntu.com/ubuntu jammy-security/restricted amd64 Packages [2545 kB]
Get:35 http://security.ubuntu.com/ubuntu jammy-security/restricted Translation-en [440 kB]
```

CMD: sudo apt install python3-pip

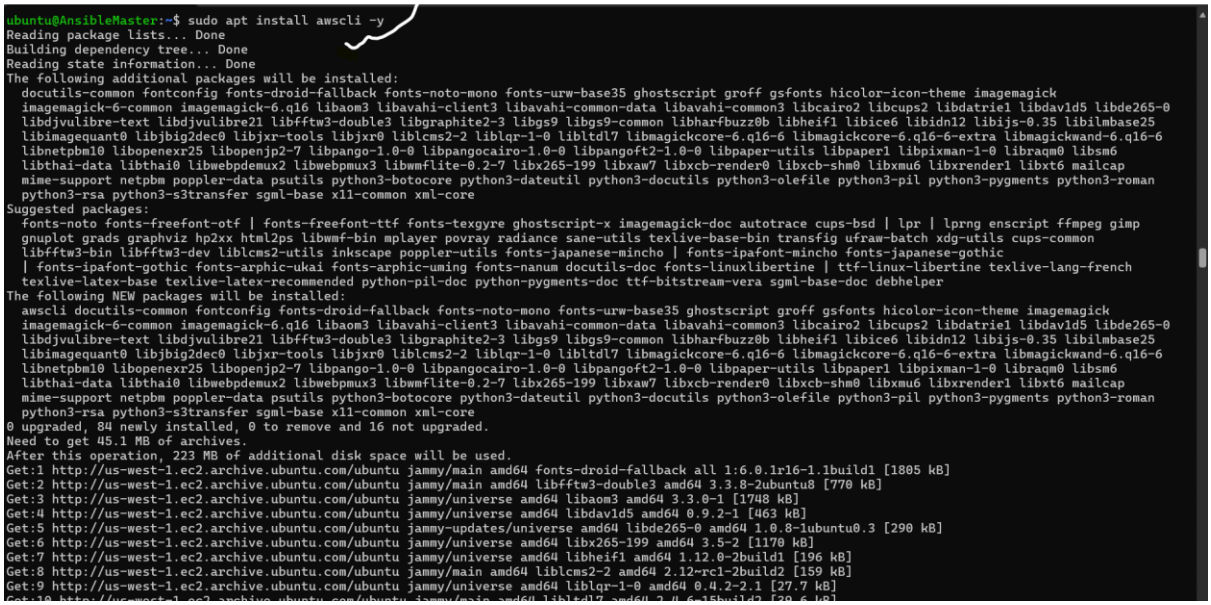
```
ubuntu@AnsibleMaster:~$ sudo apt install python3-pip
Reading package lists... Done
Building dependency tree... Done
Reading state information... Done
The following additional packages will be installed:
  build-essential bzip2 cpp cpp-11 dpkg-dev fakeroot fontconfig-config fonts-dejavu-core g++ g++-11 gcc gcc-11 gcc-11-base javascript-common
  libalgorithm-diff-perl libalgorithm-diff-xs-perl libalgorithm-merge-perl libasan6 libatomic1 libc-dev-bin libc-devtools libc6-dev libcc1-0 libcrypt-dev
  libdeflate0 libdpkg-perl libexpat1-dev libfakeroot libfile-fcntllock-perl libfontconfig1 libgcc-11-dev libgd3 libgomp1 libisl23 libitm1 libjbig0
  libjpeg-turbo8 libjpeg8 libjs-jquery libjs-sphinxdoc libjs-underscore liblsan0 libmpc3 libnsl-dev libpython3-dev libpython3.10-dev libquadmath0
  libstdc++-11-dev libtiff5 libtirpc-dev libtsan0 libubsan1 libwebp7 libxpm4 linux-libc-dev lto-disabled-list make manpages-dev python3-dev python3-wheel
  python3.10-dev rpcsvc-proto zlib1g-dev
Suggested packages:
  bzip2-doc cpp-doc gcc-11-locales debian-keyring g++-multilib g++-11-multilib gcc-11-doc gcc-multilib autoconf automake libtool flex bison gdb gcc-doc
  gcc-11-multilib apache2 | lighttpd | httpd glibc-doc bzr libgd-tools libstdc++-11-doc make-doc
The following NEW packages will be installed:
  build-essential bzip2 cpp cpp-11 dpkg-dev fakeroot fontconfig-config fonts-dejavu-core g++ g++-11 gcc gcc-11 gcc-11-base javascript-common
  libalgorithm-diff-perl libalgorithm-diff-xs-perl libalgorithm-merge-perl libasan6 libatomic1 libc-dev-bin libc-devtools libc6-dev libcc1-0 libcrypt-dev
  libdeflate0 libdpkg-perl libexpat1-dev libfakeroot libfile-fcntllock-perl libfontconfig1 libgcc-11-dev libgd3 libgomp1 libisl23 libitm1 libjbig0
  libjpeg-turbo8 libjpeg8 libjs-jquery libjs-sphinxdoc libjs-underscore liblsan0 libmpc3 libnsl-dev libpython3-dev libpython3.10-dev libquadmath0
  libstdc++-11-dev libtiff5 libtirpc-dev libtsan0 libubsan1 libwebp7 libxpm4 linux-libc-dev lto-disabled-list make manpages-dev python3-dev python3-pip
  python3-wheel python3.10-dev rpcsvc-proto zlib1g-dev
0 upgraded, 64 newly installed, 0 to remove and 16 not upgraded.
Need to get 71.3 MB of archives.
After this operation, 239 MB of additional disk space will be used.
Do you want to continue? [Y/n] y
Get:1 http://us-west-1.ec2.archive.ubuntu.com/ubuntu jammy-updates/main amd64 libc-dev-bin amd64 2.35-0ubuntu3.8 [20.3 kB]
Get:2 http://us-west-1.ec2.archive.ubuntu.com/ubuntu jammy-updates/main amd64 linux-libc-dev amd64 5.15.0-125.135 [1345 kB]
Get:3 http://us-west-1.ec2.archive.ubuntu.com/ubuntu jammy/main amd64 libcrypt-dev amd64 1:4.4.27-1 [112 kB]
Get:4 http://us-west-1.ec2.archive.ubuntu.com/ubuntu jammy/main amd64 rpcsvc-proto amd64 1.4.2-0ubuntu6 [68.5 kB]
Get:5 http://us-west-1.ec2.archive.ubuntu.com/ubuntu jammy-updates/main amd64 libtirpc-dev amd64 1.3.2-2ubuntu0.1 [192 kB]
Get:6 http://us-west-1.ec2.archive.ubuntu.com/ubuntu jammy/main amd64 libnsl-dev amd64 1.3.0-2build2 [71.3 kB]
Get:7 http://us-west-1.ec2.archive.ubuntu.com/ubuntu jammy-updates/main amd64 libc6-dev amd64 2.35-0ubuntu3.8 [2100 kB]
Get:8 http://us-west-1.ec2.archive.ubuntu.com/ubuntu jammy-updates/main amd64 gcc-11-base amd64 11.4.0-1ubuntu1~22.04 [20.2 kB]
Get:9 http://us-west-1.ec2.archive.ubuntu.com/ubuntu jammy/main amd64 libisl23 amd64 0.24-2build1 [727 kB]
Get:10 http://us-west-1.ec2.archive.ubuntu.com/ubuntu jammy/main amd64 libmpc3 amd64 1.2.1-2build1 [46.9 kB]
Get:11 http://us-west-1.ec2.archive.ubuntu.com/ubuntu jammy-updates/main amd64 cpp-11 amd64 11.4.0-1ubuntu1~22.04 [10.0 MB]
Get:12 http://us-west-1.ec2.archive.ubuntu.com/ubuntu jammy/main amd64 cpp amd64 4:11.2.0-1ubuntu1 [27.7 kB]
Get:13 http://us-west-1.ec2.archive.ubuntu.com/ubuntu jammy-updates/main amd64 libcc1-0 amd64 12.3.0-1ubuntu1~22.04 [48.3 kB]
Get:14 http://us-west-1.ec2.archive.ubuntu.com/ubuntu jammy-updates/main amd64 libgomp1 amd64 12.3.0-1ubuntu1~22.04 [126 kB]
Get:15 http://us-west-1.ec2.archive.ubuntu.com/ubuntu jammy-updates/main amd64 libitm1 amd64 12.3.0-1ubuntu1~22.04 [30.2 kB]
```

CMD: pip install ansible

CMD: pip install boto3 botocore



CMD: sudo apt install awscli –y



Then you need to configure the aws credentials.

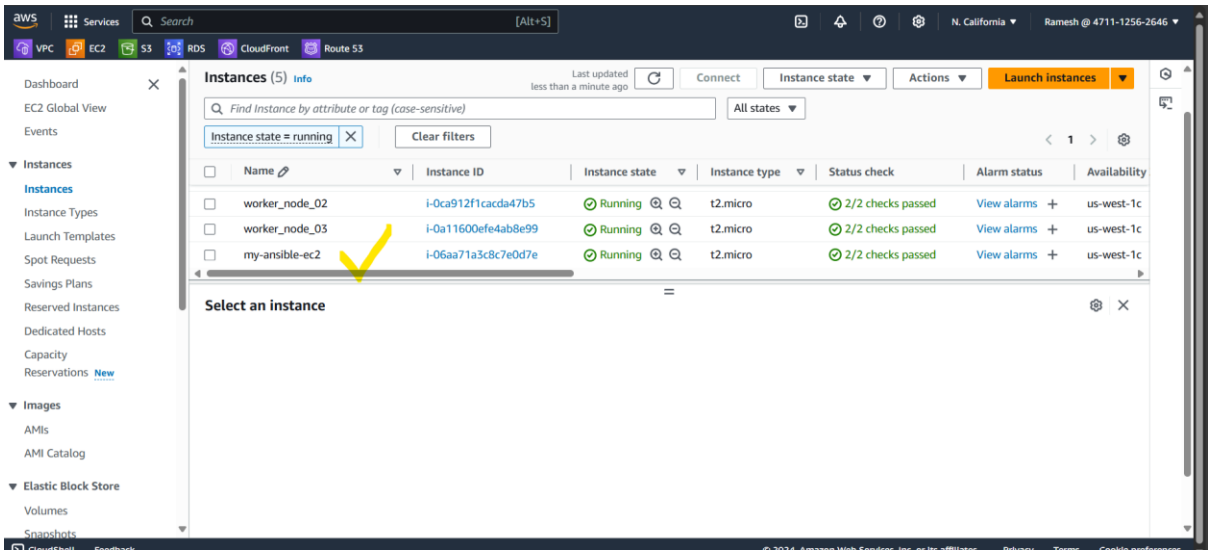
Then now you want to create .yml for create ec2 instance.

```yaml
---
- name: Launch EC2 Instance
  hosts: localhost
  gather_facts: no
  tasks:
    - name: Launch new EC2 instance
      amazon.aws.ec2_instance:
        name: "my-ansible-ec2"
        key_name: "Ncalifornia"
        instance_type: "t2.micro"
        image_id: "ami-05c65d8bb2e35991a"  # Example for Amazon Linux 2, check your region for valid AMI
        region: "us-west-1"
        wait: yes
        vpc_subnet_id: "subnet-03324022d15e30926"
        security_groups: "sg-0036f38891ec04f38"
      register: ec2

    - name: Output Instance ID
      debug:
        msg: "Launched instance ID is {{ ec2.instance_ids[0] }}"
```

Now our ec2 instance is created.

```
ubuntu@AnsibleMaster:~$ ansible-playbook created_ec2.yml

PLAY [Launch EC2 Instance] ***************************************************************************************************************************************************

TASK [Launch new EC2 instance] ***********************************************************************************************************************************************
changed: [localhost]

TASK [Output Instance ID] ****************************************************************************************************************************************************
ok: [localhost] => {
    "msg": "Launched instance ID is i-06aa71a3c8c7e0d7e"
}

PLAY RECAP *******************************************************************************************************************************************************************
localhost                  : ok=2    changed=1    unreachable=0    failed=0    skipped=0    rescued=0    ignored=0
```

Now I am check in aws console ec2 instance is created or not.



Playbook explanation:

Name: Describes the playbook's purpose, "Launch EC2 Instance".

hosts: localhost: Runs tasks on the control machine, as we are managing AWS resources remotely.

gather_facts: no: Disables unnecessary fact-gathering to save time.

Purpose: This task launches an EC2 instance with the specified configuration.
Key Parameters:

- name, key_name, instance_type, and image_id define the instance name, SSH key, instance type, and AMI.
- region, vpc_subnet_id, and security_groups set the AWS region, subnet, and security group for network and access configurations.
- register: ec2 saves instance data for later tasks.

- **Purpose**: Prints the ID of the newly created EC2 instance.
- **debug Module**: Outputs a custom message using the instance ID stored in ec2.instance_ids[0].

5) Write a ansible playbook to copy one file from node-1 to node-2.

First I login into the workernode01 there I am created one file added the content in file.

```
ubuntu@workernode01:~$ cat > workernode1.txt
this file is created in workernode_01.
^C
ubuntu@workernode01:~$ ls
workernode1.txt
ubuntu@workernode01:~$ cat workernode1.txt
this file is created in workernode_01.
ubuntu@workernode01:~$ pwd
/home/ubuntu
ubuntu@workernode01:~$
```

Now this file I want to transfer to workernode02.

In workernode02 there is no file as off now.

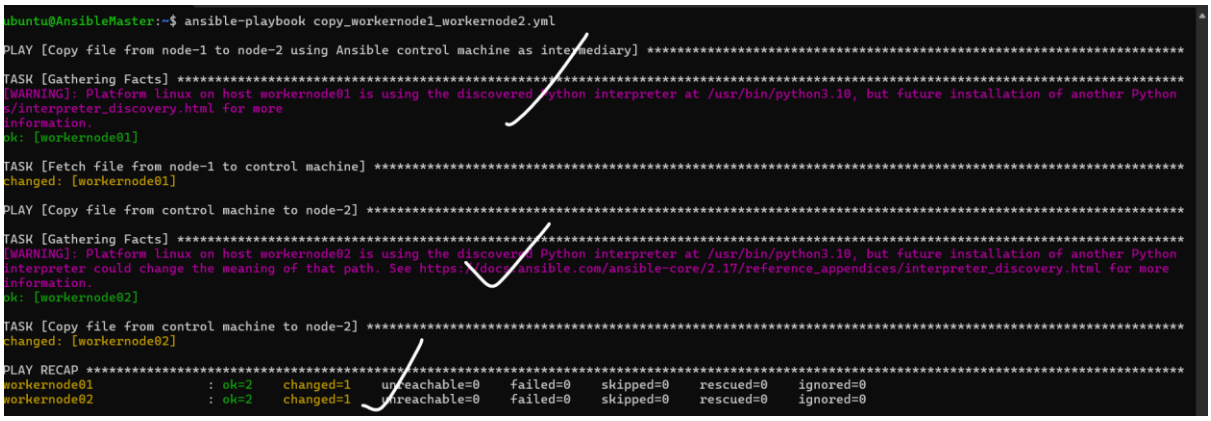
```
Last login: Mon Nov 11 09:18:49 2024 from 172.31.12.169
ubuntu@workernode02:~$ ls
ubuntu@workernode02:~$ pwd
/home/ubuntu
ubuntu@workernode02:~$
```

So I went to ansibleMaster server there I am writing .ymal file.

Through .yml we need to send workernode02.

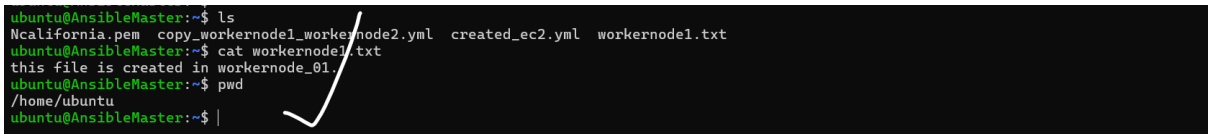
```
ubuntu@AnsibleMaster:~$ sudo cat /etc/ansible/hosts
[all]
workernode01 ansible_host=172.31.7.86 ansible_user=ubuntu ansible_ssh_private_key_file=/home/ubuntu/Ncalifornia.pem
workernode02 ansible_host=172.31.13.180 ansible_user=ubuntu ansible_ssh_private_key_file=/home/ubuntu/Ncalifornia.pem
workernode03 ansible_host=172.31.10.242 ansible_user=ubuntu ansible_ssh_private_key_file=/home/ubuntu/Ncalifornia.pem

ubuntu@AnsibleMaster:~$ cat copy_workernode1_workernode2.yml
---
- name: Copy file from node-1 to node-2 using Ansible control machine as intermediary
  hosts: workernode01
  tasks:
    - name: Fetch file from node-1 to control machine
      fetch:
        src: /home/ubuntu/workernode1.txt  # Source file on node-1
        dest: /home/ubuntu/workernode1.txt  # Temporary location on the control machine
        flat: yes                # Avoids creating a directory structure

- name: Copy file from control machine to node-2
  hosts: workernode02
  tasks:
    - name: Copy file from control machine to node-2
      copy:
        src: /home/ubuntu/workernode1.txt # Source file on the control machine (master server)
        dest: /home/ubuntu  # Destination path on node-2

ubuntu@AnsibleMaster:~$
```

The playbook is successfully executed.



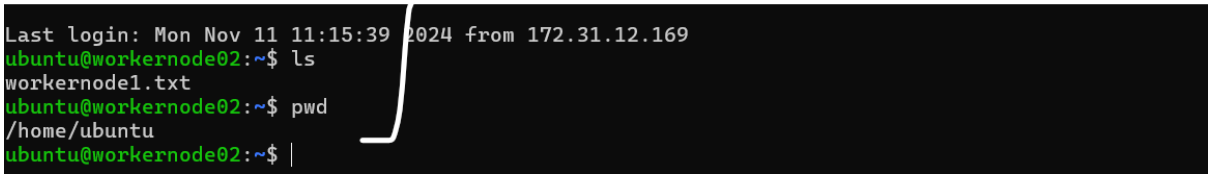
Now I want to check file is there or not.

The below one is Ansible Master server.



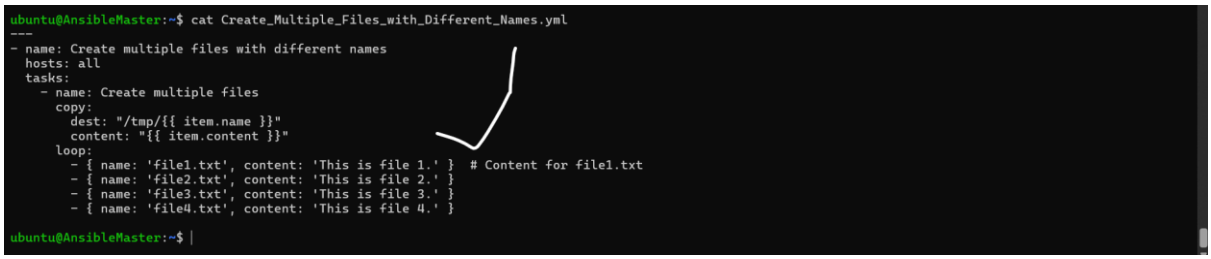
This is my workernode02.

Content is available.



6) Write a ansible playbook to create different files with different names using single playbook.

Here first created one .yml file there I am written one script to crate multiple ec2 instances.



```
ubuntu@AnsibleMaster:~$ sudo vi Create_Multiple_Files_with_Different_Names.yml
ubuntu@AnsibleMaster:~$ ansible-playbook Create_Multiple_Files_with_Different_Names.yml

PLAY [Create multiple files with different names] ***********************************************************************************************************

TASK [Gathering Facts] **************************************************************************************************************************************
[WARNING]: Platform linux on host workernode01 is using the discovered Python interpreter at /usr/bin/python3.10, but future installation of another Python
interpreter could change the meaning of that path. See https://docs.ansible.com/ansible-core/2.17/reference_appendices/interpreter_discovery.html for more
information.
ok: [workernode01]
[WARNING]: Platform linux on host workernode02 is using the discovered Python interpreter at /usr/bin/python3.10, but future installation of another Python
interpreter could change the meaning of that path. See https://docs.ansible.com/ansible-core/2.17/reference_appendices/interpreter_discovery.html for more
information.
ok: [workernode02]
[WARNING]: Platform linux on host workernode03 is using the discovered Python interpreter at /usr/bin/python3.10, but future installation of another Python
interpreter could change the meaning of that path. See https://docs.ansible.com/ansible-core/2.17/reference_appendices/interpreter_discovery.html for more
information.
ok: [workernode03]

TASK [Create multiple files] ********************************************************************************************************************************
changed: [workernode01] => (item={'name': 'file1.txt', 'content': 'This is file 1.'})
changed: [workernode03] => (item={'name': 'file1.txt', 'content': 'This is file 1.'})
changed: [workernode02] => (item={'name': 'file1.txt', 'content': 'This is file 1.'})
changed: [workernode01] => (item={'name': 'file2.txt', 'content': 'This is file 2.'})
changed: [workernode03] => (item={'name': 'file2.txt', 'content': 'This is file 2.'})
changed: [workernode02] => (item={'name': 'file2.txt', 'content': 'This is file 2.'})
changed: [workernode01] => (item={'name': 'file3.txt', 'content': 'This is file 3.'})
changed: [workernode03] => (item={'name': 'file3.txt', 'content': 'This is file 3.'})
changed: [workernode02] => (item={'name': 'file3.txt', 'content': 'This is file 3.'})
changed: [workernode01] => (item={'name': 'file4.txt', 'content': 'This is file 4.'})
changed: [workernode02] => (item={'name': 'file4.txt', 'content': 'This is file 4.'})
changed: [workernode03] => (item={'name': 'file4.txt', 'content': 'This is file 4.'})

PLAY RECAP **************************************************************************************************************************************************
workernode01               : ok=2    changed=1    unreachable=0    failed=0    skipped=0    rescued=0    ignored=0
workernode02               : ok=2    changed=1    unreachable=0    failed=0    skipped=0    rescued=0    ignored=0
workernode03               : ok=2    changed=1    unreachable=0    failed=0    skipped=0    rescued=0    ignored=0

ubuntu@AnsibleMaster:~$
```

Within the three servers we have all files.

```
ubuntu@workernode01:~$ ls
workernode1.txt
ubuntu@workernode01:~$ cd /tmp/
ubuntu@workernode01:/tmp$ ls
file1.txt  snap-private-tmp
file2.txt  systemd-private-0a97440377c148bdb0ad80ce98d5250a-chrony.service-8L2xZ1
file3.txt  systemd-private-0a97440377c148bdb0ad80ce98d5250a-systemd-logind.service-4EgLir
file4.txt  systemd-private-0a97440377c148bdb0ad80ce98d5250a-systemd-resolved.service-0JmNhR
ubuntu@workernode01:/tmp$
```

```
ubuntu@workernode02:~$ cd /tmp/
ubuntu@workernode02:/tmp$ ls
file1.txt  snap-private-tmp
file2.txt  systemd-private-4f67fdf226b14790a440e22e7ca3a672-chrony.service-DvJhCr
file3.txt  systemd-private-4f67fdf226b14790a440e22e7ca3a672-systemd-logind.service-Rgibl4
file4.txt  systemd-private-4f67fdf226b14790a440e22e7ca3a672-systemd-resolved.service-5jPvZ3
ubuntu@workernode02:/tmp$
```

```
Last login: Mon Nov 11 11:58:30 2024 from 172.31.12.169
ubuntu@workernode03:~$ ls
ubuntu@workernode03:~$ cd /tmp
ubuntu@workernode03:/tmp$ ls
file1.txt  snap-private-tmp
file2.txt  systemd-private-d2f207f19fff46c786c97d2fd6b78903-chrony.service-4yCGg7
file3.txt  systemd-private-d2f207f19fff46c786c97d2fd6b78903-systemd-logind.service-fyDmNw
file4.txt  systemd-private-d2f207f19fff46c786c97d2fd6b78903-systemd-resolved.service-H49YEh
ubuntu@workernode03:/tmp$
```

Tocmact file code is below:

```
- name: Download and Install Tomcat9 from tomcat.apache.org
  hosts: all
  become: true
  vars:
    download_url: https://dlcdn.apache.org/tomcat/tomcat-9/v9.0.97/bin/apache-tomcat-9.0.97.tar.gz
    local_tomcat_path: /tmp/apache-tomcat-9.0.97.tar.gz
  tasks:
    - name: Install OpenJDK 17
      apt:
        name: openjdk-17-jre-headless
        update_cache: yes
        state: present

    - name: Validate if Java is available
      shell: java -version

    - name: Create Tomcat group
      group:
        name: tomcat
        state: present

    - name: Create Tomcat user
      user:
        name: tomcat
        group: tomcat
        state: present

    - name: Create a Directory /opt/tomcat9
      file:
        path: /opt/tomcat9
        state: directory
```

```
    mode: 0755
    owner: tomcat
    group: tomcat

- name: Download Tomcat archive locally
  local_action:
    module: get_url
    url: "{{ download_url }}"
    dest: "{{ local_tomcat_path }}"
  delegate_to: localhost

- name: Copy Tomcat archive to remote hosts
  copy:
    src: "{{ local_tomcat_path }}"
    dest: /tmp/apache-tomcat-9.0.97.tar.gz
    mode: 0755

- name: Extract Tomcat archive
  unarchive:
    src: /tmp/apache-tomcat-9.0.97.tar.gz
    dest: /opt/tomcat9
    mode: 0755
    remote_src: yes

- name: Move files to the /opt/tomcat9 directory
  shell: mv /opt/tomcat9/apache-tomcat-9.0.97/* /opt/tomcat9
  args:
    removes: /opt/tomcat9/apache-tomcat-9.0.97

- name: Ensure ownership and permissions for Tomcat directory
  file:
    path: /opt/tomcat9
```

```
    state: directory
    owner: tomcat
    group: tomcat
    recurse: yes

- name: Create a service file for Tomcat
  copy:
    content: |-
      [Unit]
      Description=Tomcat Service
      After=network.target

      [Service]
      Type=forking
      User=tomcat
      Environment="CATALINA_PID=/opt/tomcat9/logs/tomcat.pid"
      Environment="CATALINA_BASE=/opt/tomcat9"
      Environment="CATALINA_HOME=/opt/tomcat9"
      Environment="CATALINA_OPTS=-Xms512M -Xmx1024M -server -XX:+UseParallelGC"
      ExecStart=/opt/tomcat9/bin/startup.sh
      ExecStop=/opt/tomcat9/bin/shutdown.sh
      Restart=on-abnormal

      [Install]
      WantedBy=multi-user.target
    dest: /etc/systemd/system/tomcat.service

- name: Reload systemd to apply changes
  systemd:
    daemon-reload: yes

- name: Enable and start Tomcat service
```

```
  systemd:
    name: tomcat
    enabled: yes
    state: started

- name: Connect to Tomcat server on port 8080 and check status 200 - Try 5 times
  uri:
    url: http://localhost:8080
  register: result
  until: result.status == 200
  retries: 5
  delay: 10
```