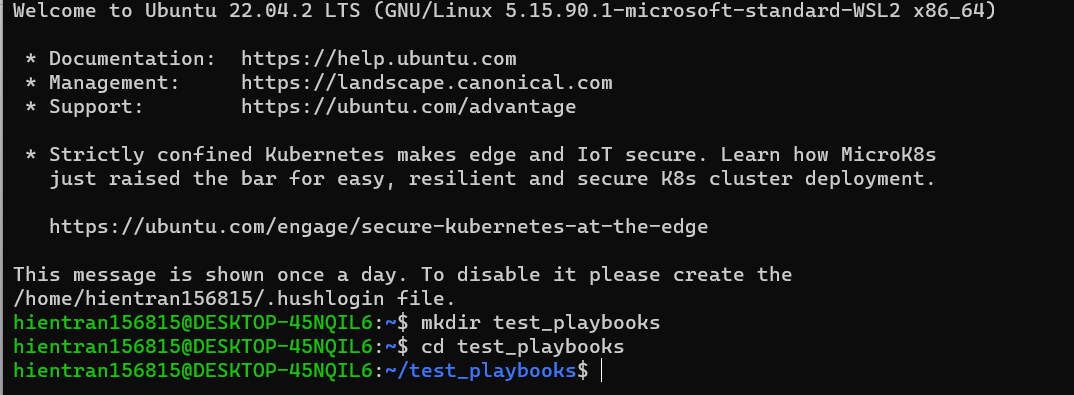
Ansible Playbooks

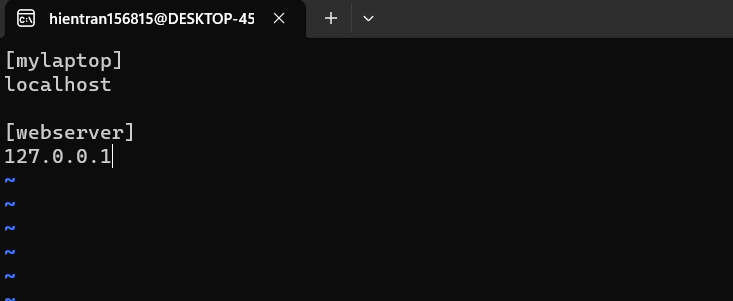
1. Create a new directory where we can create our playbook



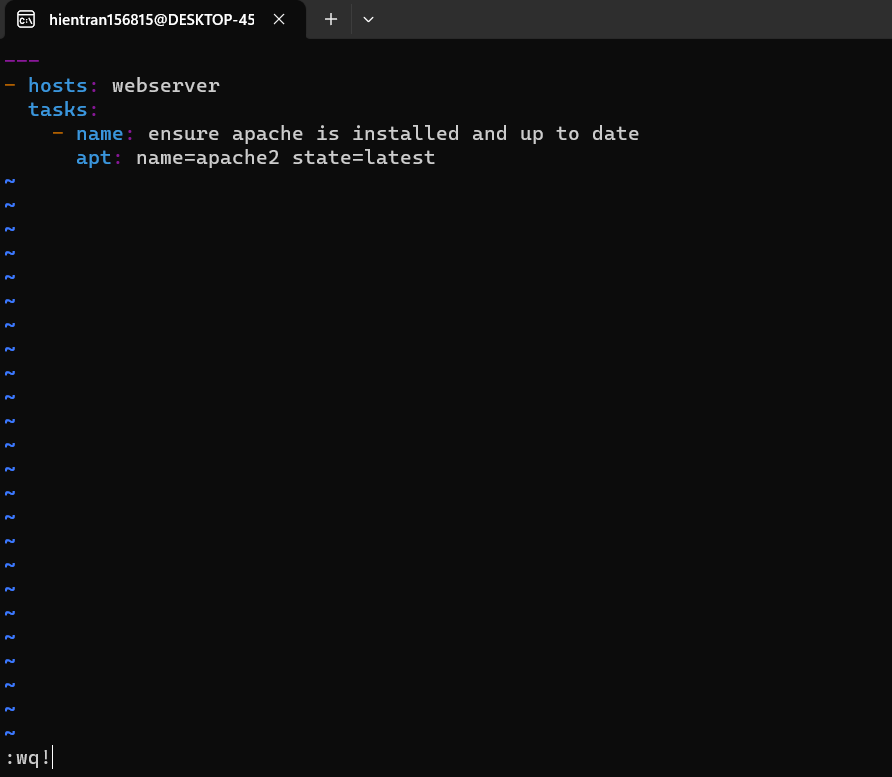
1. Create a new hosts file in the directory

touch hosts

vi hosts



1. Create a new file webserver-playbook.yml



**---**

**- hosts: webserver**

**tasks:**

**- name: ensure apache is installed and up to date**

**apt: name=apache2 state=latest**

1. Obtain the copy of a default Apache2 configuration file

**wget** [**https://raw.githubusercontent.com/vincesesto/practical\_ansible\_ed2/main/chapter\_2/test\_playbooks/000-default.conf**](https://raw.githubusercontent.com/vincesesto/practical_ansible_ed2/main/chapter_2/test_playbooks/000-default.conf)

1. Copy module to add the configuration file to our webserver-playbook.yml

A notify action will ensure if the conf file ever changes, Apache2 will be restarted

1. Make sure Apache2 service is running and enable it to start up when the system is booted
2. Make sure Apache2 is restarted if there are any changes
3. Set up the handlers to respond to the “restart apache”

**---**

**- hosts: webserver**

**tasks:**

**- name: ensure apache is installed and up to date**

**apt: name=apache2 state=latest**

**- name: write the apache config file**

**copy: src=000-default.conf dest=/etc/apache2/sites-available/000-default.conf**

**notify:**

**- restart apache2**

**- name: apache is running (and enable it at boot)**

**service: name=apache2 state=started enabled=yes**

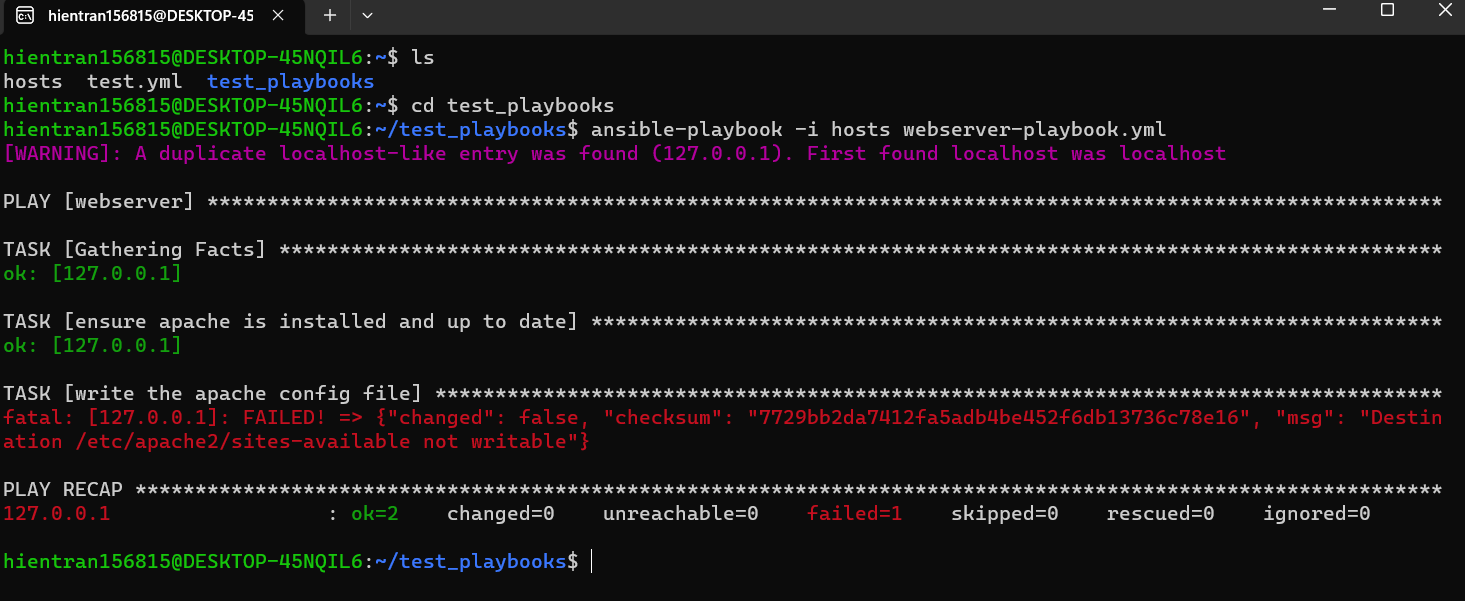
**handlers:**

**- name: restart apache2**

**service: name=apache2 state=restarted**

1. Run the playbook

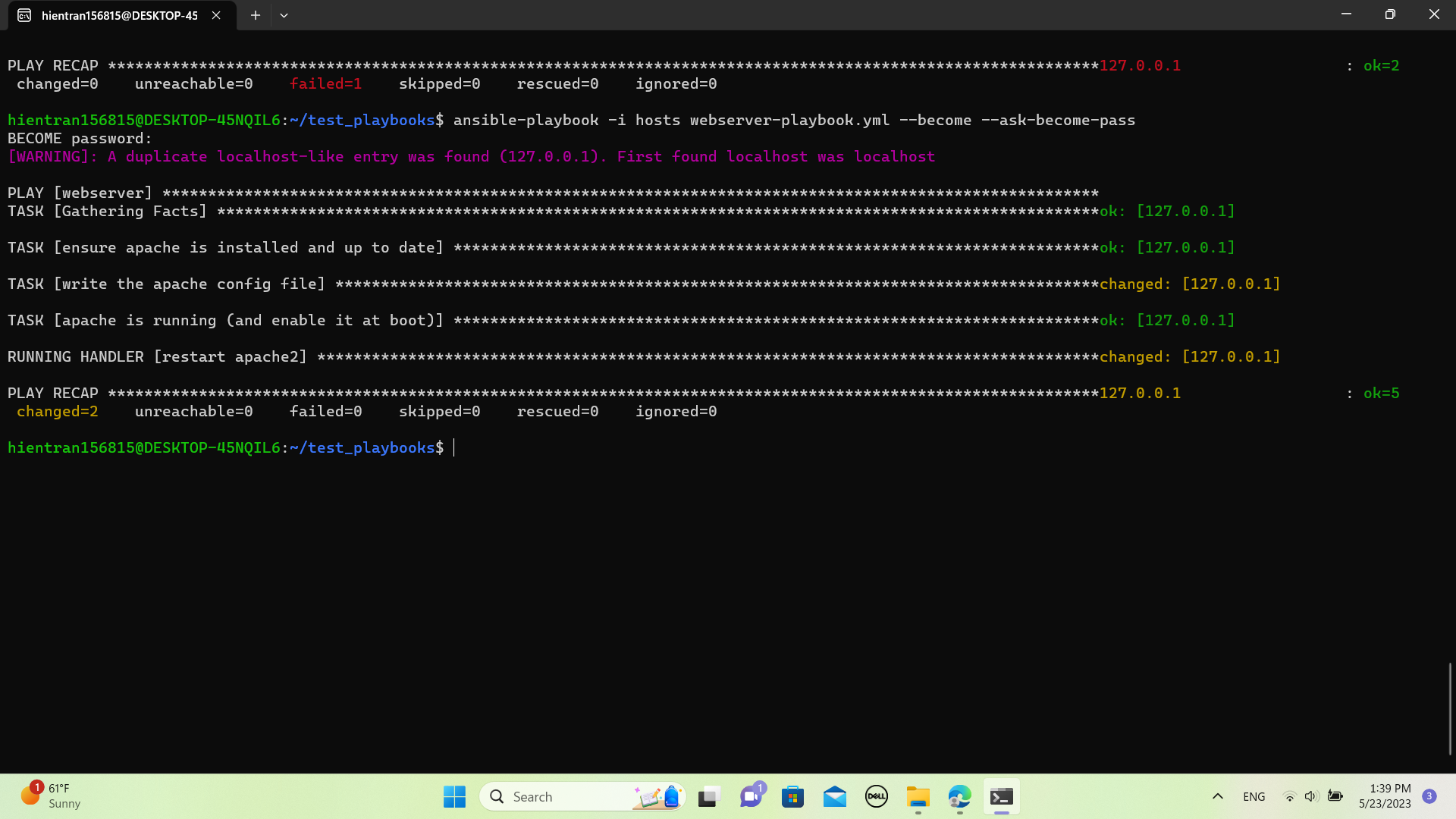
**ansible-playbook -i hosts webserver-playbook.yml**



**TROUBLESHOOT**

**ansible-playbook -i hosts webserver-playbook.yml --become --ask-become-**

**pass**

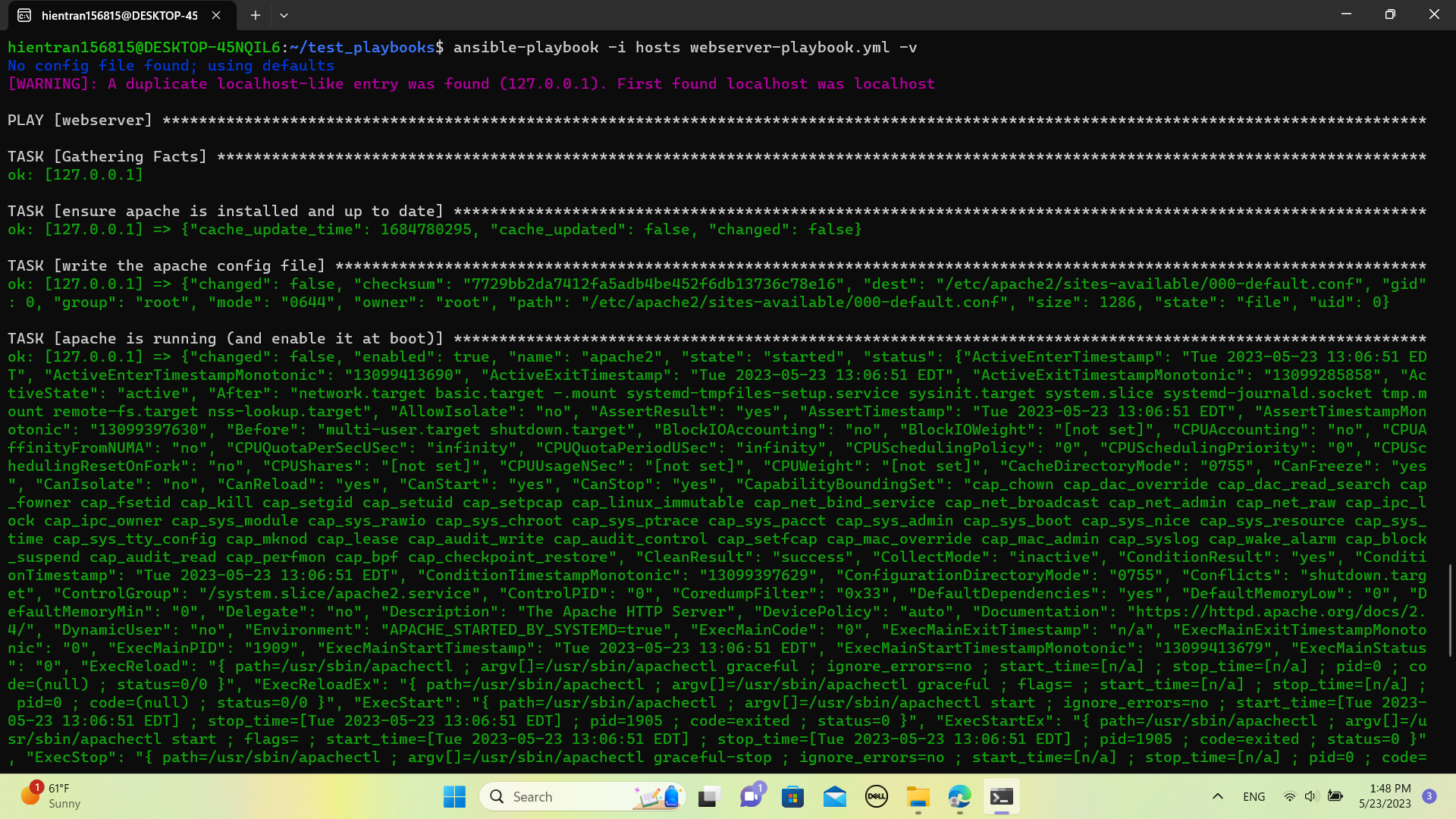
****

1. **Verify**

[**http://0.0.0.0**](http://0.0.0.0)

**To see more output printed out**

Add **-v**

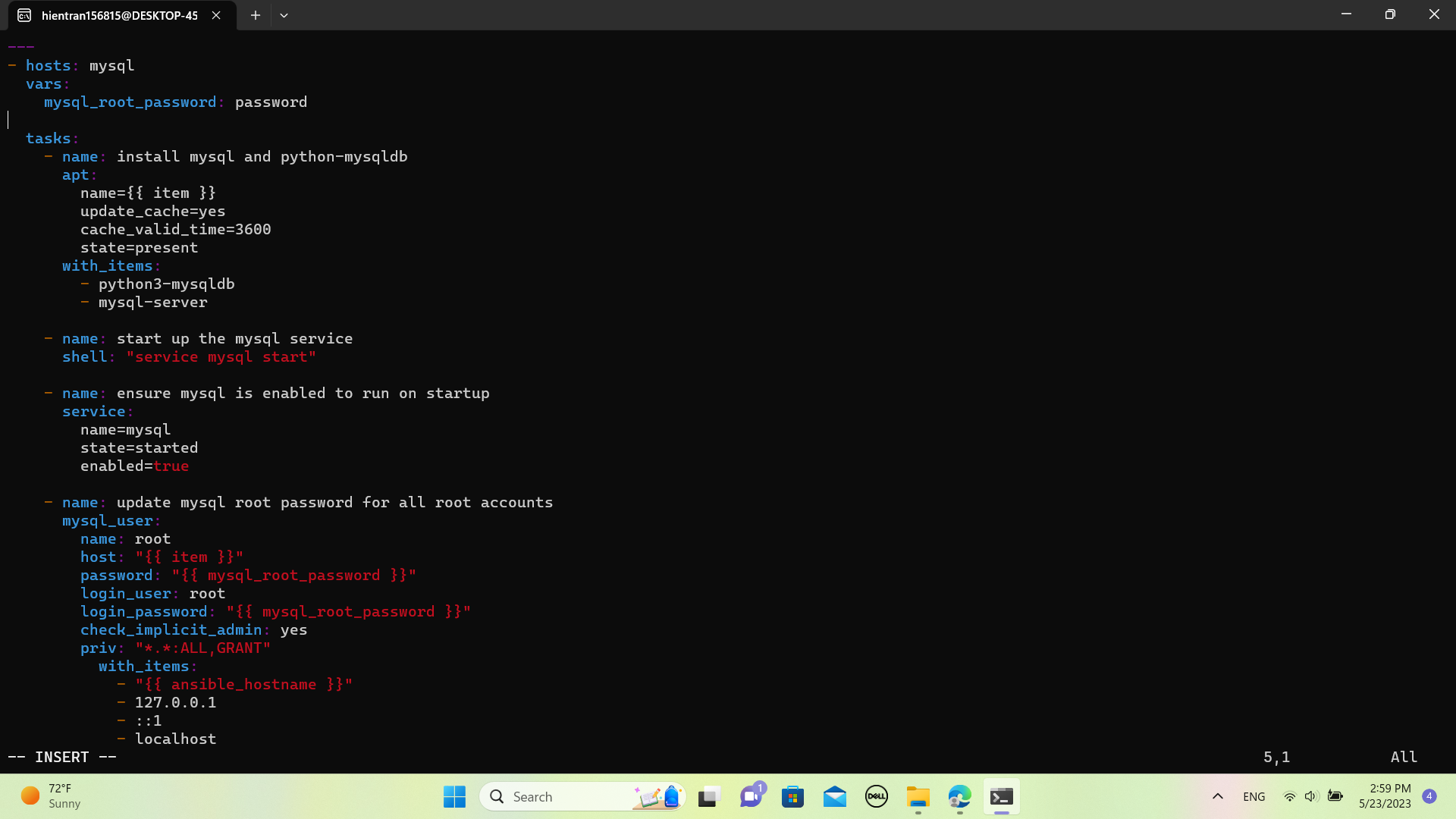
****

**Adding More Playbook Functionality with Variables and Loops**

1. **Add another entry with localhost IP address**

****

1. **Create a new playbook file called dbserver-playbook.yml**

****

**---**

**- hosts: mysql**

**vars:**

**mysql\_root\_password: password**

**tasks:**

**- name: install mysql and python-mysqldb**

**apt:**

**name={{ item }}**

**update\_cache=yes**

**cache\_valid\_time=3600**

**state=present**

**with\_items:**

**- python3-mysqldb**

**- mysql-server**

**- name: start up the mysql service**

**shell: "service mysql start"**

**- name: ensure mysql is enabled to run on startup**

**service:**

**name=mysql**

**state=started**

**enabled=true**

**- name: update mysql root password for all root accounts**

**mysql\_user:**

**name: root**

**host: "{{ item }}"**

**password: "{{ mysql\_root\_password }}"**

**login\_user: root**

**login\_password: "{{ mysql\_root\_password }}"**

**check\_implicit\_admin: yes**

**priv: "\*.\*:ALL,GRANT"**

**with\_items:**

**- "{{ ansible\_hostname }}"**

**- 127.0.0.1**

**- ::1**

**- localhost**

**- name: create a new database**

**mysql\_db:**

**name=testdb**

**state=present**

**login\_user=root**

**- name: add sample data to database**

**copy: src=dump.sql dest=/tmp/dump.sql**

**- name: insert sample data to database**

**mysql\_db:**

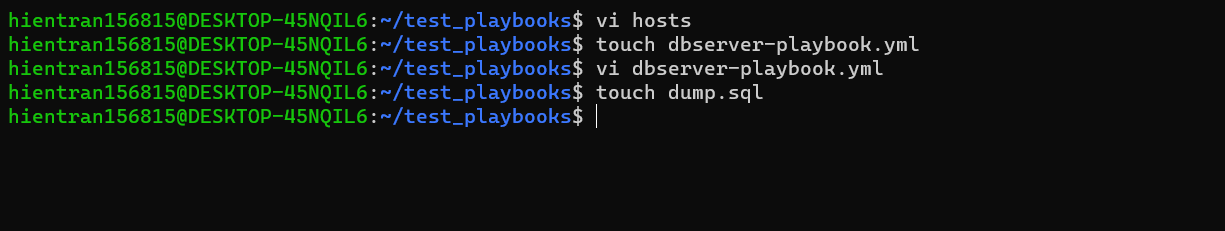
**name=testdb**

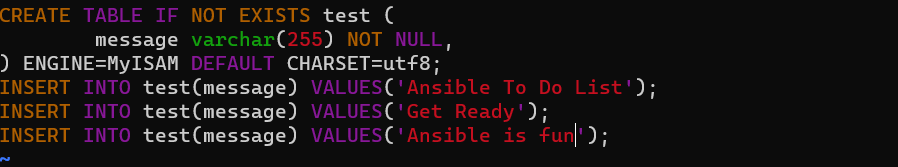
**state=import**

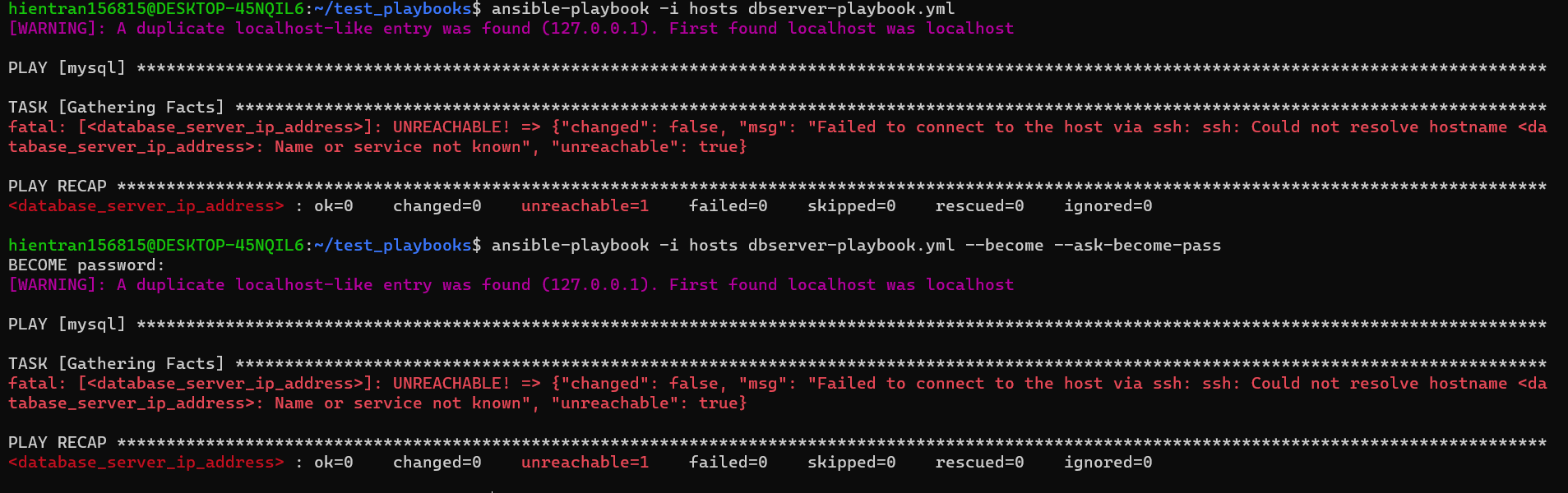
**target=/tmp/dump.sql**

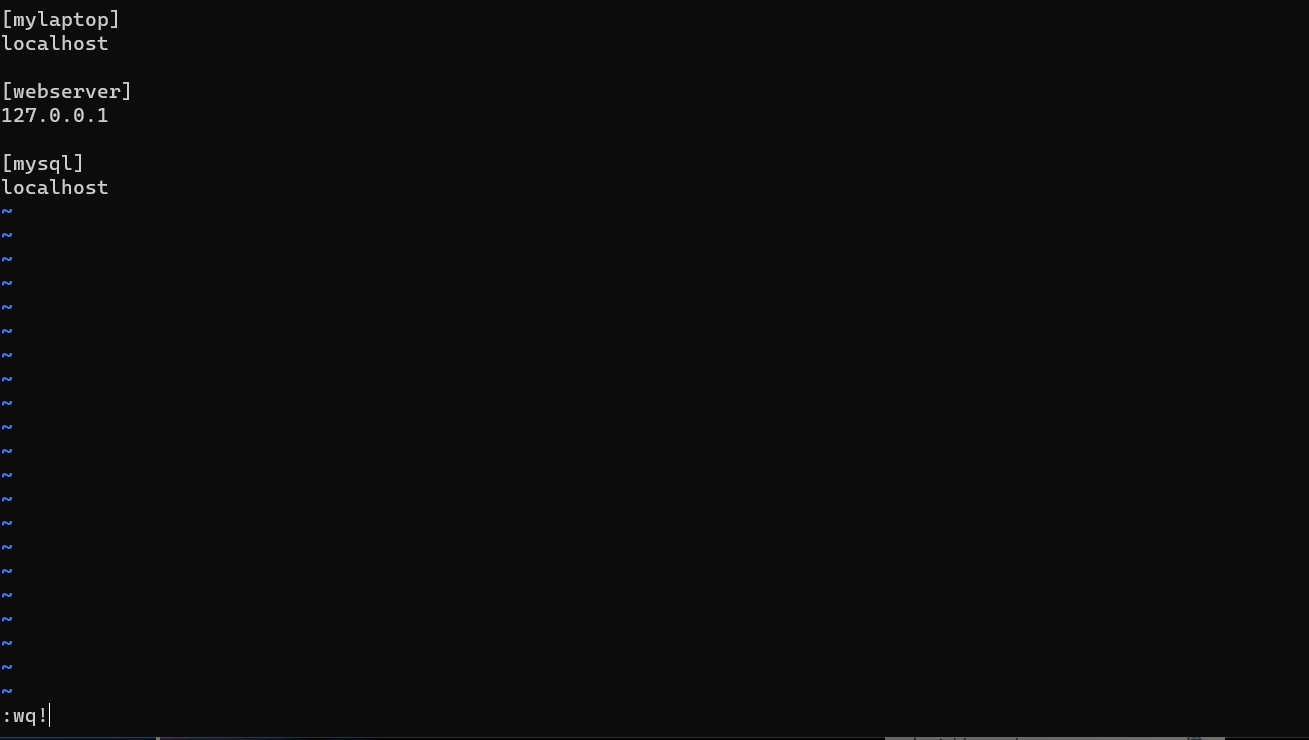
**login\_user=root**

**Create a sql file**

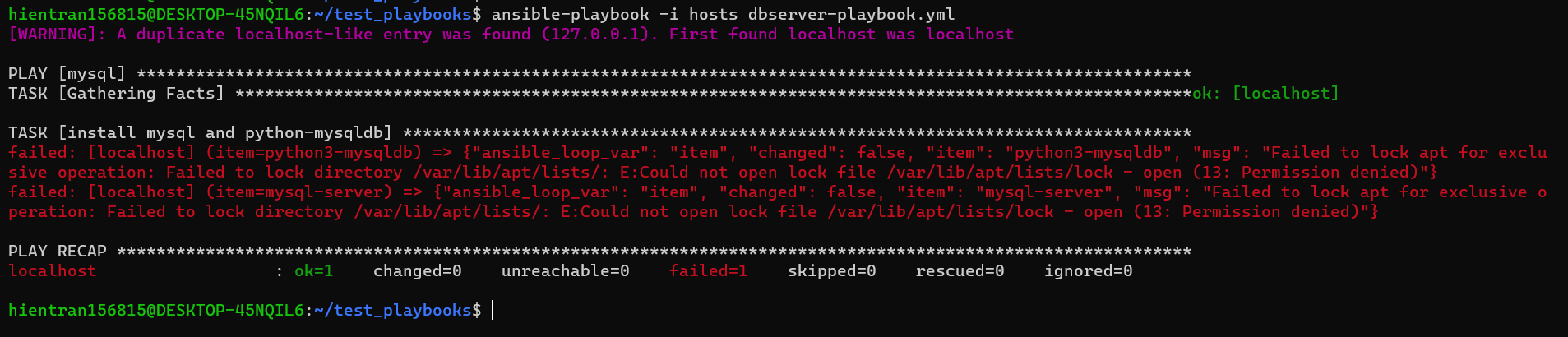
****

****

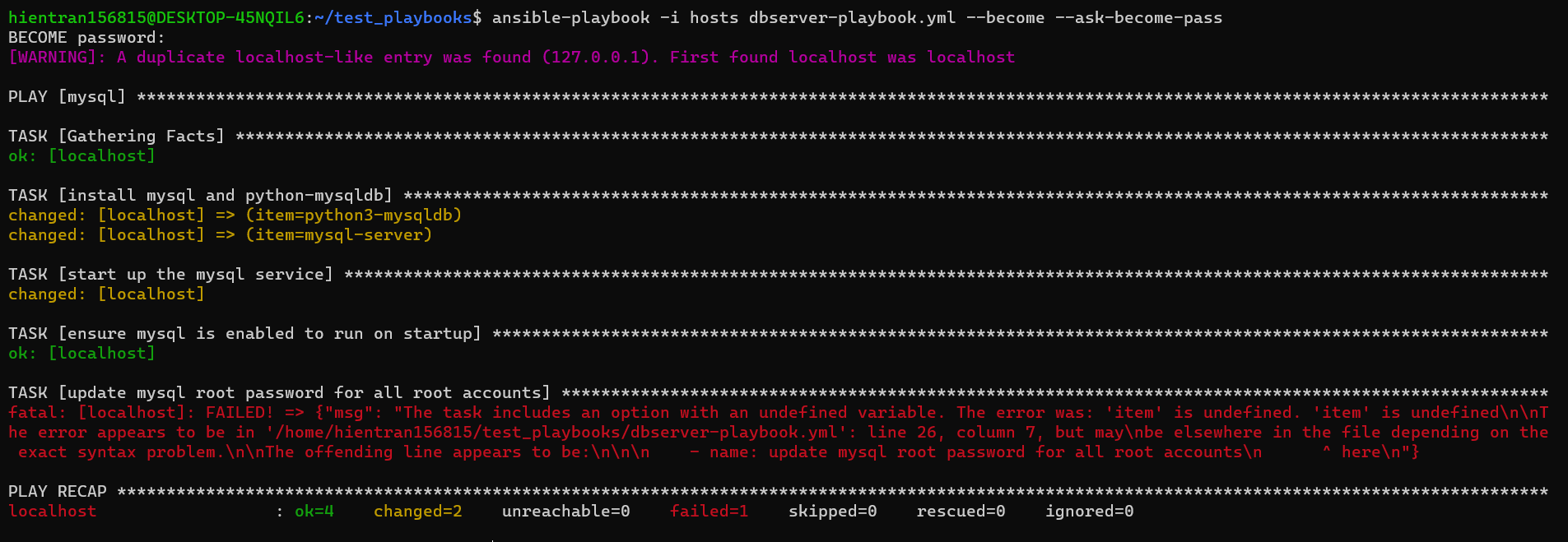
****

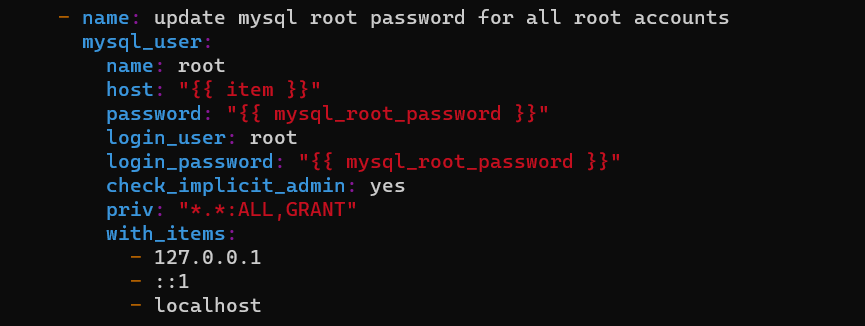
**TROUBLESHOOT  
**

**Rerun**

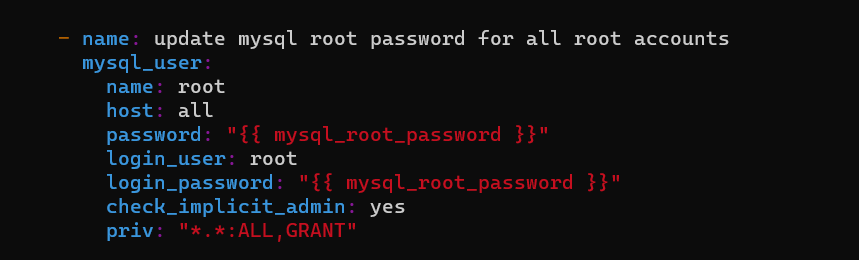
****

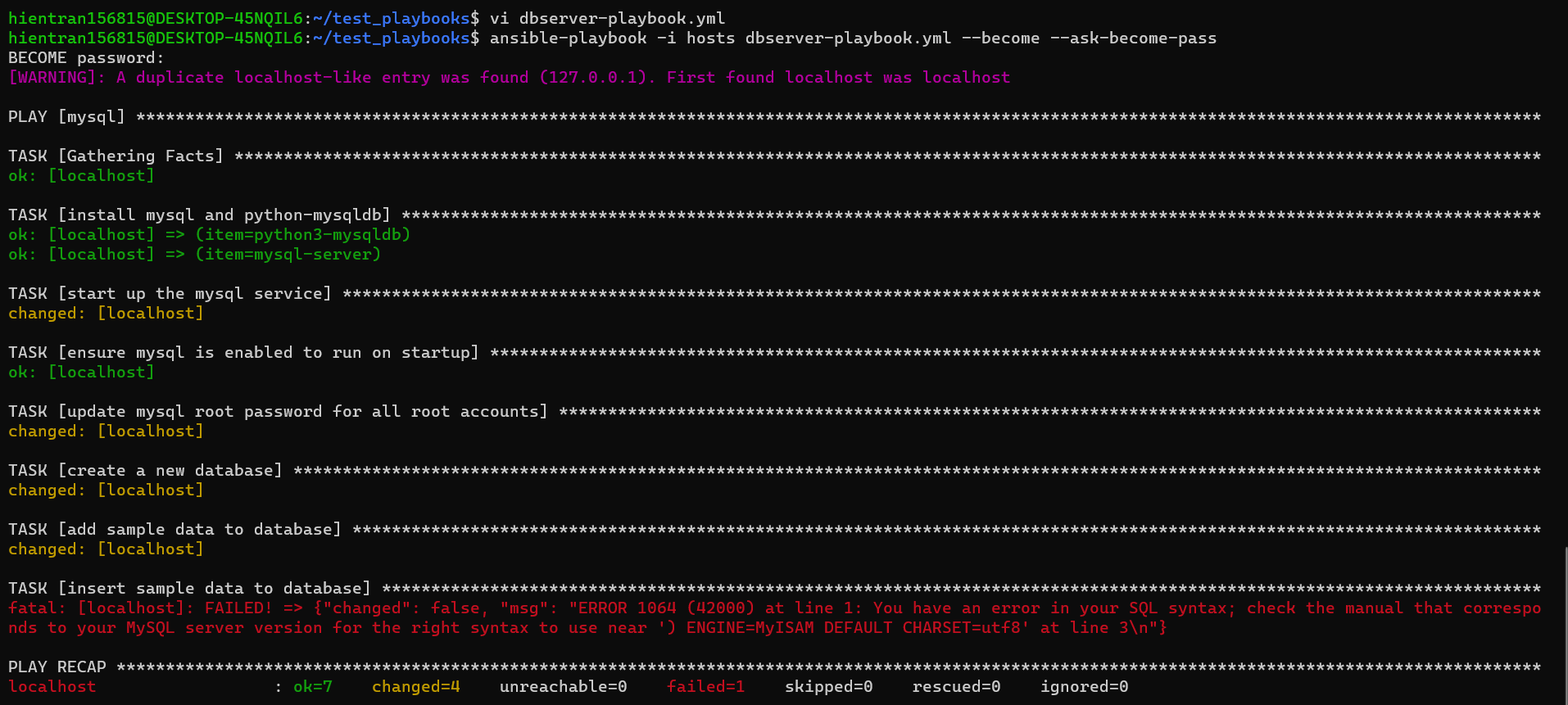
**TROUBLESHOOT**

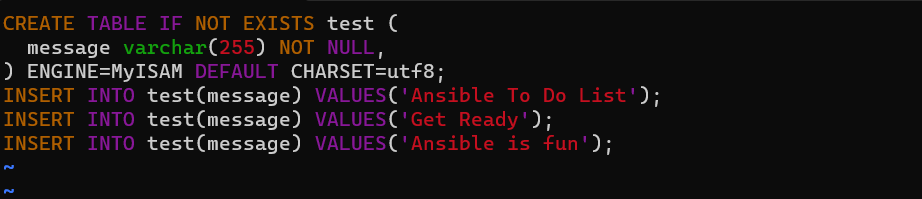
****

****

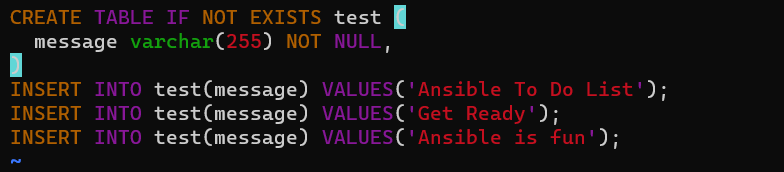
**Still Showed the same Error**

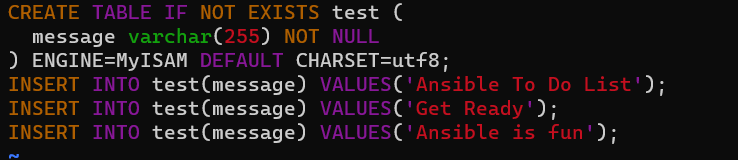
****

****

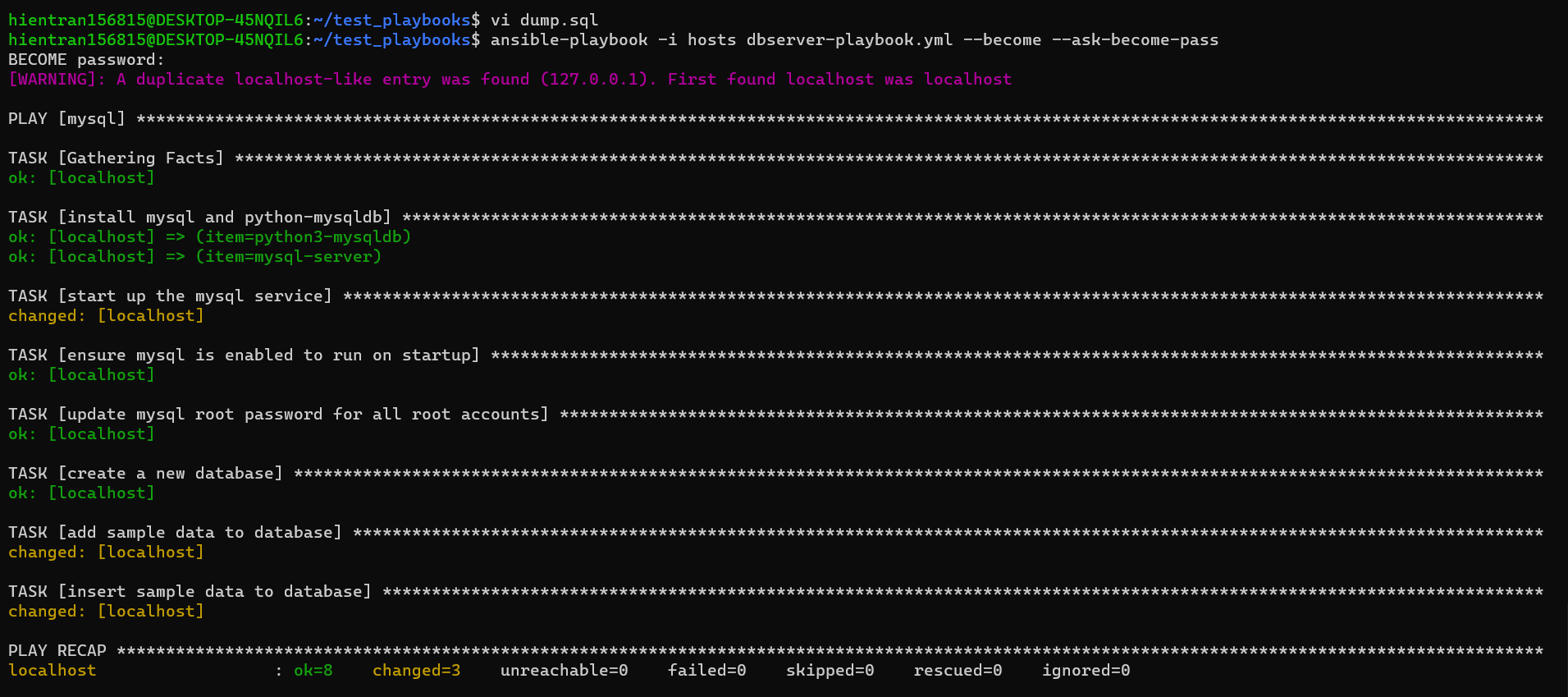
**TROUBLESHOOT  
**

**Still showed an error**

**  
ERROR occurs at the end of line 2 - should not have ‘,’**

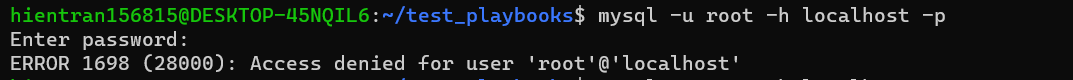
****

**Now run successfully**

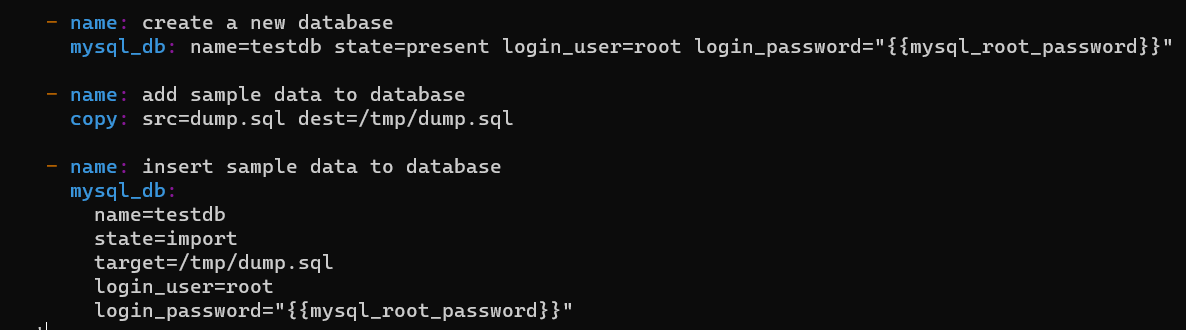
****

**→ Database is created and deployed to the server**

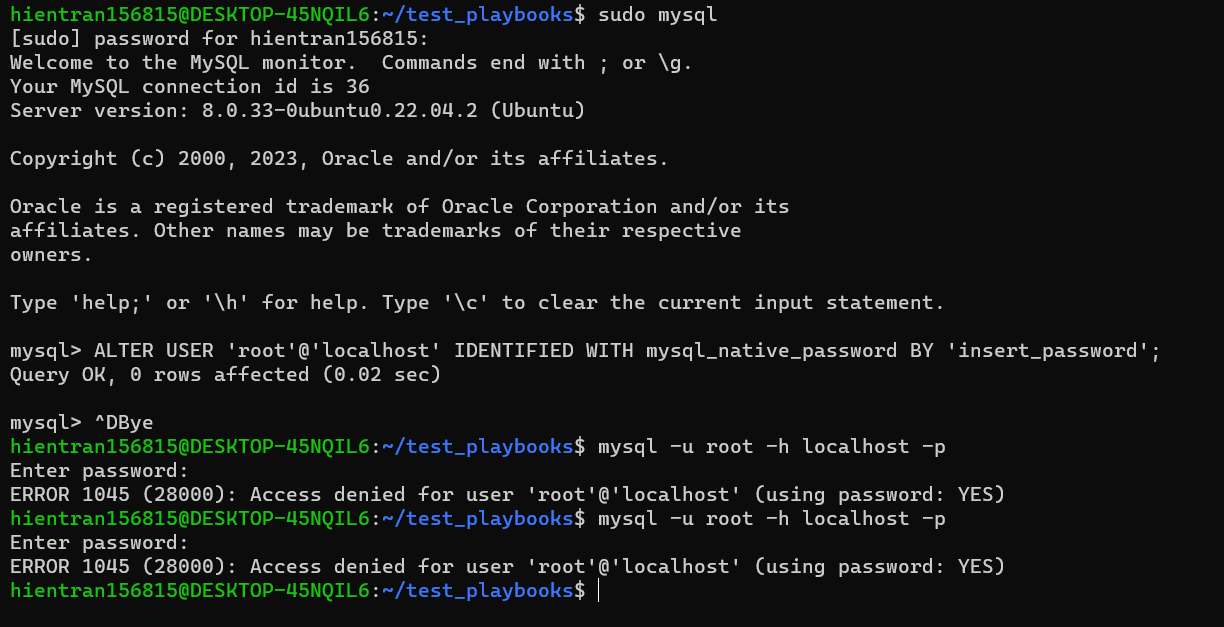
**Test passwords**

****

**TROUBLESHOOT**

****

**Rerun the ansible-playbook, it worked**

****

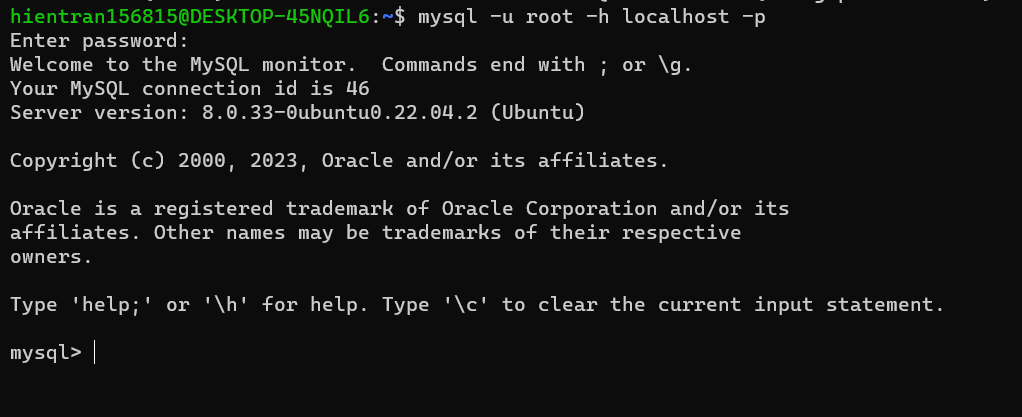
**Then sudo mysql**

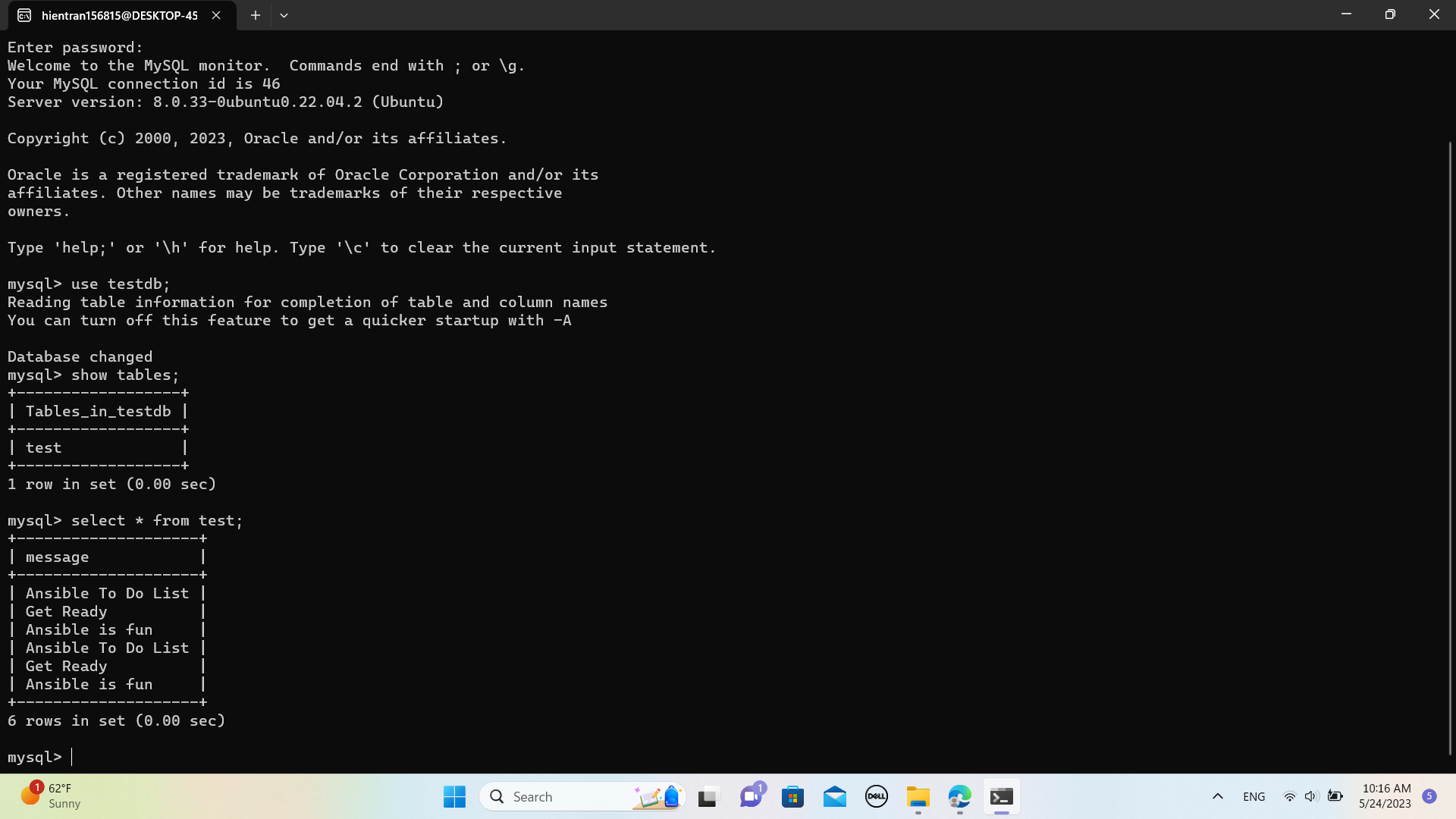
**Accessed denied**

**Rerun mysql -u root -h localhost -p**

**Password is insert\_password because in the previous step:**

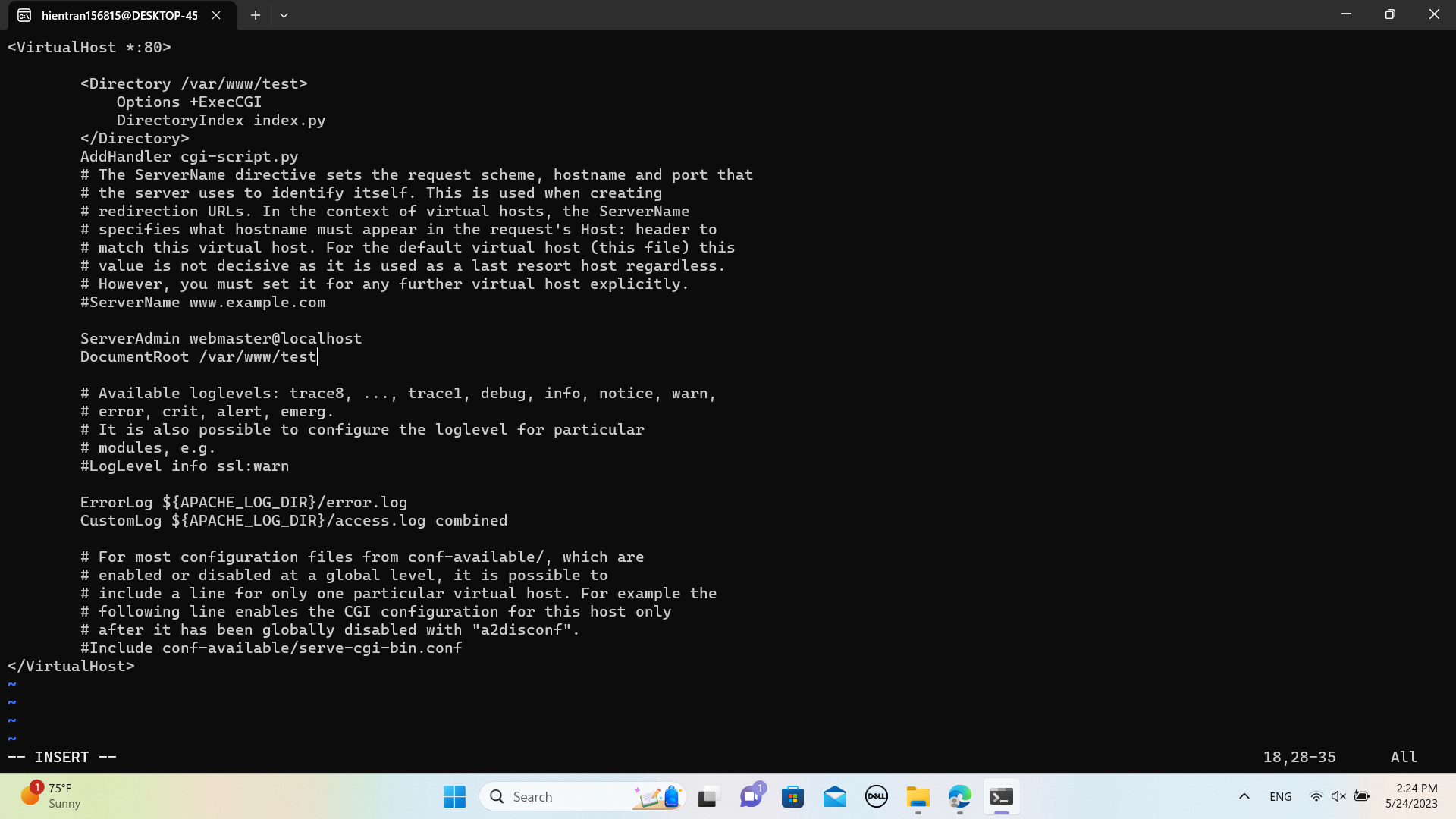
**ALTER USER 'root'@'localhost' IDENTIFIED WITH mysql\_native\_password BY 'insert\_password';**



IT WORKS  


**Set up Python for our LAMP stack**

**Open the 000-default.conf**

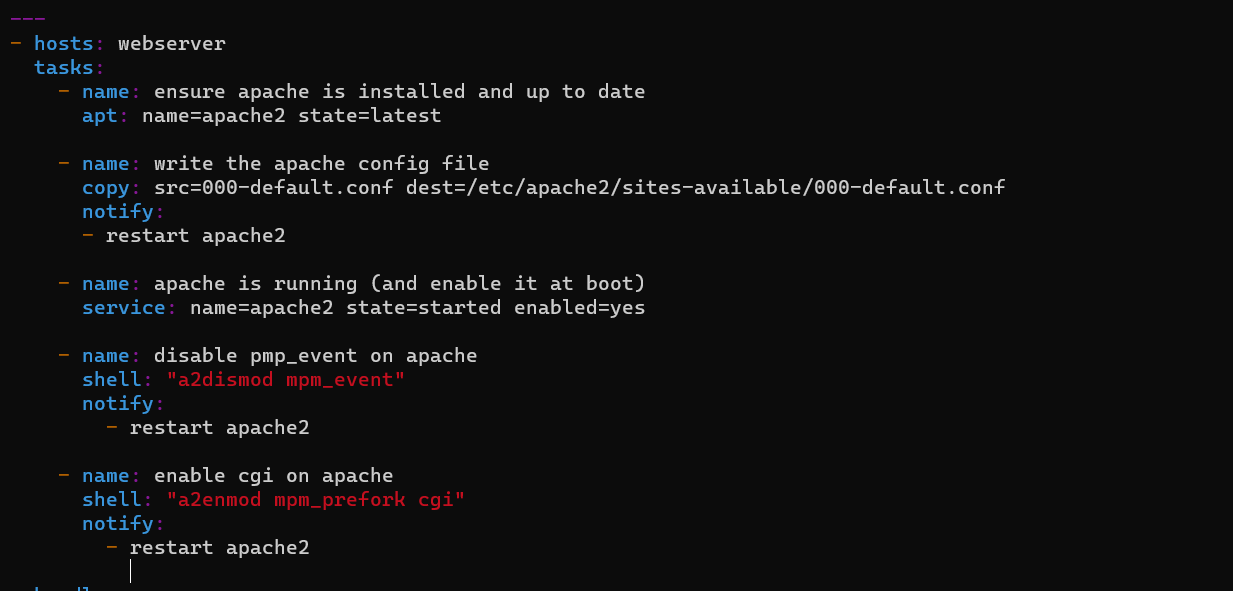


The configuration is setting up a new directory called /var/www/test which will run our Python code

Our web server playbook already uses the 000-default.conf as part of the installation,

Turn off multiprocessor mode and activate cgi

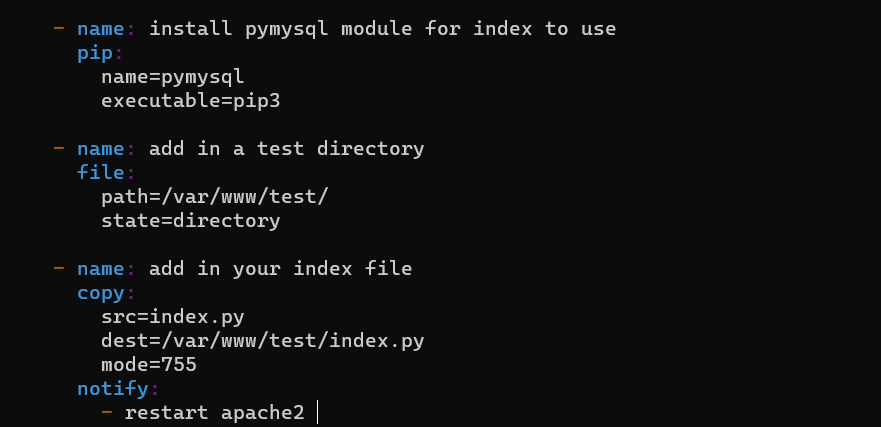
Notify handler to let Apache2 know it requires restart after making these changes



Install pymysql Python module using pip3

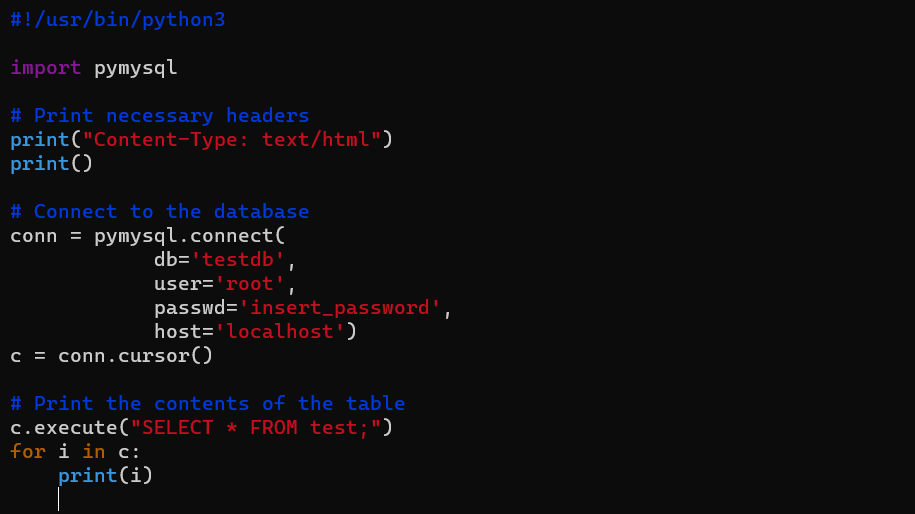
Create DocumentRoot directory of /var/www/test

Add new index script into this directory

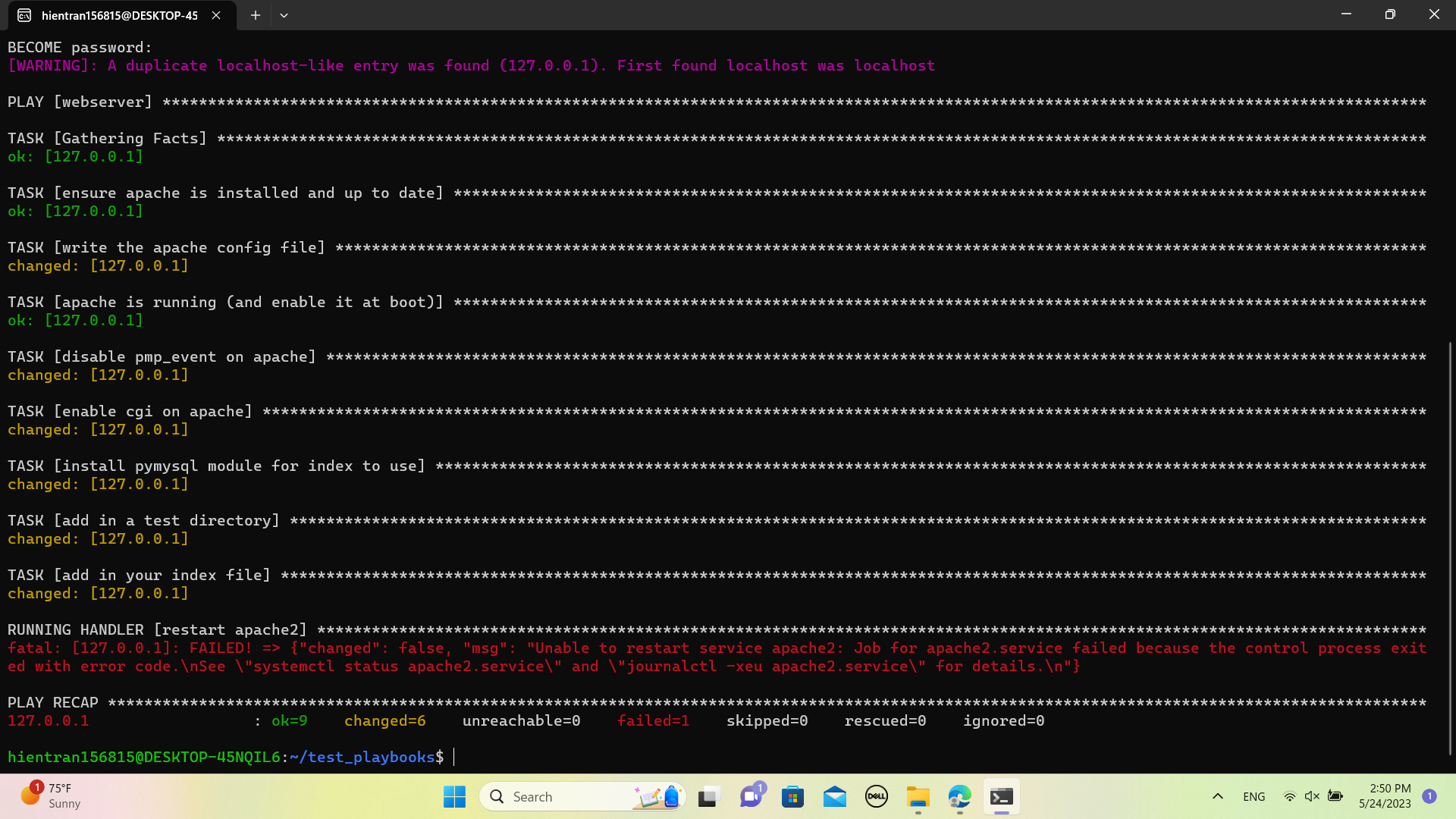


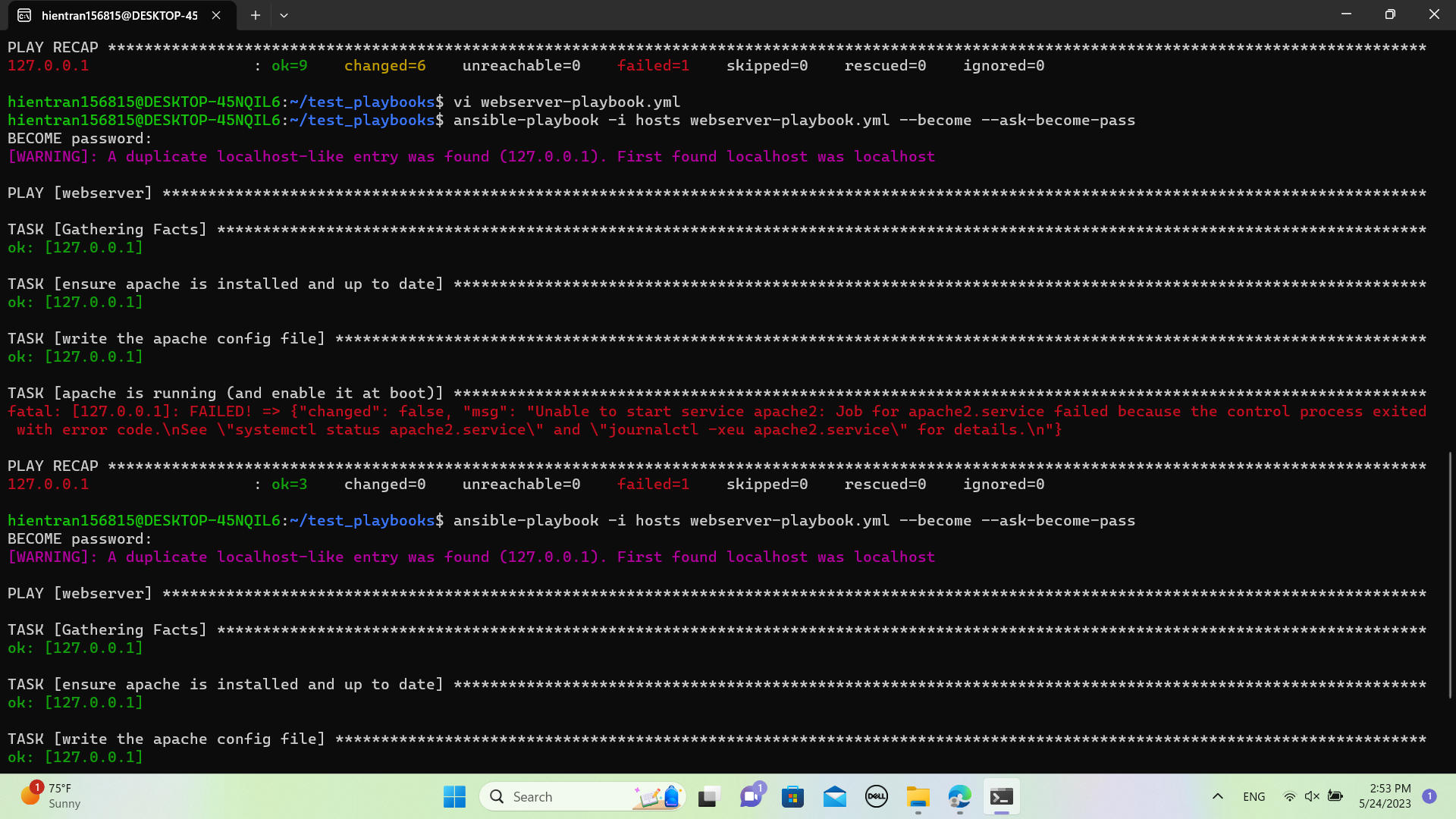
Create index.py

touch index.py

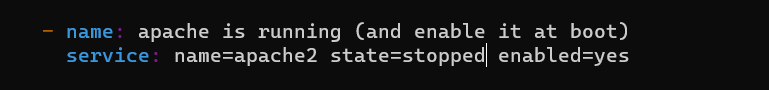


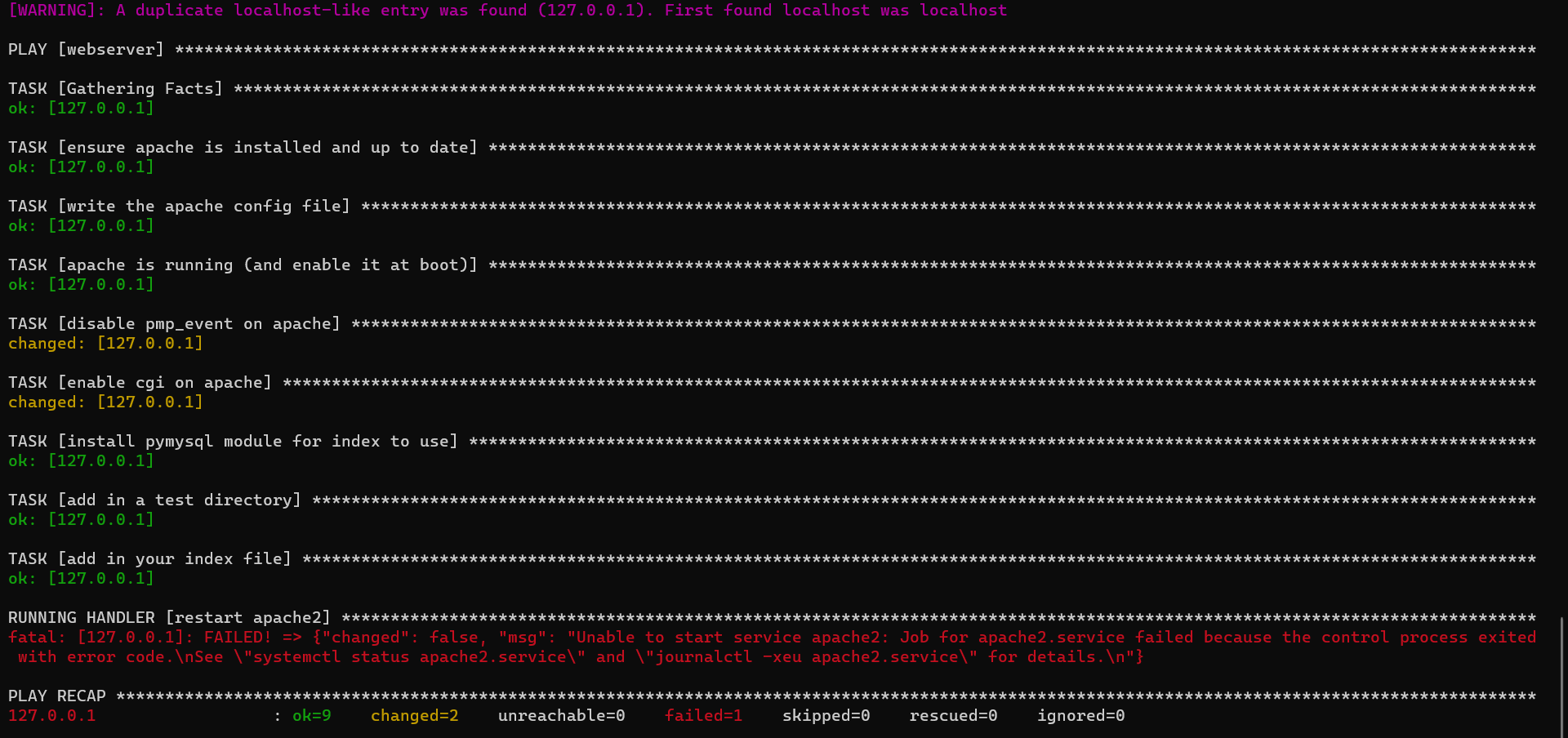
Run



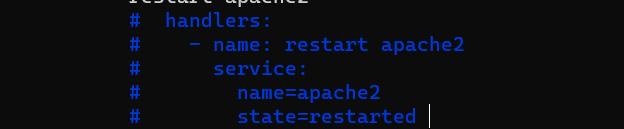


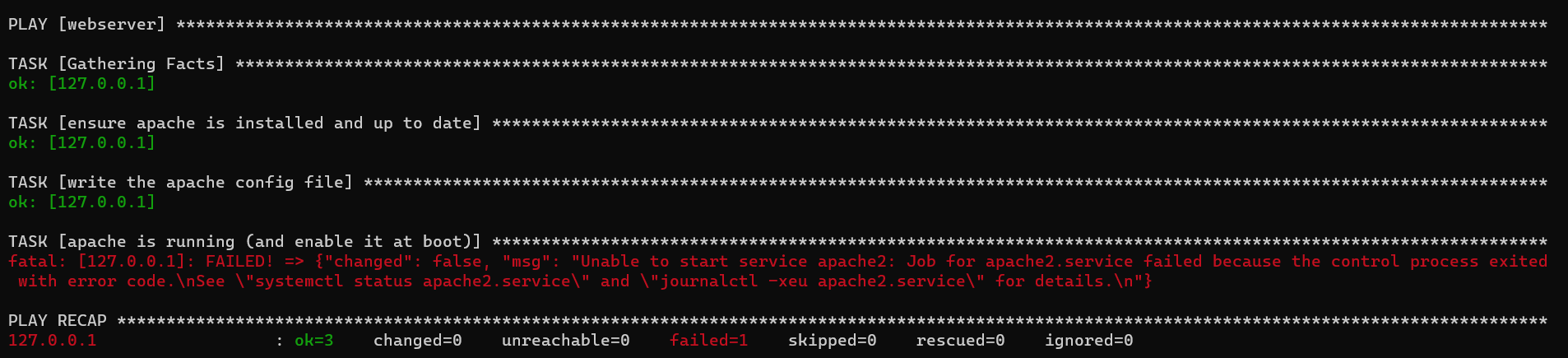
TROUBLESHOOT

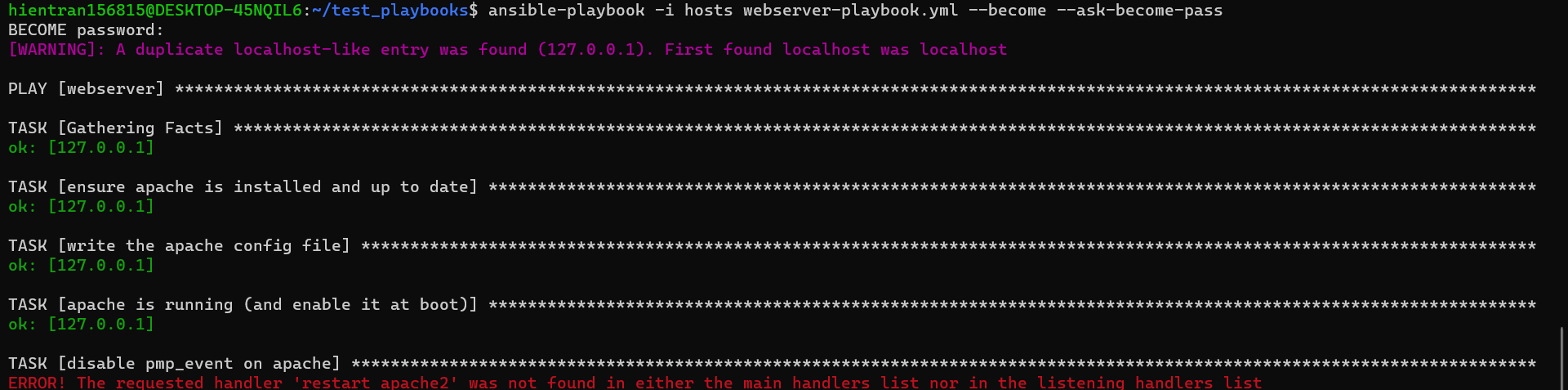
Change the state to stopped  




Commented out the handlers



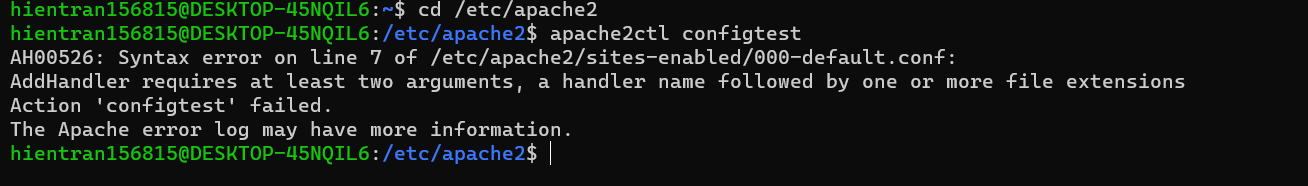


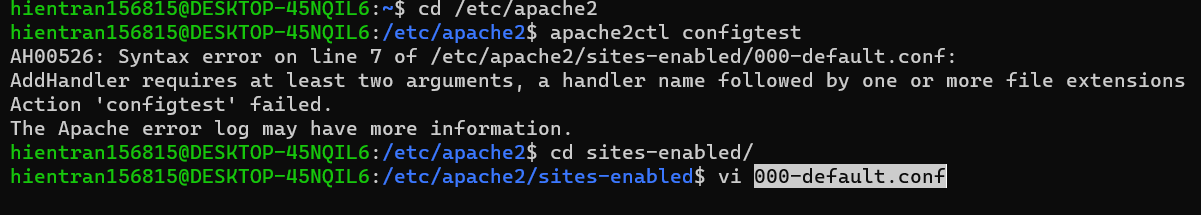


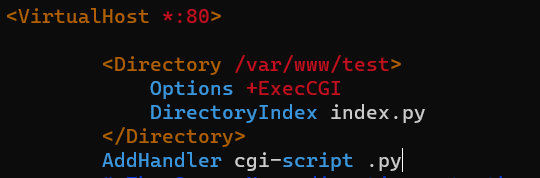
TROUBLESHOOT

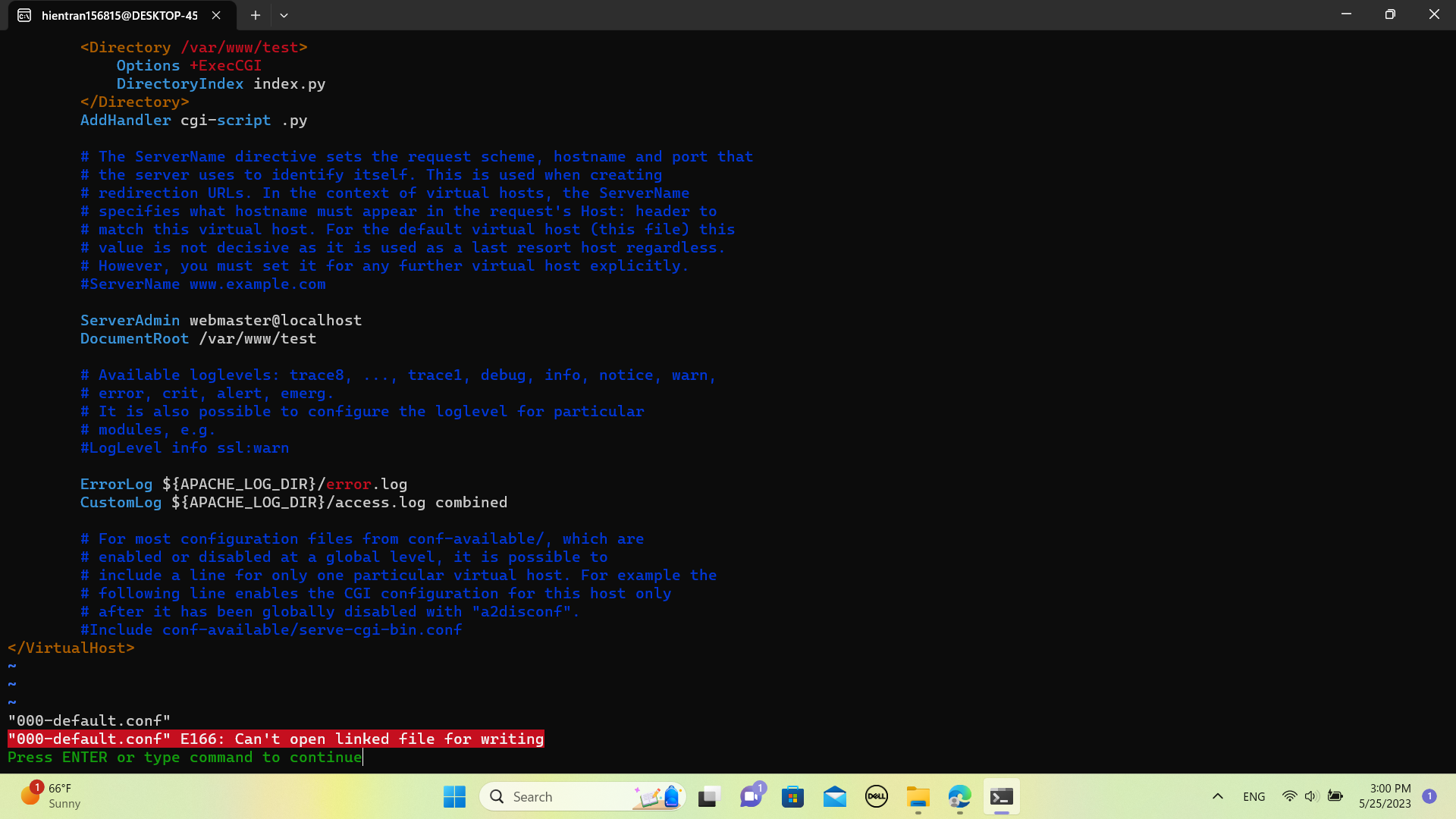
[apache2 - Apache not able to restart - Ask Ubuntu](https://askubuntu.com/questions/629995/apache-not-able-to-restart)

Error is in this file

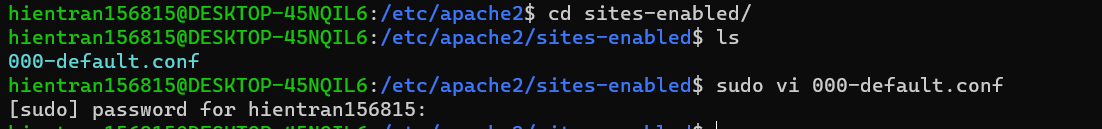








Used sudo vi to edit without getting the error

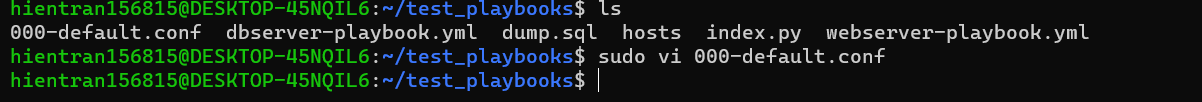


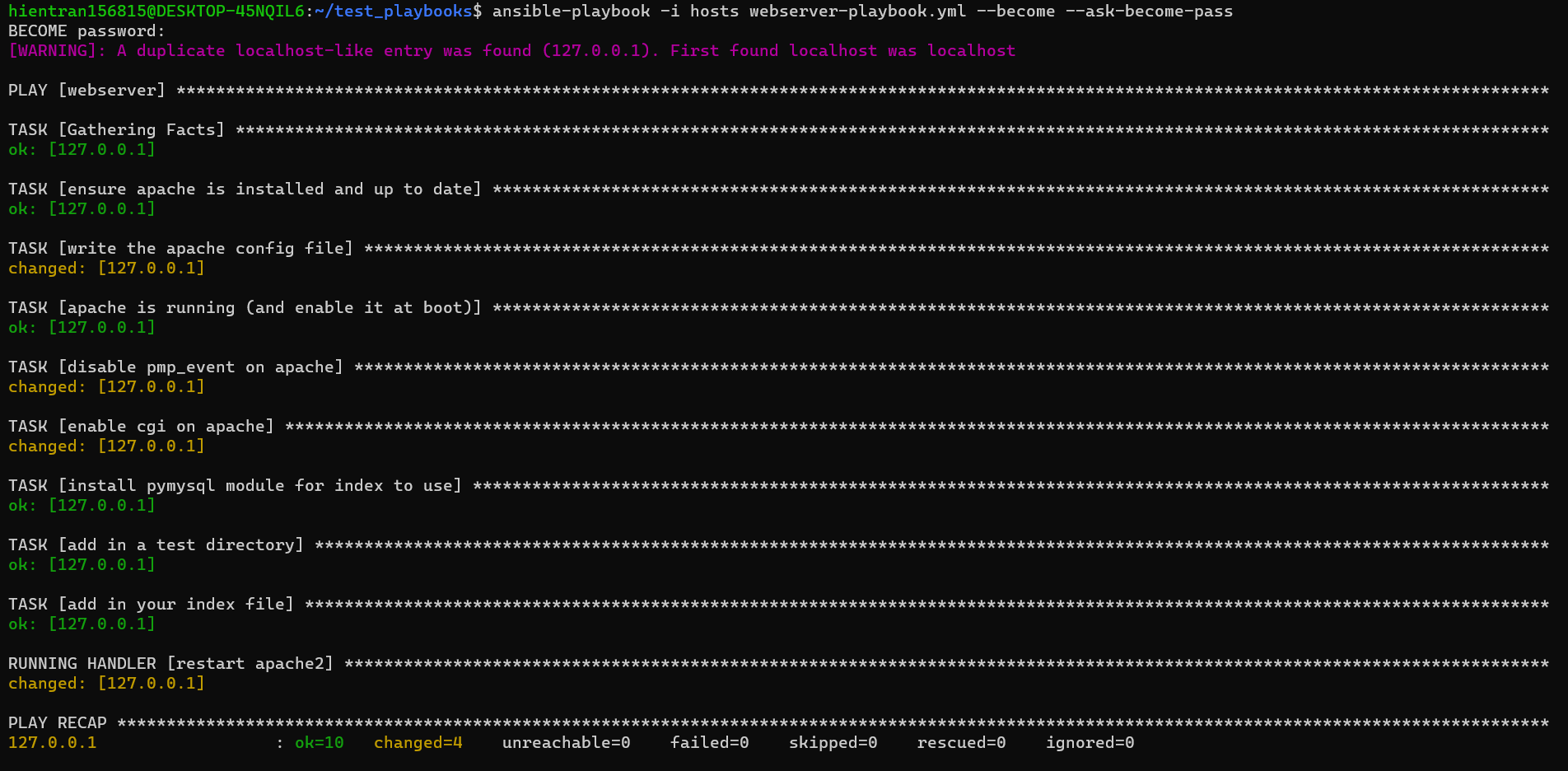
Rerun the playbook

Still failed

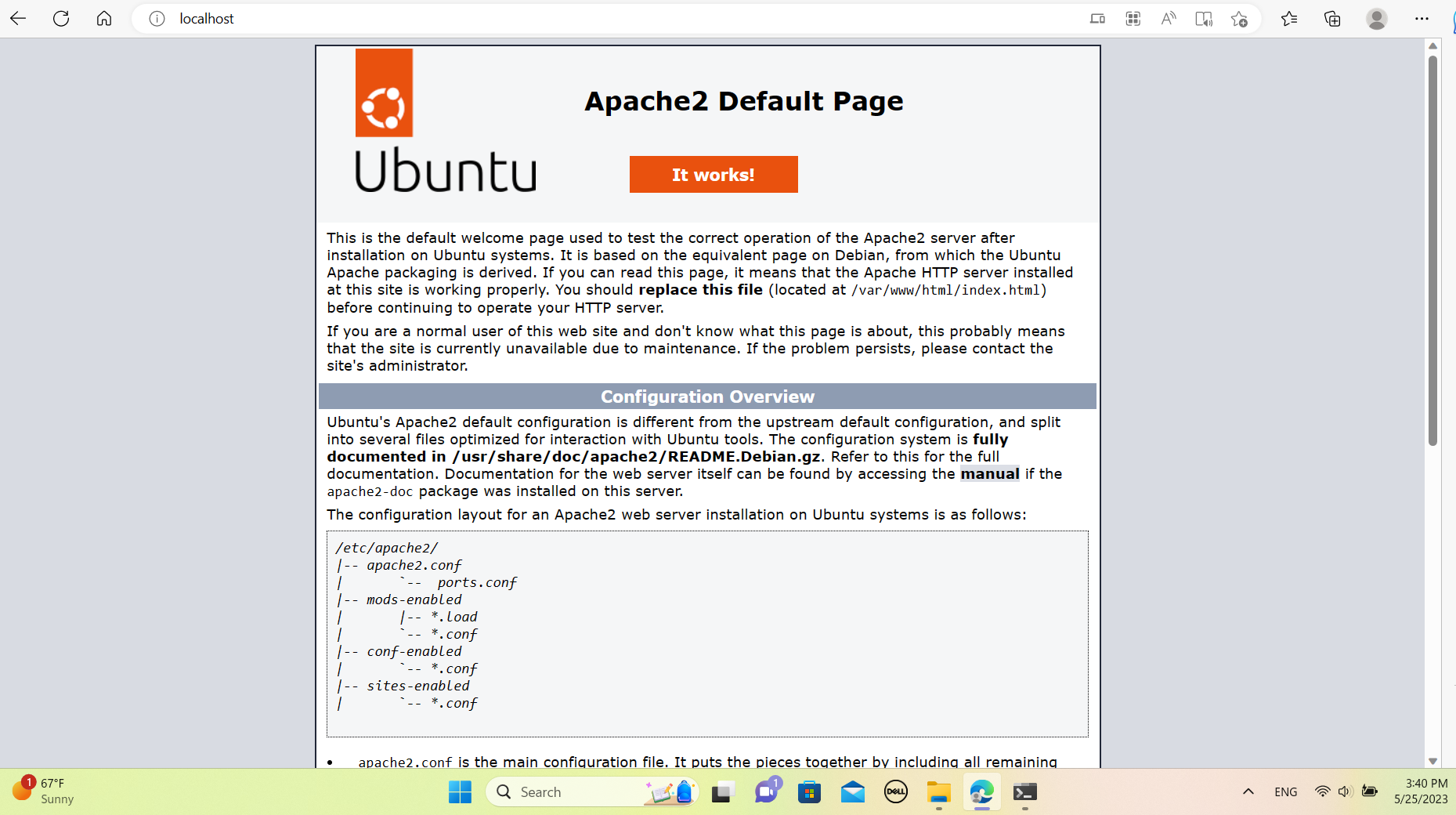
TROUBLESHOOT

Just changed here

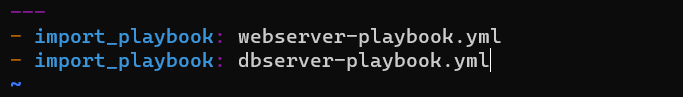




Go to http://localhost



Import both of the previous playbooks into the new-playbook.yml



Run playbook using new command

**ansible-playbook -i hosts new-playbook.yml**

