**Academic Year: 2024-25 Semester: V Class / Branch: TE IT**

**Subject: DevOPs Lab (DL)**

**Subject Lab In-charge: Prof. Sujata Oak**

**EXPERIMENT NO.11**

**Aim: To deploy a web application by provisioning LAMP Stack using ansible playbook.**

**Theory:** A LAMP stack is a bundle of four different software technologies that developers use to build websites and web applications. LAMP is an acronym for the operating system, Linux; the web server, Apache; the database server, MySQL; and the programming language, PHP. All four of these technologies are open source, which means they are community maintained and freely available for anyone to use. Developers use LAMP stacks to create, host, and maintain web content. It is a popular solution that powers many of the websites you commonly use today.

**ANSIBLE PLAYBOOK:**

Ansible playbooks are a vital part of Ansible and the core component of every Ansible configuration.An Ansible playbook is a file that contains a set of instructions that Ansible can use to automate tasks on remote hosts. Playbooks are written in YAML, a human-readable markup language.

A playbook typically consists of one or more plays, a collection of tasks run in sequence. Each task is a single instruction that Ansible can execute, such as installing a package, configuring a service, or copying a file.

By using Ansible playbooks, IT operations teams can automate infrastructure provisioning, configuration management, application deployment, and other operational tasks. Playbooks provide a concise and human-readable way to describe the desired automation workflows, making managing and scaling infrastructure configurations easier.

**STEP1: Clone ansible code from my github repository**

**Ansible-master:**

root@ip-172-31-18-177:~/.ssh# cd ~

root@ip-172-31-18-177:~# ls

snap

root@ip-172-31-18-177:~# mkdir ansible-lab

root@ip-172-31-18-177:~# cd ansible-lab/

root@ip-172-31-18-177:~/ansible-lab# git clone https://github.com/sujataoak799/ansible-codes.git

Cloning into 'ansible-codes'...

remote: Enumerating objects: 23, done.

remote: Counting objects: 100% (6/6), done.

remote: Compressing objects: 100% (5/5), done.

remote: Total 23 (delta 1), reused 4 (delta 1), pack-reused 17 (from 1)

Receiving objects: 100% (23/23), 8.63 KiB | 1.73 MiB/s, done.

Resolving deltas: 100% (6/6), done.

root@ip-172-31-18-177:~/ansible-lab# ls

ansible-codes

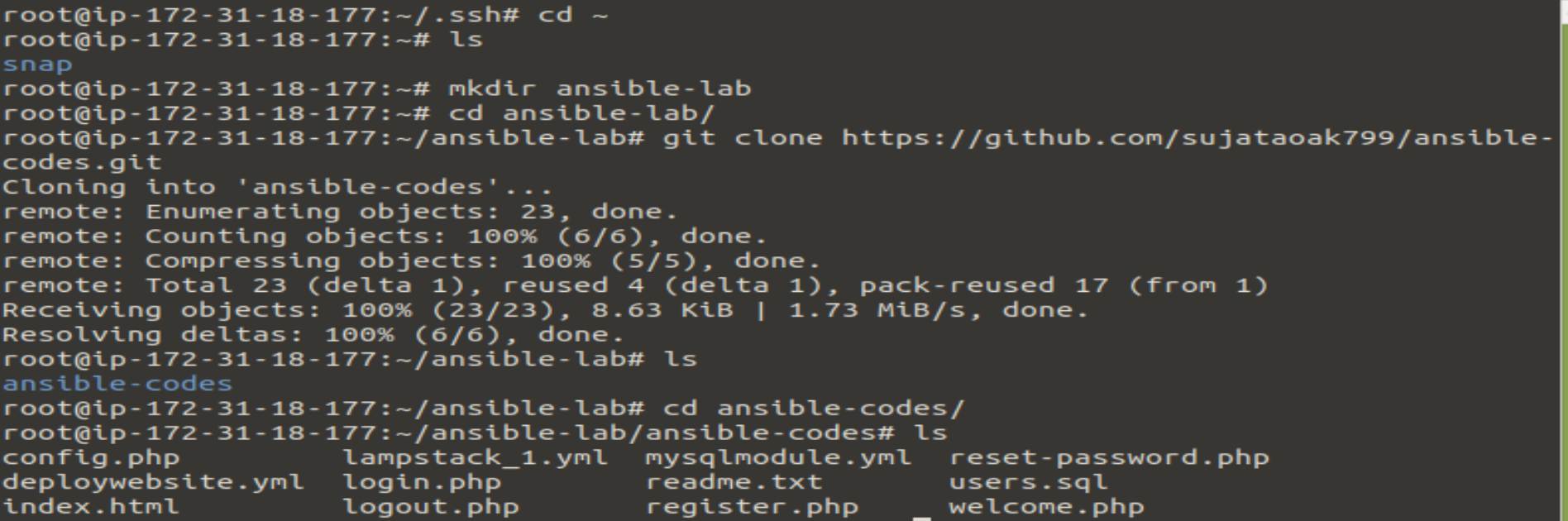
root@ip-172-31-18-177:~/ansible-lab# cd ansible-codes/

root@ip-172-31-18-177:~/ansible-lab/ansible-codes# ls

**config.php lampstack\_1.yml mysqlmodule.yml reset-password.php**

**deploywebsite.yml login.php readme.txt users.sql**

**index.html logout.php register.php welcome.php**

****

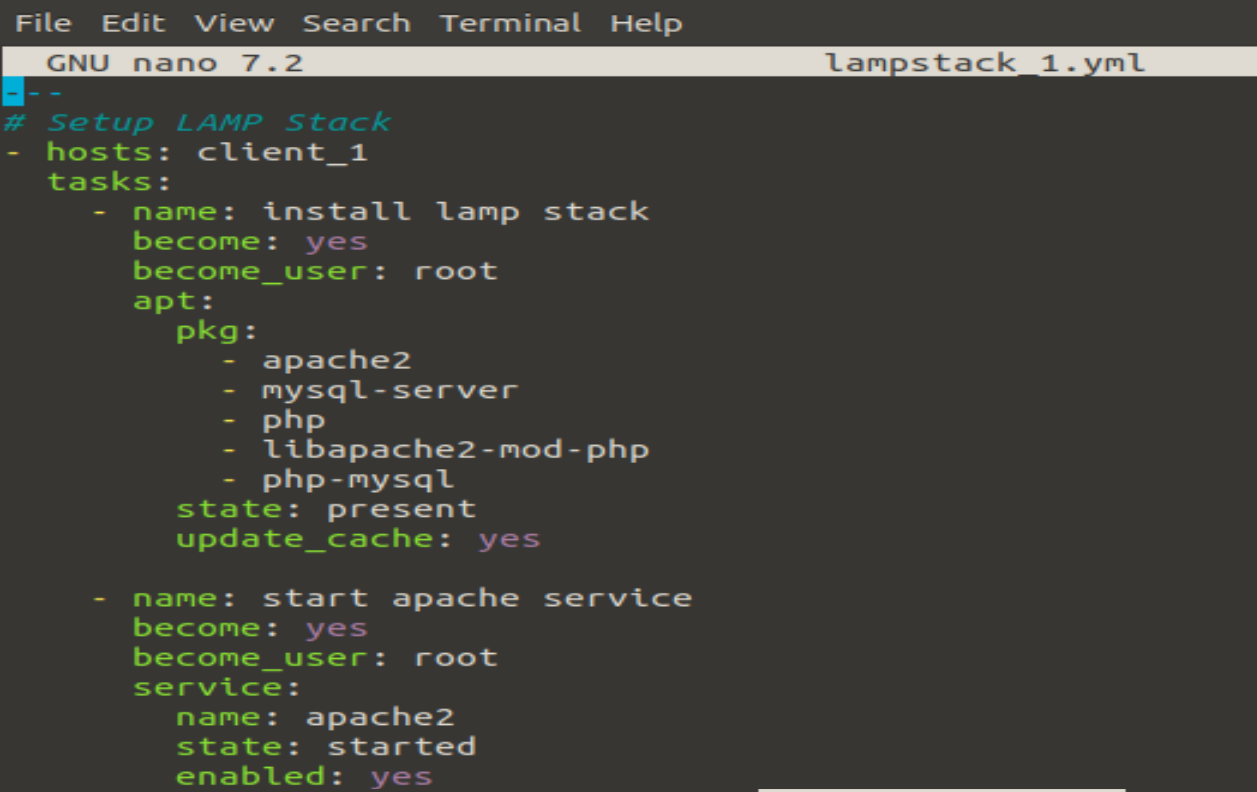
**STEP2:**

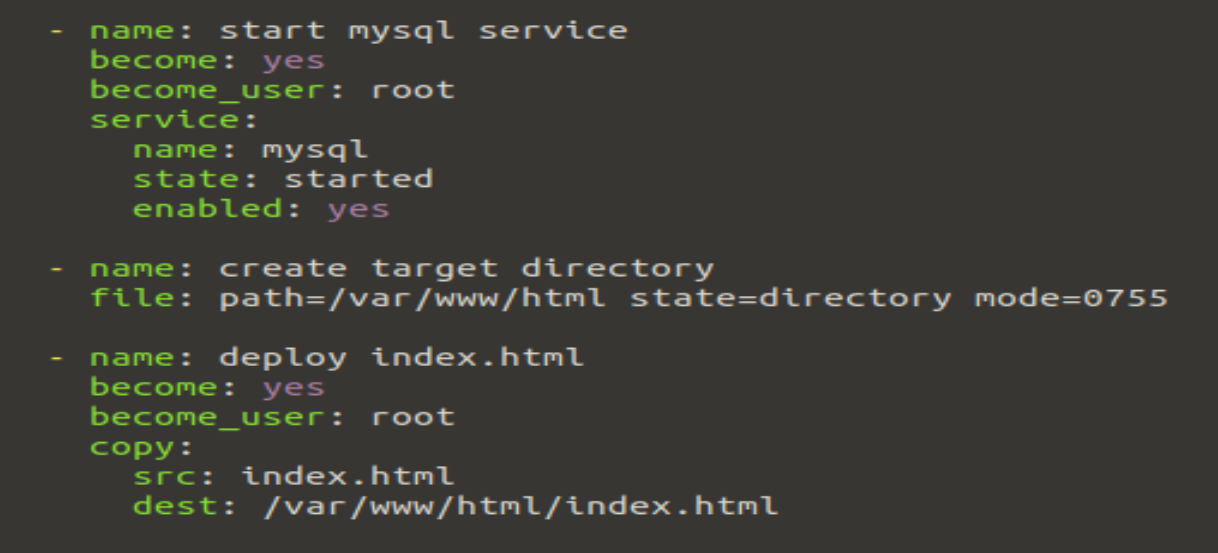
**Now all my files are in ansible-master machine and I need to deploy it on my ansible-slave machine. So we will be configuring our ansible-slave machine to host our full stack application.**

**The first playbook which I am going to setup on ansible-slave machine is lampstack\_1.yml**

**root@ip-172-31-18-177:~/ansible-lab/ansible-codes# nano lampstack\_1.yml**

****

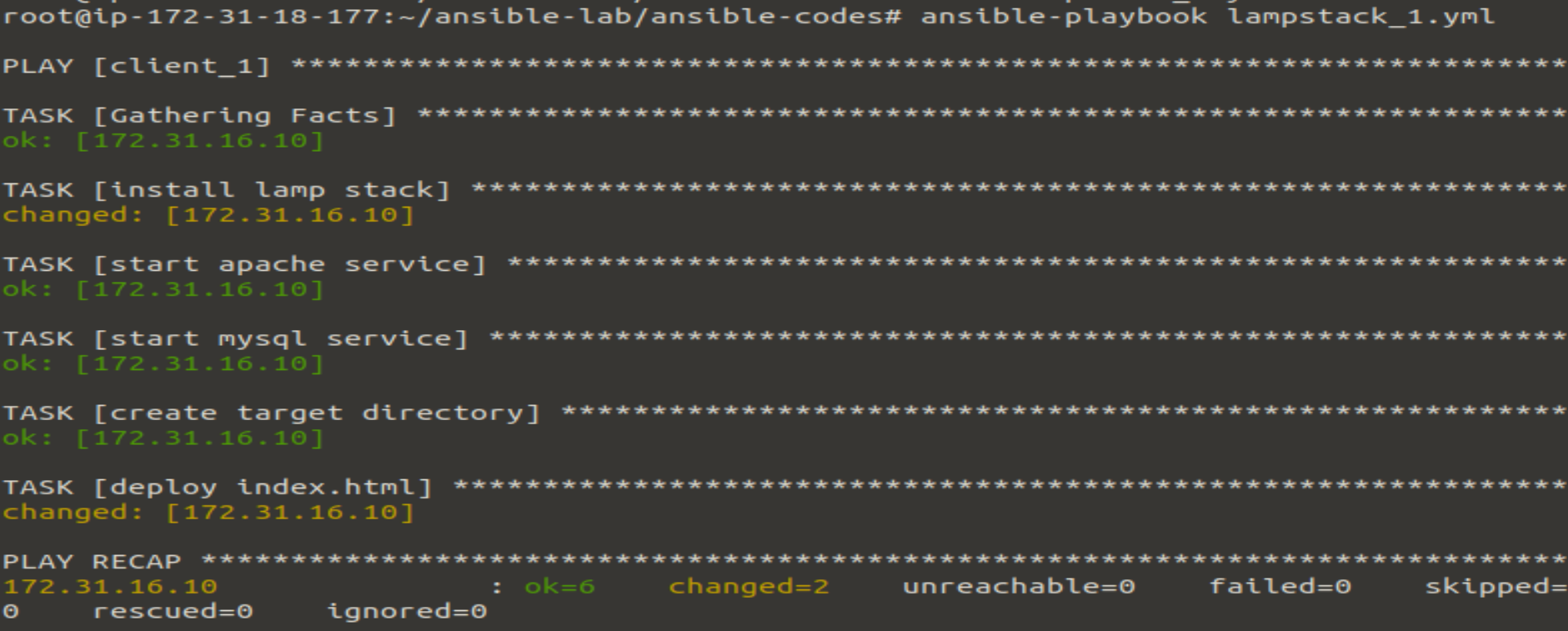




Save it.

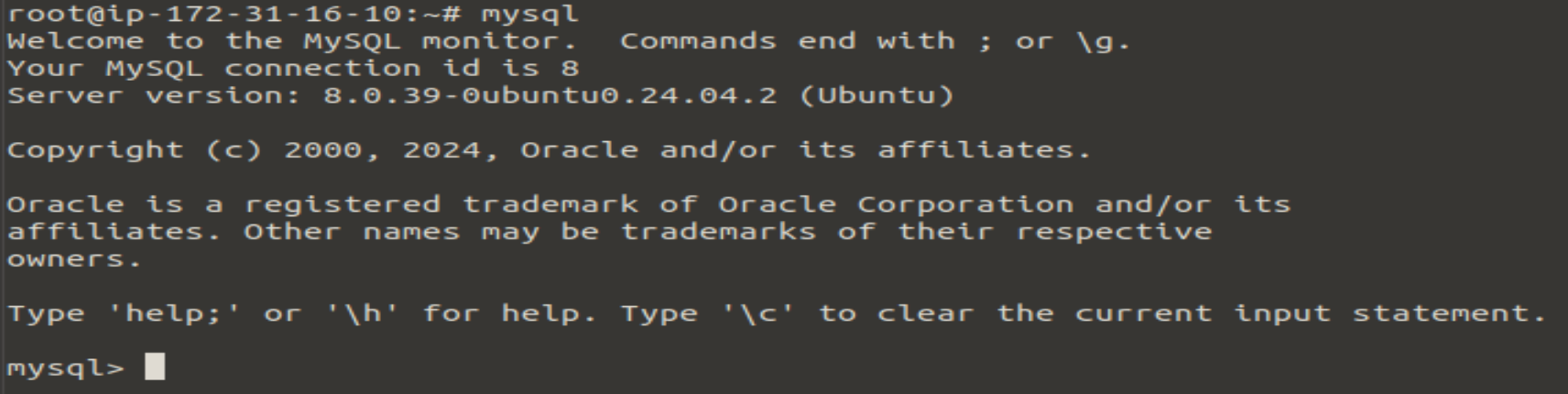
**STEP3: How to Run/Execute a playbook.**

**root@ip-172-31-18-177:~/ansible-lab/ansible-codes# ansible-playbook lampstack\_1.yml**

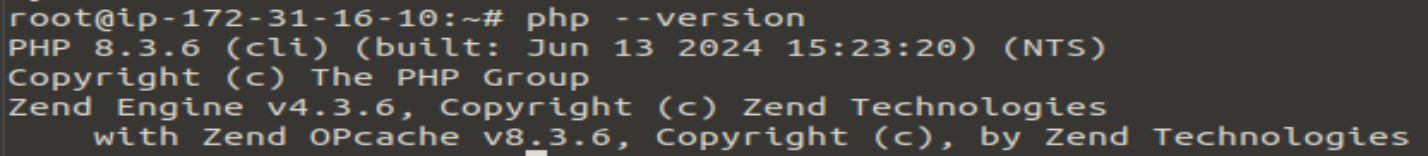
****

**Ansible-slave:**

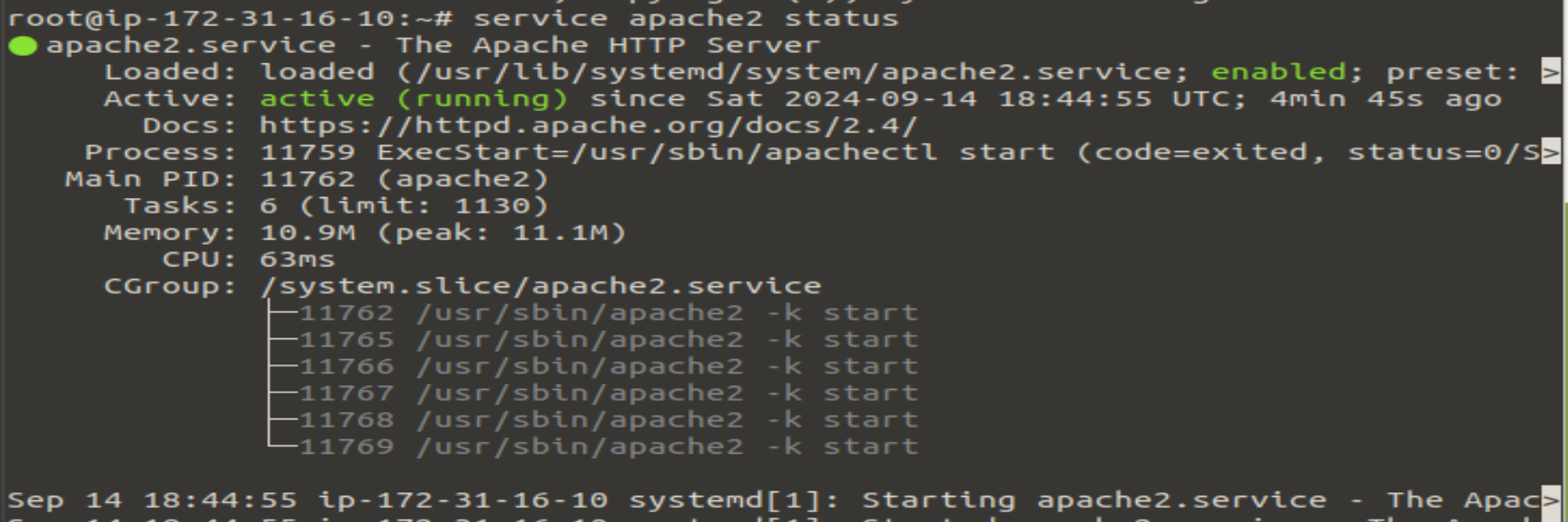
root@ip-172-31-16-10:~# mysql



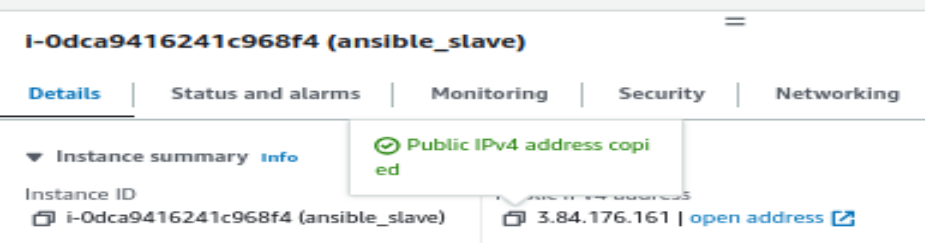
root@ip-172-31-16-10:~# php --version

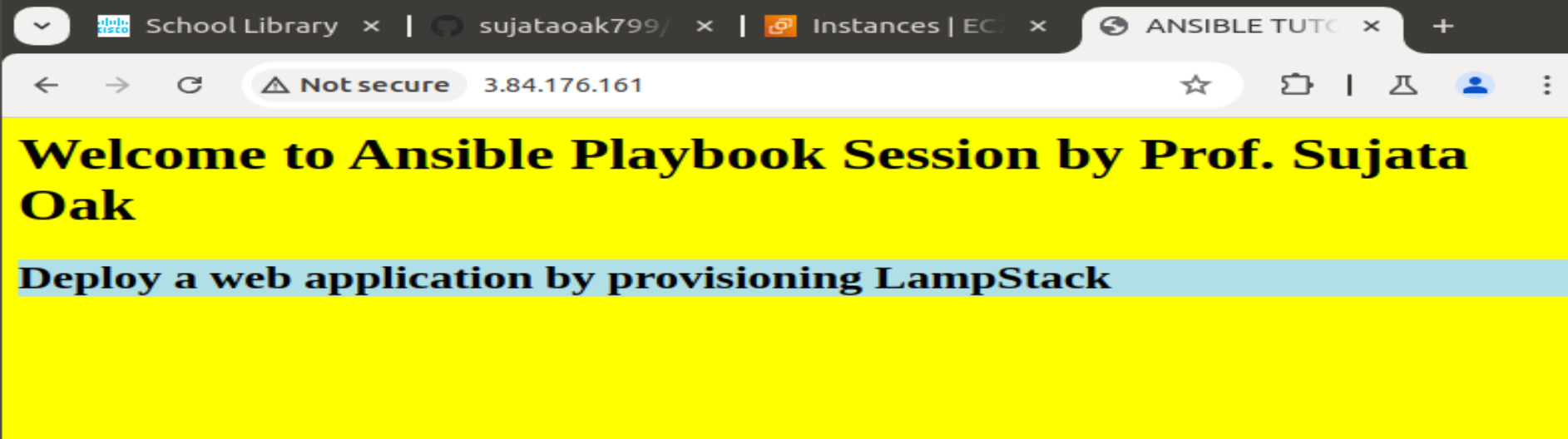


root@ip-172-31-16-10:~# service apache2 status



Once apache service status is active. Copy IPv4 address of ansible-slave machine in browser and you can see the deployment of index.html page.





**Conclusion:** In the experiment, successfully implemented provisioning lamp stack on ubuntu machine using ansible playbook.

[**https://aws.amazon.com/what-is/lamp-stack/**](https://aws.amazon.com/what-is/lamp-stack/)

[**https://www.simplilearn.com/what-is-ansible-playbook-article#how\_to\_write\_an\_ansible\_playbook**](https://www.simplilearn.com/what-is-ansible-playbook-article#how_to_write_an_ansible_playbook)