

Devops Lab 7 (BCSL657D)

✅ **Option 1: Use Windows Subsystem for Linux (WSL) – ● RECOMMENDED**

What's WSL?

WSL allows you to run a full Linux terminal inside Windows **without a VM**. It's lightweight and works perfectly for Ansible labs.

🔧 Steps to Set Up WSL + Ansible

Step 1: Install WSL (Windows Subsystem for Linux)

Open **PowerShell as Administrator**, and run this:

powershell

CopyEdit

wsl --install

This will:

- Install **Ubuntu** as the default Linux distro
- Automatically enable WSL and reboot your system

● If it says "The requested operation requires elevation," make sure you're **running PowerShell as Administrator**.

Step 2: Open Ubuntu (WSL)

After installation and reboot:

1. Search for "**Ubuntu**" in the Start menu and open it.
2. It will prompt you to set a **Linux username and password** (this is separate from your Windows account).

Now you're in a **real Linux terminal inside Windows** — this is where you can run:

bash

CopyEdit

sudo apt update

sudo apt install ansible -y

✅ That will install Ansible successfully.

🔍 Verify Ansible

bash

CopyEdit

ansible --version

You should see output like:

css

CopyEdit

ansible [core 2.13.x]

🚫 Do *NOT* Run sudo apt in PowerShell

PowerShell is **not a Linux terminal**. Only use sudo, apt, etc., in the **Ubuntu app**.

✅ **Important:** From now on, **do all Ansible-related lab work from inside the Ubuntu WSL terminal, not from PowerShell**.

✅ Lab Execution Plan (VTU Format)

🔧 Step 1: Create a Working Directory

Open your Ubuntu terminal and run:

bash

CopyEdit

mkdir ~/ansible-lab

cd ~/ansible-lab

📁 Step 2: Create an Inventory File

Create a file named hosts:

bash

CopyEdit

nano hosts

Paste this:

ini

CopyEdit

[local]

localhost ansible_connection=local

Then save:

- Press CTRL + O → Enter to save
 - Press CTRL + X to exit
-

Step 3: Test Ansible is Working

Run a simple ping module:

bash

CopyEdit

ansible -i hosts local -m ping

You should see:

json

CopyEdit

```
localhost | SUCCESS => {  
  "changed": false,  
  "ping": "pong"  
}
```

Step 4: Create a Sample Playbook

Create a file named install_nginx.yml:

bash

CopyEdit

nano install_nginx.yml

Paste this basic playbook:

yaml

CopyEdit

- name: Install and start NGINX on localhost

hosts: local

become: yes

tasks:

- name: Install NGINX

apt:

name: nginx

state: present

update_cache: yes

- name: Ensure NGINX is running

service:

name: nginx

state: started

enabled: yes

Save and exit (CTRL+O, Enter, then CTRL+X)

► Step 5: Run the Playbook

bash

CopyEdit

ansible-playbook -i hosts install_nginx.yml

You'll see output showing the tasks being executed.

Step 6: Verify the Result

Run:

bash

CopyEdit

`curl http://localhost`

You should see HTML output from the NGINX welcome page. 🎉

What to Include in Your VTU Lab Report

- Objective of the experiment
- Description of Inventory, Playbooks, and Modules
- Screenshots or output of:
 - hosts file
 - `ansible -m ping`
 - Playbook
 - Playbook execution
 - curl output
- Conclusion