Assignment 1: Setting up a Virtual Machine and Getting Started

Project Description

In this assignment, I learned how to create and configure a virtualized development environment using VMware Workstation Pro and Ubuntu 24.04 LTS. This hands-on project is designed to help set up a full Python environment with Jupyter Notebooks. I'll go through the process of installing VMware, configuring a virtual machine, installing Ubuntu, setting up Python, and accessing Jupyter Notebooks from the VM.

Step 1: Download and Install VMware Workstation Pro

1. **Go to the VMware Workstation Download Page.** (https://www.vmware.com/products/desktop-hypervisor/workstation-and-fusion)
2. Click **"Download Fusion or Workstation"** – you'll be redirected to Broadcom's website.
3. **Register an account:**
    - Click **Login** (top right), then **Register**.

- Enter your email, verify using the sent code, fill in your details, and create your account.

4. **After logging in**, click **“My Dashboard”** on the left panel.
5. Search for **“VMware Workstation Pro”**, click on the link.
6. **Scroll down and choose the version based on your OS** (e.g., Workstation Pro for Windows).
7. Choose the **“Personal Use”** option.
8. **Complete the form with your name and address**, then click **Submit**.
9. Click **Download** and run the installer after it's done.
10. **Follow the installation prompts.**
11. After installation, open VMware Workstation and choose **“Use VMware Workstation for Personal Use”**.
12. VMware is ready to use.

Step 2: Install Ubuntu on the VM

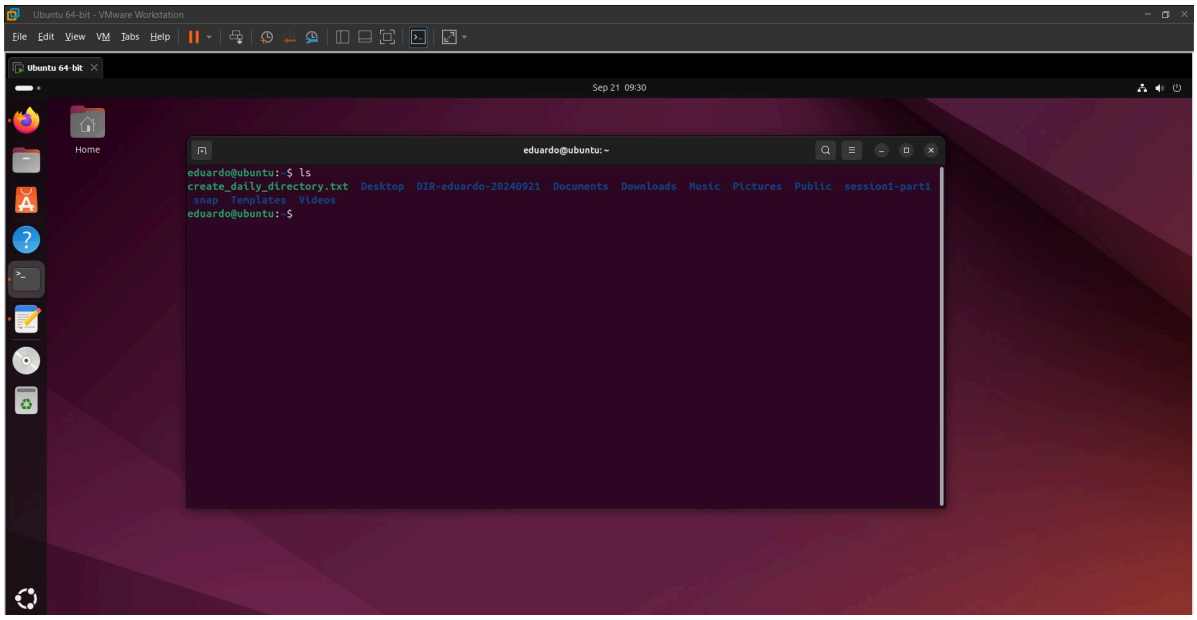
1. **Download Ubuntu 24.04 LTS .iso file** from:
https://ubuntu.com/download/desktop

(You can skip entering your email — the download will proceed regardless.)

2. **In VMware Workstation:**
   - Click **Create a New Virtual Machine**.
   - Select **“I will install the operating system later”**.
   - Choose:
     - **Guest OS**: Linux
     - **Version**: Ubuntu 64-bit
     - **Name your VM**: e.g., `Ubuntu 24.04 LTS`
     - **Choose a location to store the VM files**: e.g., `D:\Ubuntu24.04`

3. **Disk setup:**
    - Maximum disk size (GB): `100 GB`
    - Choose “Split virtual disk into multiple files”
    - Click **Finish**
4. Click **Finish** to create the VM.
5. Select the VM “**Ubuntu 24.04 LTS**”, click on “**Edit virtual machine settings**”
    - Go to **CD/DVD,** choose “Use ISO image file”, then choose location `D:\Ubuntu24.04` and then the `.iso` file.
    - Leave all other setting as default
6. **Select the VM “Ubuntu 24.04 LTS” once again** then click on the “**Start**” button to boot the VM.
7. **A pop-up will appear. Choose “Try or install Ubuntu” and press ENTER.** Ubuntu Installation will start.
8. **Follow installation prompts:**
    - Choose language, keyboard layout, etc.
    - Choose **Install Ubuntu**
    - Choose **Interactive Installation**
    - Choose **Default Selection**
    - Enable:
        - *Install third-party software*
        - *Support for additional media formats*
    - Choose **Erase disk and install Ubuntu** (safe for virtual machines).
9. **Create your user:**
    - Name, computer name, username, password.
    - Set your timezone.
10. Click **Install**, wait ~5-10 mins.

11. Click **Restart Now** when prompted.
12. **Login with your new credentials.**
13. If prompted, click **Skip for now** on Ubuntu Pro screen.
14. **Open a terminal and type** `ls` to verify it's working.



Step 3: Install Python 3.8

1. **Open terminal in Ubuntu.**
2. **Run the following commands:**

```
sudo apt update
sudo apt install software-properties-common
sudo apt install python3.8
```

3. **Add Python 3.8 as the default** `python` **command:**
   - Access shell configuration: `vim ~/.bashrc`
   - Add alias python to shell configuration: `alias python=python3`
   - Update shell configuration: `source ~/.bashrc`

→ Note: now python runs under alias command python

4. **Confirm installation**: `python --version`

Step 4: Install Jupyter Notebook

1. Install Jupyter library:

```
sudo apt install jupyter-core
sudo apt install jupyter
```

2. Launch Jupyter locally:

```
jupyter notebook
```

3. **A browser window should open.** Click on **New > Python 3** and try:

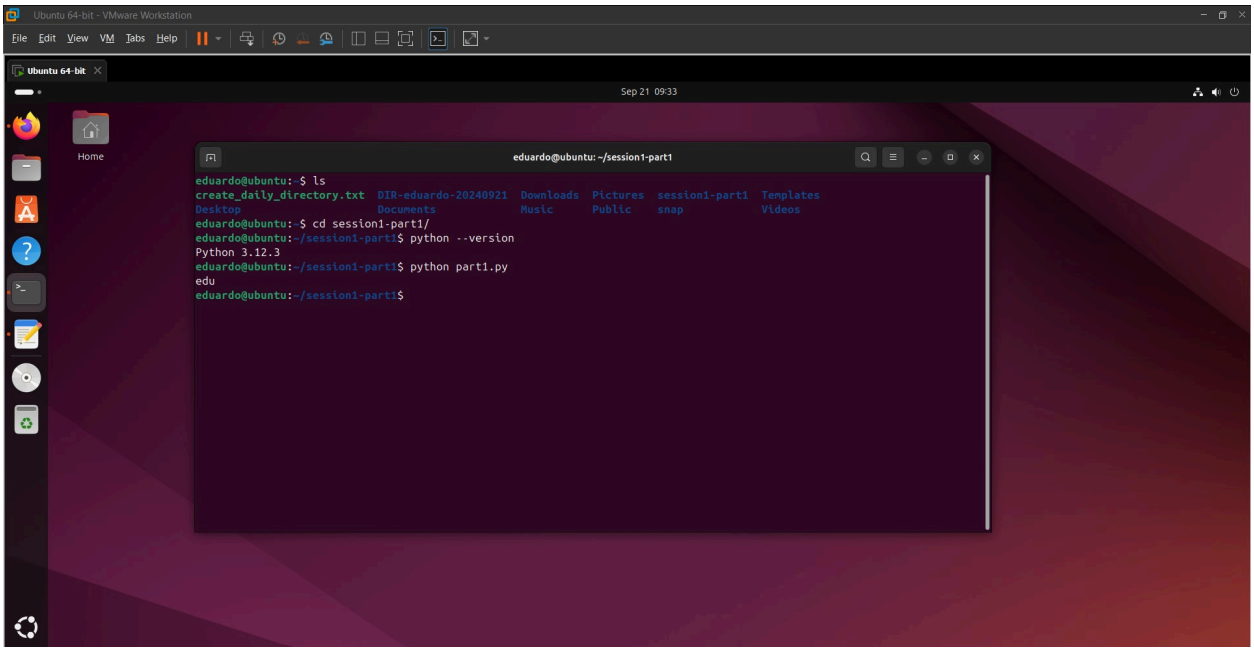
```
print("edu")
```

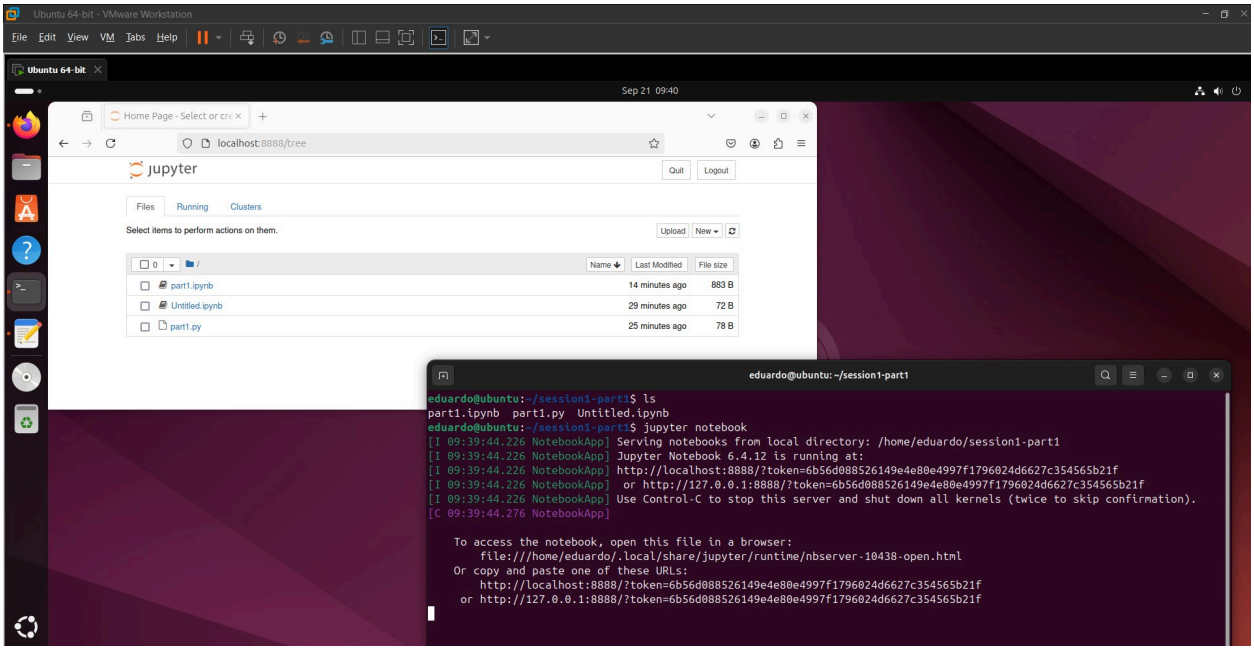
4. To **save your notebook as a Python script**:

- Go to **File > Download as > Python (.py)**
- Move the file: `mv ~/Downloads/part1.py .`
- Example content of `part1.py`:

  ```
  #!/usr/bin/env python
  print("edu")
  ```

- Run it with: `python part1.py`





Step 5: Access Jupyter Notebook Server Remotely

1. **To allow remote access:**

`jupyter notebook --port 8889 --allow-root --no-browser --ip=0.0.0.0`

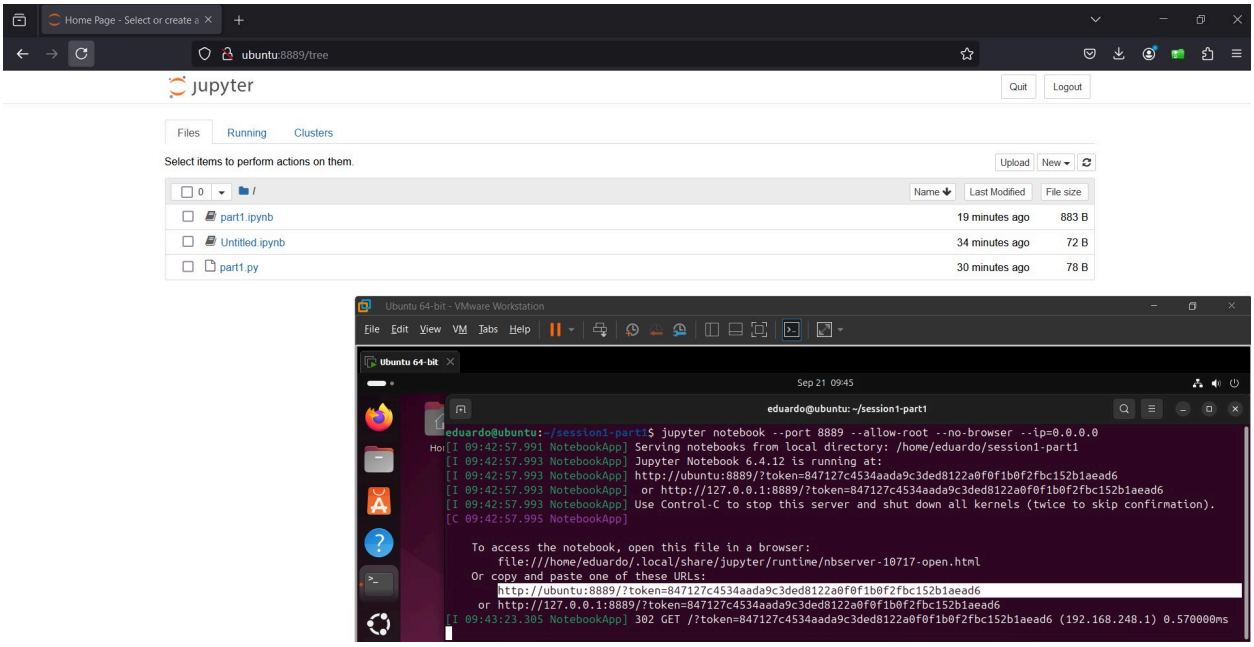
Explanation:

- `--ip=0.0.0.0`: Listens on all network interfaces (your local IP, public IP, etc.).
- `--no-browser`: Prevents it from auto-opening in your local browser.
- `--port=8889`: Sets a custom port.
- `--allow-root`: Allows running as root (not recommended unless necessary).

When you run the command, Jupyter will show a URL like this:

> http://ubuntu:8889/?token=XXXXXXXXXXXXXXX

- Copy and share link to remotely access.
- Below is a screenshot of my VM running Jupyter:



Note:

- Jupyter is best for interactive development experimentation, line by line, but not production use.
- You can export notebooks as `.py` files to reuse the code in other environments.
- Make sure your VM network settings allow for port access if using remote Jupyter.
