Ubuntu Server Setup Cheat Sheet

1️⃣ Install and Configure SSH

Install SSH Server

sudo apt update && sudo apt install -y openssh-server
sudo systemctl enable ssh
sudo systemctl start ssh

Check SSH Status

systemctl status ssh

Find Ubuntu IP Address

ip a | grep inet

Connect from Windows

ssh your-username@192.168.x.x

Replace `your-username` and `192.168.x.x` with your actual username and IP.

2️⃣ Set Up Shared Folder (Windows ↔ Ubuntu)

Step 1: Create a Shared Folder in Windows

- Create a folder in Windows, e.g., `C:\SharedFolder`.
- Open **VirtualBox → Settings → Shared Folders → Add New Folder**.
    - **Folder Path**: `C:\SharedFolder`
    - **Folder Name**: `SharedFolder`

- **Auto-mount**: ✅ Checked
- **Make Permanent**: ✅ Checked

Step 2: Mount the Shared Folder in Ubuntu

1. Restart Ubuntu VM.

Check if auto-mounted:

```
ls /media
```

2.

If not, manually mount:

```
sudo mount -t vboxsf SharedFolder /mnt
```

3.

Access your shared folder:

```
cd /mnt
ls
```

4.

3️⃣ Port Forwarding for SSH (If Using NAT)

Step 1: Configure VirtualBox Port Forwarding

1. **VirtualBox → Settings → Network → Adapter 1 (NAT) → Advanced → Port Forwarding**.
2. **Add New Rule**:
   - **Name**: `SSH`
   - **Protocol**: `TCP`

- **Host IP**: `127.0.0.1`
- **Host Port**: `2222`
- **Guest IP**: *(Leave blank)*
- **Guest Port**: `22`

3. Restart Ubuntu VM.

Step 2: Connect via SSH Using Port Forwarding

ssh -p 2222 your-username@127.0.0.1

4️⃣ Use Bridged Networking for Easier Access

1. **VirtualBox** → **Settings** → **Network** → Change `Attached to:` from **NAT** to **Bridged Adapter**.

Restart Ubuntu VM and check new IP:

ip a

2.

Now SSH directly without port forwarding:

ssh your-username@<new-ip>

3.

5️⃣ Install Development Tools

Basic Tools

sudo apt install -y curl wget git vim build-essential unzip

Node.js & npm (via nvm)

```
curl -fsSL https://raw.githubusercontent.com/nvm-sh/nvm/v0.39.4/install.sh | bash
source ~/.bashrc
nvm install --lts
nvm use --lts
```

Python, pip & virtualenv

```
sudo apt install -y python3 python3-pip
pip3 install --upgrade pip
pip3 install virtualenv
```

MySQL Server

```
sudo apt install -y mysql-server
sudo systemctl enable mysql
sudo systemctl start mysql
```

PostgreSQL

```
sudo apt install -y postgresql postgresql-contrib
sudo systemctl enable postgresql
sudo systemctl start postgresql
```

Docker & Docker Compose

```
sudo apt install -y docker.io docker-compose
sudo systemctl enable docker
sudo systemctl start docker
sudo usermod -aG docker $USER
```

Nginx & Firewall Setup

```
sudo apt install -y nginx
sudo systemctl enable nginx
sudo systemctl start nginx
sudo ufw allow 'Nginx Full'
sudo ufw allow OpenSSH
sudo ufw enable
```

🎉 You're All Set! 🎉