Of course! Let's properly organize everything for you:
Here’s a **full, clean, step-by-step guide** (including errors you faced and their solutions) for:

Task: Implement Network Virtualization using VirtualBox on Windows 11

🛠 Full Steps to Implement Network Virtualization Using VirtualBox

1. Install VirtualBox

- Download from VirtualBox Official Site.
- Install it normally on your Windows 11 machine.

Done

2. Create Two Virtual Machines (VMs)

- Click "New" → Create two VMs (Ubuntu OS recommended for practice).
- Example:
- VM1: `Ubuntu-VM1`
- VM2: `Ubuntu-VM2`

Done

3. Configure Network Setting

Set both VMs to the same Internal Network:

- Open VirtualBox → Select your VM → Settings → Network
- Adapter 1:
- Enable Network Adapter →
- Attached to: **Internal Network**
- Name: `intnet`
- Do this for **both** VMs!

Done

4. Start and Login into VMs

- Start VM1 and VM2.
- Login into both machines.

Done

5. Become a sudo user (Handle Permissions)

Error 1:

```
vboxuser is not in the sudoers file. This incident will be reported.
```

Reason: You are not an admin user.

Solution:

- Either login as root:
- `su -`
-
- Or add user to `sudo` group:
- `su -`
- `usermod -aG sudo vboxuser`
- `logout`
-
- Then login again and you will have `sudo` access.

Done

6. Find the Correct Network Interface Name

Run inside each VM:

```
ip a
```

You will see something like:

```
enp0s3
```

Note it! (NOT `eth0` in new systems.)

7. Assign Static IP Addresses

Inside **VM1**:

```
sudo ip addr add 192.168.1.10/24 dev enp0s3
```

Inside **VM2**:

```
sudo ip addr add 192.168.1.20/24 dev enp0s3
```

(Replace `enp0s3` if your device name is different.)

Error 2:

```
cannot find device 'eth0'
```

Reason: No device named `eth0` exists on new Ubuntu systems.

Solution:

- Use correct device name found by `ip a`, such as `enp0s3`.

Done

8. Verify IP Assignment

Run inside each VM:

```
ip a
```

Check that:

- VM1 has `192.168.1.10`
- VM2 has `192.168.1.20`

Done

9. Test the Virtual Network (Ping Test)

From **VM1**, ping VM2:

```
ping 192.168.1.20
```

From **VM2**, ping VM1:

```
ping 192.168.1.10
```

If you get continuous "64 bytes from..." replies → SUCCESS.

10. Troubleshooting Ping Failure (Optional)

If `ping` fails:

a) Check if VMs are on same "Internal Network"

- VirtualBox → Settings → Network → Adapter 1 → Internal Network → Name `intnet`

b) Disable firewall inside VM:

```
sudo ufw disable
```

Done

📋 Quick Summary Table

Step	Command / Action	Purpose
Install VirtualBox	Download + Install	Setup environment
Create 2 VMs	New VM (Ubuntu OS)	Create test machines
Set Internal Network	Adapter 1 → Internal Network (`intnet`)	Private network
Become sudo user	`su -` or `usermod -aG sudo user`	Gain admin rights
Find network interface	`ip a`	Find correct device name
Assign IP	`sudo ip addr add 192.168.1.x/24 dev enp0s3`	Setup manual IP
Verify IPs	`ip a`	Confirm
Ping Test	`ping other_vm_ip`	Test communication
Troubleshoot	Disable firewall	If ping fails

Final Diagram (ASCII Style)

```
+--------------------------------+       +--------------------------------+
| Ubuntu-VM1                     |       | Ubuntu-VM2                     |
| IP: 192.168.1.10/24             | <-->  | IP: 192.168.1.20/24             |
| Interface: enp0s3 (InternalNet) |       | Interface: enp0s3 (InternalNet) |
+--------------------------------+       +--------------------------------+
         ↖            VirtualBox Internal Network          ↗
```

`(intnet)`

Important Notes:

- Internal Network (intnet) is isolated from external internet (no browsing unless you configure NAT additionally).
- IPs must be in same subnet: `192.168.1.x/24`
- Interface names differ depending on Linux version.

Congratulations!

You have implemented **basic network virtualization** using **VirtualBox** on **Windows 11**!

Would you also like me to give you a **very small assignment** (optional) to test your virtualization setup? It will be like setting up a small web server between two VMs!
Just say "**yes, give assignment**"!