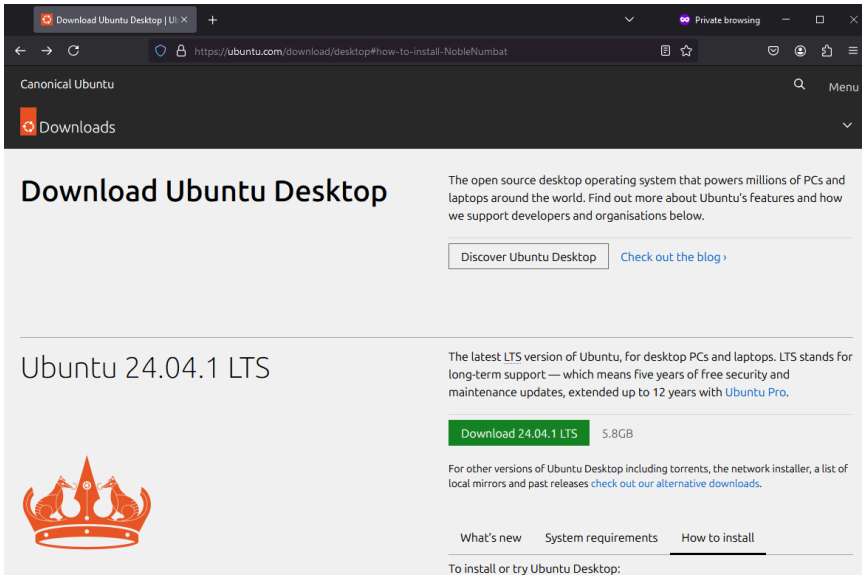
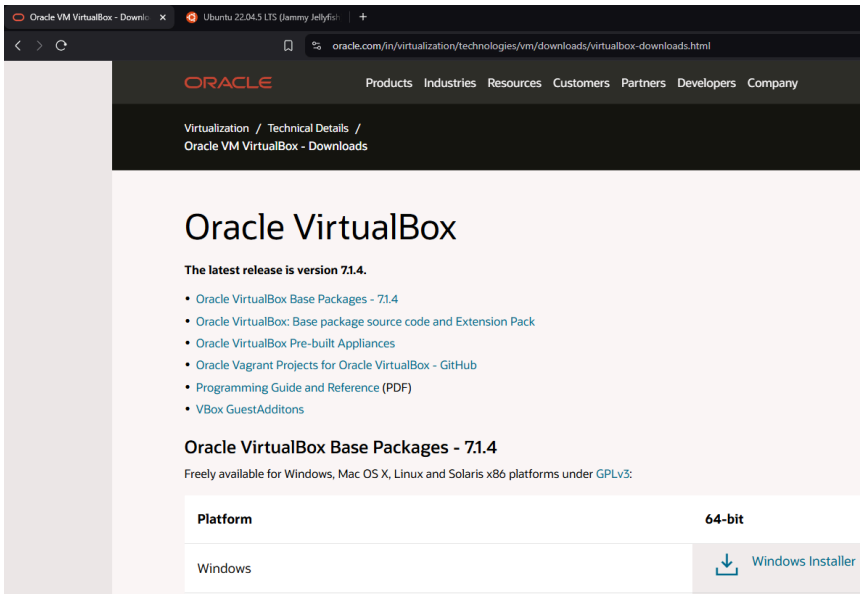
Practical 1

Part A: Installation of Ubuntu

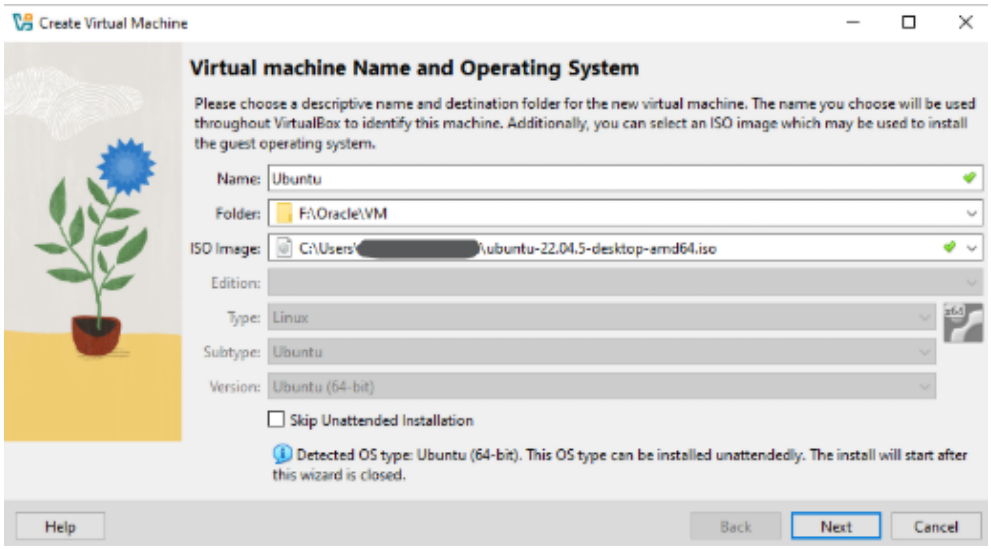
Step 1: Download Ubuntu ISO file



Step 2: Download and Install VirtualBox

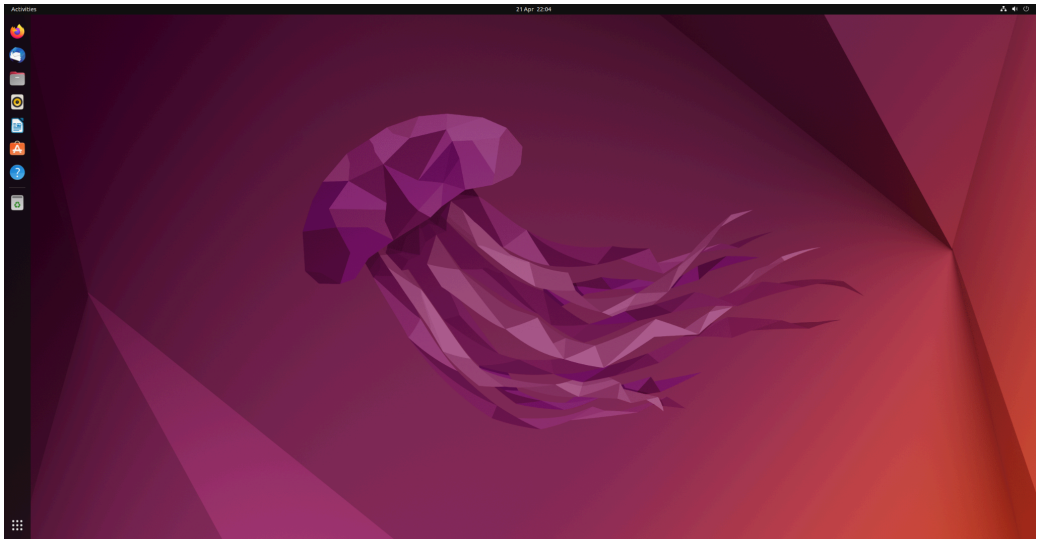


Step 3: Create a new Virtual Machine. Select the location of the .iso file.



Choose the required specifications for your VM.

Step 4: The Virtual Machine is installed and ready to use.



Part B: Choice based shell script

emp.txt

```
system@system-virtual-machine:~$ cat > emp.txt
abc 123 computer 10000 7000
mno 345 management 14000 10000
pqr 532 hr 15000 12000
xyz 143 computer 10000 7000
asd 756 sales 7000 6000
```

(write script and stick output)

```
  GNU nano 6.2                          menu.sh
#!/bin/bash


count_employees() {
    count=$(grep "technical" emp.txt | wc -l)
    echo "Total number of employees in 'technical' department: $count"
}

machines_required() {
    count=$(grep "technical" emp.txt | wc -l)
    echo "Number of machines required: $count"
}

menu() {
    echo "Menu options"
    echo "************************************"
    echo "1) Total number of employees in 'technical' department"
    echo "2) Number of machines required"
    echo "3) Exit"
}

while true; do
    menu
    read -p "Select an option (1-3): " choice

    case $choice in
        1) count_employees ;;
        2) machines_required ;;
        3) echo "Exiting..."; exit 0 ;;
        *) echo "Invalid option. Please try again." ;;
    esac

    read -p "Do you wish to continue? (y/n): " continue_choice
    if [[ "$continue_choice" != "y" ]]; then
        break
    fi
done
```

```
system@system-virtual-machine:~$ ./menu.sh
Menu options
**************************************
1) Total number of employees in 'technical' department
2) Number of machines required
3) Exit
Select an option (1-3): 1
Total number of employees in 'technical' department: 3
Do you wish to continue? (y/n): y
Menu options
**************************************
1) Total number of employees in 'technical' department
2) Number of machines required
3) Exit
Select an option (1-3): 2
Number of machines required: 3
Do you wish to continue? (y/n): y
Menu options
**************************************
1) Total number of employees in 'technical' department
2) Number of machines required
3) Exit
Select an option (1-3): 3
Exiting...
system@system-virtual-machine:~$
```

Practical 2

```
#!/bin/bash

display_emps(){
echo "Employees:"
grep $dept emp.txt
}

file_links(){
ls -l | awk '$2 > 1 {print $2, $9}'
}

while true;
do
echo "Menu:"
echo "1) Display specific dept emp"
echo "2) List files having more than 1 link"
echo "3) Exit"
read -p "Select option (1-4):" choice

case $choice in
1) read -p "Enter dept:" dept; display_emps;;
2) file_links;;
3) echo "Exiting"; exit 0;;
*) echo "Invalid option";;
esac
done
```

```
system@Ubuntu:~$ ./shsc.sh
Menu:
1) Display specific dept emp
2) List files having more than 1 link
3) Exit
Select option (1-4):1
Enter dept:comp
Employees:
101   abc     comp 10000 20000
105   mno     comp 20000 25000
201   zxc     comp 20000 30000
Menu:
1) Display specific dept emp
2) List files having more than 1 link
3) Exit
Select option (1-4):2
52
2 Desktop
2 Documents
2 Downloads
2 emp2.txt
2 emp.txt
2 Music
2 Pictures
2 Public
4 snap
2 Templates
2 Videos
Menu:
1) Display specific dept emp
2) List files having more than 1 link
3) Exit
Select option (1-4):3
Exiting
system@Ubuntu:~$
```

Practical 3

Part A - Networking Commands
NFS (Network File System)
] ifconfig/ipconfig
] ping
] route
] traceroute
] netstat | more
] whoami
] hostname
] ifplugstatus
] dig (Domain Information Groper)
-> Discuss: Explain Header, Answer, Statistics Sections
-> Illustrate: Give example

Part B
SAMBA server - to implement NFS
FTP

Practical 4

Part A: Choice-based shell script
You are working as a LSA, write the for following network commands

1. Check current ubuntu machine is working
2. Check the underlying windows machine is working and ubuntu and windows able to communicate
3. Send only 10 packets to www.google.com and check if it is up and running
4. Change the frequency of sending the packet by 3 millisecond
5. Find out the network statistics
6. Query if www.facebook.com is up and running
7. Display current hostname
8. Accept the name of the directory from the user and create a directory with the name specified
9. Display the current user
10. Ask the user if they wish to continue

```
while true;
do
    echo "Menu:"
    echo "1. Check current ubuntu machine is working"
    echo "2. Check the underlying windows machine is working and
ubuntu and windows able to communicate"
    echo "3. Send only 10 packets to www.google.com and check if
it is up and running"
    echo "4. Change the frequency of sending the packet by 3
millisecond"
    echo "5. Check network statistics"
    echo "6. Query if www.facebook.com is up and running"
    echo "7. Display current hostname"
    echo "8. Accept the name of the directory from the user and
create a directory with the name specified"
    echo "9. Display current user"
    echo "10. Exit"
    read -p "Enter your choice: " choice
```

```
    case $choice in
        1) ping -c 4 localhost;;
        2) ping -c 4 172.23.0.73;;
        3) ping -c 10 google.com;;
        4) ping -c 4 -i 0.003 localhost;;
        5) netstat -s;;
        6) dig www.facebook.com;;
        7) hostname;;
        8) read -p "Enter directory name: " dirname && mkdir -p
"$dirname" && echo "Directory '$dirname' created.";;
        9) whoami;;
        10) echo "Exiting..."; exit 0;;
        *) echo "Invalid choice";;
    esac

    read -p "Do you want to continue? (y/n): " cont
    [[ $cont != "y" ]] && break
done
```

```
Menu:
1. Check current ubuntu machine is working
2. Check the underlying windows machine is working and ubuntu and
windows able to communicate
3. Send only 10 packets to www.google.com and check if it is up an
d running
4. Change the frequency of sending the packet by 3 millisecond
5. Check network statistics
6. Query if www.facebook.com is up and running
7. Display current hostname
8. Accept the name of the directory from the user and create a dir
ectory with the name specified
9. Display current user
10. Exit
Enter your choice: 5
Ip:
    Forwarding: 2
    1864 total packets received
    1 with invalid addresses
    0 forwarded
    0 incoming packets discarded
    1861 incoming packets delivered
    1603 requests sent out
    20 outgoing packets dropped
    OutTransmits: 1603
Icmp:
    91 ICMP messages received
    0 input ICMP message failed
    ICMP input histogram:
```

Part B: SSH

Install SSH Server: `sudo apt-get install openssh-server`

```
system@system-virtual-machine:~$ ssh
usage: ssh [-46AaCfGgKkMNnqsTtVvXxYy] [-B bind_interface]
           [-b bind_address] [-c cipher_spec] [-D [bind_address:]port]
           [-E log_file] [-e escape_char] [-F configfile] [-I pkcs11]
           [-i identity_file] [-J [user@]host[:port]] [-L address]
           [-l login_name] [-m mac_spec] [-O ctl_cmd] [-o option] [-p port]
           [-Q query_option] [-R address] [-S ctl_path] [-W host:port]
           [-w local_tun[:remote_tun]] destination [command [argument ...]]
system@system-virtual-machine:~$ sudo apt-get install openssh-server
Reading package lists... Done
Building dependency tree... Done
Reading state information... Done
openssh-server is already the newest version (1:8.9p1-3ubuntu0.10).
0 upgraded, 0 newly installed, 0 to remove and 182 not upgraded.
```

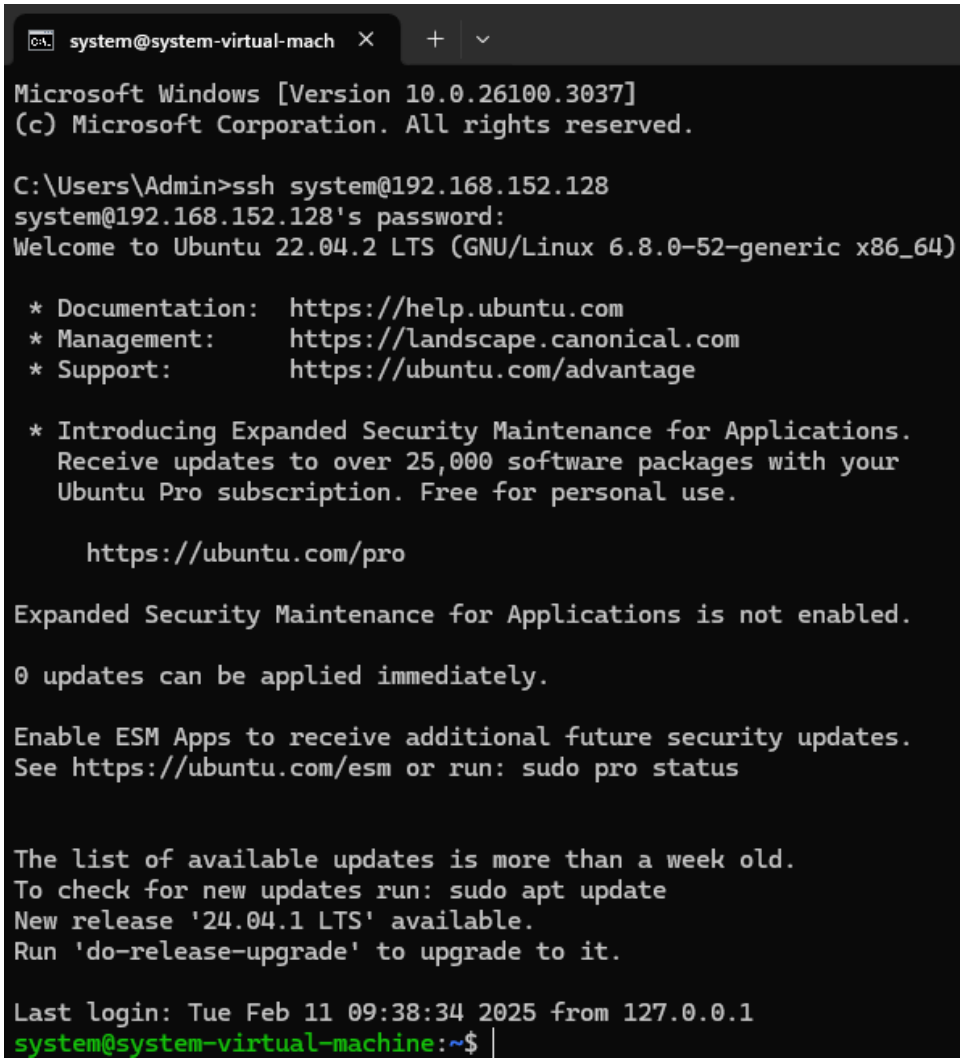
```
system@system-virtual-machine:~$ sudo service ssh status
● ssh.service - OpenBSD Secure Shell server
     Loaded: loaded (/lib/systemd/system/ssh.service; enabled; vendor preset: enabled)
     Active: active (running) since Tue 2025-02-11 08:40:14 IST; 48min ago
       Docs: man:sshd(8)
             man:sshd_config(5)
    Process: 930 ExecStartPre=/usr/sbin/sshd -t (code=exited, status=0/SUCCESS)
   Main PID: 956 (sshd)
      Tasks: 1 (limit: 4551)
     Memory: 2.9M
        CPU: 61ms
     CGroup: /system.slice/ssh.service
             └─956 "sshd: /usr/sbin/sshd -D [listener] 0 of 10-100 startups"

Feb 11 08:40:14 system-virtual-machine systemd[1]: Starting OpenBSD Secure Shell serve>
Feb 11 08:40:14 system-virtual-machine sshd[956]: Server listening on 0.0.0.0 port 22.
Feb 11 08:40:14 system-virtual-machine systemd[1]: Started OpenBSD Secure Shell server.
Feb 11 08:40:14 system-virtual-machine sshd[956]: Server listening on :: port 22.
lines 1-17/17 (END)
```

Discuss service command (can use this example to explain)

```
system@system-virtual-machine:~$ sudo service ssh status
○ ssh.service - OpenBSD Secure Shell server
     Loaded: loaded (/lib/systemd/system/ssh.service; enabled; vendor preset: enabled)
     Active: inactive (dead) since Tue 2025-02-11 09:31:12 IST; 33s ago
       Docs: man:sshd(8)
             man:sshd_config(5)
    Process: 930 ExecStartPre=/usr/sbin/sshd -t (code=exited, status=0/SUCCESS)
    Process: 956 ExecStart=/usr/sbin/sshd -D $SSHD_OPTS (code=exited, status=0/SUCCESS)
   Main PID: 956 (code=exited, status=0/SUCCESS)
        CPU: 63ms

Feb 11 08:40:14 system-virtual-machine systemd[1]: Starting OpenBSD Secure Shell serve>
Feb 11 08:40:14 system-virtual-machine sshd[956]: Server listening on 0.0.0.0 port 22.
Feb 11 08:40:14 system-virtual-machine systemd[1]: Started OpenBSD Secure Shell server.
Feb 11 08:40:14 system-virtual-machine sshd[956]: Server listening on :: port 22.
Feb 11 09:31:12 system-virtual-machine sshd[956]: Received signal 15; terminating.
Feb 11 09:31:12 system-virtual-machine systemd[1]: Stopping OpenBSD Secure Shell serve>
Feb 11 09:31:12 system-virtual-machine systemd[1]: ssh.service: Deactivated successful>
Feb 11 09:31:12 system-virtual-machine systemd[1]: Stopped OpenBSD Secure Shell server.
lines 1-18/18 (END)
```

```
system@system-virtual-machine:~$ ssh localhost
system@localhost's password:
Welcome to Ubuntu 22.04.2 LTS (GNU/Linux 6.8.0-52-generic x86_64)

 * Documentation:  https://help.ubuntu.com
 * Management:     https://landscape.canonical.com
 * Support:        https://ubuntu.com/advantage

 * Introducing Expanded Security Maintenance for Applications.
   Receive updates to over 25,000 software packages with your
   Ubuntu Pro subscription. Free for personal use.

     https://ubuntu.com/pro

Expanded Security Maintenance for Applications is not enabled.

0 updates can be applied immediately.

Enable ESM Apps to receive additional future security updates.
See https://ubuntu.com/esm or run: sudo pro status


The list of available updates is more than a week old.
To check for new updates run: sudo apt update
Last login: Mon Feb 10 09:22:52 2025 from 192.168.152.1
```

```
system@system-virtual-machine:~$ ping -c 4 www.google.com
PING www.google.com (172.217.174.68) 56(84) bytes of data.
64 bytes from bom07s25-in-f4.1e100.net (172.217.174.68): icmp_seq=1
ttl=128 time=4.65 ms
64 bytes from bom07s25-in-f4.1e100.net (172.217.174.68): icmp_seq=2
ttl=128 time=4.82 ms
64 bytes from bom07s25-in-f4.1e100.net (172.217.174.68): icmp_seq=3
ttl=128 time=4.79 ms
64 bytes from bom07s25-in-f4.1e100.net (172.217.174.68): icmp_seq=4
ttl=128 time=4.82 ms

--- www.google.com ping statistics ---
4 packets transmitted, 4 received, 0% packet loss, time 3005ms
rtt min/avg/max/mdev = 4.654/4.771/4.824/0.068 ms
system@system-virtual-machine:~$ logout
Connection to localhost closed.
```


system@system-virtual-mach
Microsoft Windows [Version 10.0.26100.3037]
(c) Microsoft Corporation. All rights reserved.
C:\Users\Admin>ssh system@192.168.152.128
system@192.168.152.128's password:
Welcome to Ubuntu 22.04.2 LTS (GNU/Linux 6.8.0-52-generic x86_64)
* Documentation: https://help.ubuntu.com
* Management: https://landscape.canonical.com
* Support: https://ubuntu.com/advantage
* Introducing Expanded Security Maintenance for Applications.
Receive updates to over 25,000 software packages with your
Ubuntu Pro subscription. Free for personal use.
https://ubuntu.com/pro
Expanded Security Maintenance for Applications is not enabled.
0 updates can be applied immediately.
Enable ESM Apps to receive additional future security updates.
See https://ubuntu.com/esm or run: sudo pro status
The list of available updates is more than a week old.
To check for new updates run: sudo apt update
New release '24.04.1 LTS' available.
Run 'do-release-upgrade' to upgrade to it.
Last login: Tue Feb 11 09:38:34 2025 from 127.0.0.1
system@system-virtual-machine:~$

Practical 5

FTP(File Transfer Protocol):

- It is a standard way to move files between computers on a network.
- It's used to upload and download files, and to transfer data between computers on a local network.
- FTP uses a client-server model.
- It opens two connections between the devices- one connection is for commands and replies, and the other is for data transfer.

Step 1: Install FTP server

```
sudo apt-get install vsftpd
```

```
system@server:~$ sudo apt-get install vsftpd
Reading package lists... Done
Building dependency tree... Done
Reading state information... Done
vsftpd is already the newest version (3.0.5-0ubuntu1.1).
0 upgraded, 0 newly installed, 0 to remove and 184 not upgraded.
```

Step 2: Note down the IP address of your system

```
ip a
```

```
system@server:~$ ip a
1: lo: <LOOPBACK,UP,LOWER_UP> mtu 65536 qdisc noqueue state UNKNOWN group
 default qlen 1000
    link/loopback 00:00:00:00:00:00 brd 00:00:00:00:00:00
    inet 127.0.0.1/8 scope host lo
       valid_lft forever preferred_lft forever
    inet6 ::1/128 scope host
       valid_lft forever preferred_lft forever
2: ens33: <BROADCAST,MULTICAST,UP,LOWER_UP> mtu 1500 qdisc fq_codel state
 UP group default qlen 1000
    link/ether 00:0c:29:f5:01:5d brd ff:ff:ff:ff:ff:ff
    altname enp2s1
    inet 192.168.152.128/24 brd 192.168.152.255 scope global dynamic nopr
efixroute ens33
       valid_lft 1692sec preferred_lft 1692sec
    inet6 fe80::9187:c084:c428:15f7/64 scope link noprefixroute
       valid_lft forever preferred_lft forever
```

Step 3: Create a User account

```
sudo adduser demo_tycs
```

```
system@server:~$ sudo adduser demo_tycs
Adding user `demo_tycs' ...
Adding new group `demo_tycs' (1004) ...
Adding new user `demo_tycs' (1004) with group `demo_tycs' ...
Creating home directory `/home/demo_tycs' ...
Copying files from `/etc/skel' ...
New password:
Retype new password:
passwd: password updated successfully
Changing the user information for demo_tycs
Enter the new value, or press ENTER for the default
        Full Name []: tycs_user1
        Room Number []: 2
        Work Phone []: 1234567890
        Home Phone []:
        Other []:
Is the information correct? [Y/n] Y
```

Step 4: Switch to the newly created user

`sudo -i -u demo_tycs`

```
system@server:~$ pwd
/home/system
system@server:~$ sudo -i -u demo_tycs
demo_tycs@server:~$ pwd
/home/demo_tycs
```

Step 5: Create the FTP folder

`mkdir test`

```
demo_tycs@server:~$ mkdir test
demo_tycs@server:~$ ls -l
total 4
drwxrwxr-x 2 demo_tycs demo_tycs 4096 Feb 25 08:38 test
```

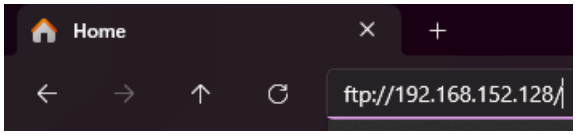
Step 6: Add content to the file and set permissions

`cat > demo.txt`

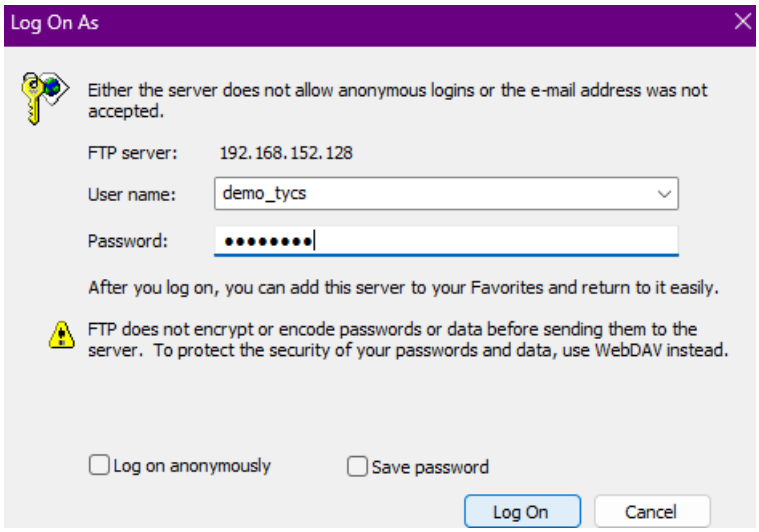
`chmod 777 demo.txt`

```
demo_tycs@server:~/test$ cat > demo.txt
Demo for FTP.
demo_tycs@server:~/test$ ls -l
total 4
-rw-rw-r-- 1 demo_tycs demo_tycs 14 Feb 25 08:58 demo.txt
demo_tycs@server:~/test$ cat demo.txt
Demo for FTP.
demo_tycs@server:~/test$ chmod 777 demo.txt
```

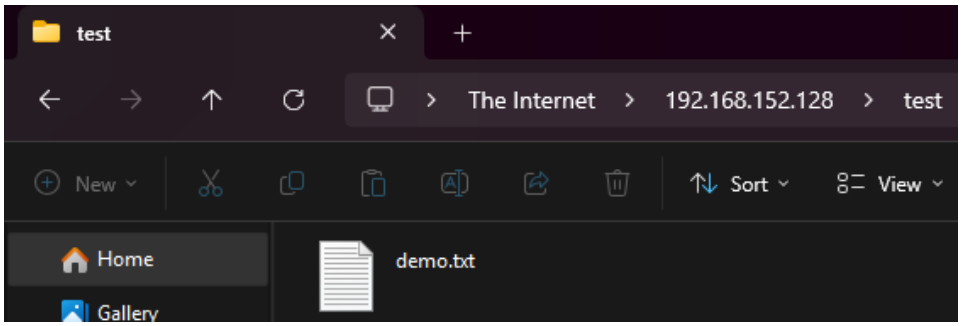
Step 7: On your host system (Windows), open file explorer and in the address bar, enter the ftp URL with your server's IP address.



Step 8: Enter the user account details created on the server.



The directories of the FTP server will be visible.



Practical 6

NFS (Network File System):

- NFS is a distributed file system protocol used in linux for sharing files and directories across networks.
- It allows users to access remote files as if they were stored locally on their machines.
- NFS operates on a client-server model, where the server hosts the shared resources and the client accesses them.
- NFS relies on RPCs (Remote Procedure calls) for facilitating operations like reading and writing.
- Key benefits for using NFS, provides:
  a. centralized Data Management.
  b. Efficient Disk Space usage.
  c. Simplified user Management.

1. Setting up NFS server

Step 1: Install NFS server

```
sudo apt-get install nfs-kernel-server
```

```
system@server:~$ sudo apt-get install nfs-kernel-server
Reading package lists... Done
Building dependency tree... Done
Reading state information... Done
nfs-kernel-server is already the newest version (1:2.6.1-1ubuntu1.2).
0 upgraded, 0 newly installed, 0 to remove and 184 not upgraded.
```

Step 2: Make 2 directories at root level, named ‘public’ and ‘private’, and given them permissions

```
system@server:~$ mkdir public
system@server:~$ mkdir private
system@server:~$ ls -l public
total 0
system@server:~$ ls -l private
total 0
system@server:~$ chmod 777 public
system@server:~$ chmod 777 private
```

Step 3: Open the /etc/exports file and add the last two lines

```
sudo nano /etc/exports
```

```
  GNU nano 6.2                          /etc/exports *
# /etc/exports: the access control list for filesystems which may be exported
#               to NFS clients.  See exports(5).
#
# Example for NFSv2 and NFSv3:
# /srv/homes       hostname1(rw,sync,no_subtree_check) hostname2(ro,sync,no_sub>
#
# Example for NFSv4:
# /srv/nfs4        gss/krb5i(rw,sync,fsid=0,crossmnt,no_subtree_check)
# /srv/nfs4/homes  gss/krb5i(rw,sync,no_subtree_check)
#
/public *(ro,sync,no_subtree_check)
/private 192.168.152.128/24(rw,sync,no_subtree_check)
```

Step 4: Export the directories and Start the NFS server

```
sudo exportfs -avrf
sudo systemctl start nfs-kernel-server
sudo systemctl enable nfs-kernel-server
sudo systemctl status nfs-kernel-server
```

```
system@server:~$ sudo exportfs -arvf
exporting 192.168.152.128/24:/private
exporting *:/public
system@server:~$ sudo systemctl start nfs-kernel-server
system@server:~$ sudo systemctl enable nfs-kernel-server
Synchronizing state of nfs-kernel-server.service with SysV service script with /
lib/systemd/systemd-sysv-install.
Executing: /lib/systemd/systemd-sysv-install enable nfs-kernel-server
system@server:~$ sudo systemctl status nfs-kernel-server
● nfs-server.service - NFS server and services
     Loaded: loaded (/lib/systemd/system/nfs-server.service; enabled; vendor pr>
    Drop-In: /run/systemd/generator/nfs-server.service.d
             └─order-with-mounts.conf
     Active: active (exited) since Mon 2025-02-24 09:24:05 IST; 24h ago
   Main PID: 6122 (code=exited, status=0/SUCCESS)
        CPU: 20ms

Feb 24 09:24:05 server.example.com systemd[1]: Starting NFS server and services>
Feb 24 09:24:05 server.example.com exportfs[6121]: exportfs: can't open /etc/ex>
Feb 24 09:24:05 server.example.com systemd[1]: Finished NFS server and services.
lines 1-11/11 (END)
```

2. Setting up NFS Client

Step 1: In a new terminal, install NFS client package

```
sudo apt-get install nfs-common
```

```
system@server:~$ sudo apt-get install nfs-common
Reading package lists... Done
Building dependency tree... Done
Reading state information... Done
nfs-common is already the newest version (1:2.6.1-1ubuntu1.2).
0 upgraded, 0 newly installed, 0 to remove and 184 not upgraded.
```

Step 2: Check for directories shared via the NFS

`showmount -e 192.168.152.128`

```
system@server:~$ showmount -e 192.168.152.128
Export list for 192.168.152.128:
/public  *
/private 192.168.152.128/24
```

Step 3: Create directories in 'mnt' directory under root

`sudo mkdir /mnt/public`
`sudo mkdir /mnt/private`

```
system@server:~$ sudo mkdir /mnt/public
system@server:~$ sudo mkdir /mnt/private
```

Step 4: Mount the shared directories from server to local directories

`sudo mount -t nfs 192.168.152.128:/public /mnt/public`
`sudo mount -t nfs 192.168.152.128:/private /mnt/private`

```
system@server:~$ sudo mount -t nfs 192.168.152.128:/public /mnt/public
system@server:~$ sudo mount -t nfs 192.168.152.128:/private /mnt/private
```

Step 5: Verify the files have mounted on the client

`mount`

```
system@server:~$ mount
sysfs on /sys type sysfs (rw,nosuid,nodev,noexec,relatime)
proc on /proc type proc (rw,nosuid,nodev,noexec,relatime)
udev on /dev type devtmpfs (rw,nosuid,relatime,size=1941864k,nr_inodes=485466,mo
de=755,inode64)
devpts on /dev/pts type devpts (rw,nosuid,noexec,relatime,gid=5,mode=620,ptmxmod
e=000)
tmpfs on /run type tmpfs (rw,nosuid,nodev,noexec,relatime,size=396140k,mode=755,
inode64)
/dev/sda3 on / type ext4 (rw,relatime,errors=remount-ro)
securityfs on /sys/kernel/security type securityfs (rw,nosuid,nodev,noexec,relat
ime)
```

```
nsfs on /run/snapd/ns/snap-store.mnt type nsfs (rw)
/var/lib/snapd/snaps/core20_2496.snap on /snap/core20/2496 type squashfs (ro,nod
ev,relatime,errors=continue,threads=single,x-gdu.hide)
sunrpc on /run/rpc_pipefs type rpc_pipefs (rw,relatime)
nfsd on /proc/fs/nfsd type nfsd (rw,relatime)
192.168.152.128:/public on /mnt/private type nfs4 (rw,relatime,vers=4.2,rsize=52
4288,wsize=524288,namlen=255,hard,proto=tcp,timeo=600,retrans=2,sec=sys,clientad
dr=192.168.152.128,local_lock=none,addr=192.168.152.128)
192.168.152.128:/public on /mnt/public type nfs4 (rw,relatime,vers=4.2,rsize=524
288,wsize=524288,namlen=255,hard,proto=tcp,timeo=600,retrans=2,sec=sys,clientadd
r=192.168.152.128,local_lock=none,addr=192.168.152.128)
```

Practical 7

MySQL

Step 1: Install MySQL packages

```
system@server:~$ sudo apt-get install mysql-server-*
Reading package lists... Done
Building dependency tree... Done
Reading state information... Done
Note, selecting 'mysql-server-5.5' for glob 'mysql-server-*'
Note, selecting 'mysql-server-5.6' for glob 'mysql-server-*'
Note, selecting 'mysql-server-5.7' for glob 'mysql-server-*'
Note, selecting 'mysql-server-8.0' for glob 'mysql-server-*'
Note, selecting 'mysql-server-core-5.5' for glob 'mysql-server-*'
Note, selecting 'mysql-server-core-5.6' for glob 'mysql-server-*'
Note, selecting 'mysql-server-core-5.7' for glob 'mysql-server-*'
Note, selecting 'mysql-server-core-8.0' for glob 'mysql-server-*'
mysql-server-8.0 is already the newest version (8.0.41-0ubuntu0.22.04.1).
mysql-server-core-8.0 is already the newest version (8.0.41-0ubuntu0.22.04.1).
0 upgraded, 0 newly installed, 0 to remove and 181 not upgraded.
```

Step 2: Open up the MySQL prompt.

```
system@server:~$ sudo mysql
Welcome to the MySQL monitor.  Commands end with ; or \g.
Your MySQL connection id is 8
Server version: 8.0.41-0ubuntu0.22.04.1 (Ubuntu)

Copyright (c) 2000, 2025, Oracle and/or its affiliates.

Oracle is a registered trademark of Oracle Corporation and/or its
affiliates. Other names may be trademarks of their respective
owners.

Type 'help;' or '\h' for help. Type '\c' to clear the current input statement.

mysql>
```

(write the queries in the images)

```
mysql> create database test;
Query OK, 1 row affected (0.01 sec)

mysql> use test;
Database changed
```

```
mysql> create table student(name varchar(10), rollno varchar(5));
Query OK, 0 rows affected (0.04 sec)

mysql> alter table student add column (m1 int, m2 int, m3 int, total int);
Query OK, 0 rows affected (0.06 sec)
Records: 0  Duplicates: 0  Warnings: 0

mysql> insert into student values ("abc","1",45,65,66,176),
    -> ("bcd","2",65,34,64,163), ("ghi","3",76,56,76,208),
    -> ("ijk","4",76,56,78,210), ("xyz","5",67,76,45,188);
Query OK, 5 rows affected (0.00 sec)
Records: 5  Duplicates: 0  Warnings: 0
```

```
mysql> select * from student;
+------+--------+------+------+------+-------+
| name | rollno | m1   | m2   | m3   | total |
+------+--------+------+------+------+-------+
| abc  | 1      |   45 |   65 |   66 |   176 |
| bcd  | 2      |   65 |   34 |   64 |   163 |
| ghi  | 3      |   76 |   56 |   76 |   208 |
| ijk  | 4      |   76 |   56 |   78 |   210 |
| xyz  | 5      |   67 |   76 |   45 |   188 |
+------+--------+------+------+------+-------+
5 rows in set (0.00 sec)

mysql> select * from student where total>200;
+------+--------+------+------+------+-------+
| name | rollno | m1   | m2   | m3   | total |
+------+--------+------+------+------+-------+
| ghi  | 3      |   76 |   56 |   76 |   208 |
| ijk  | 4      |   76 |   56 |   78 |   210 |
+------+--------+------+------+------+-------+
2 rows in set (0.00 sec)
```

Q] Create emp table with name ,id, basic salary, hra, da
Use update and add gross salary for all

```
mysql> use test
Database changed
mysql> create table emp(name varchar(5), id int, basicsal int, hra
int, sd int, da int, grosssal int);
Query OK, 0 rows affected (0.01 sec)

mysql> insert into emp values ("abc",1,50000,10000,5000,10000,null)
,("qwe",2,40000,10000,4000,10000,null),("asd",3,45000,75000,4500,75
000,null),("fgh",4,50000,10000,5000,10000,null);
Query OK, 4 rows affected (0.01 sec)
Records: 4  Duplicates: 0  Warnings: 0

mysql> update emp set grosssal=basicsal+hra+da+sd;
Query OK, 4 rows affected (0.01 sec)
Rows matched: 4  Changed: 4  Warnings: 0

mysql> select * from emp;
+------+------+----------+-------+------+-------+----------+
| name | id   | basicsal | hra   | sd   | da    | grosssal |
+------+------+----------+-------+------+-------+----------+
| abc  |    1 |    50000 | 10000 | 5000 | 10000 |    75000 |
| qwe  |    2 |    40000 | 10000 | 4000 | 10000 |    64000 |
| asd  |    3 |    45000 | 75000 | 4500 | 75000 |   199500 |
| fgh  |    4 |    50000 | 10000 | 5000 | 10000 |    75000 |
+------+------+----------+-------+------+-------+----------+
4 rows in set (0.00 sec)
```

Practical 8

phpMyAdmin

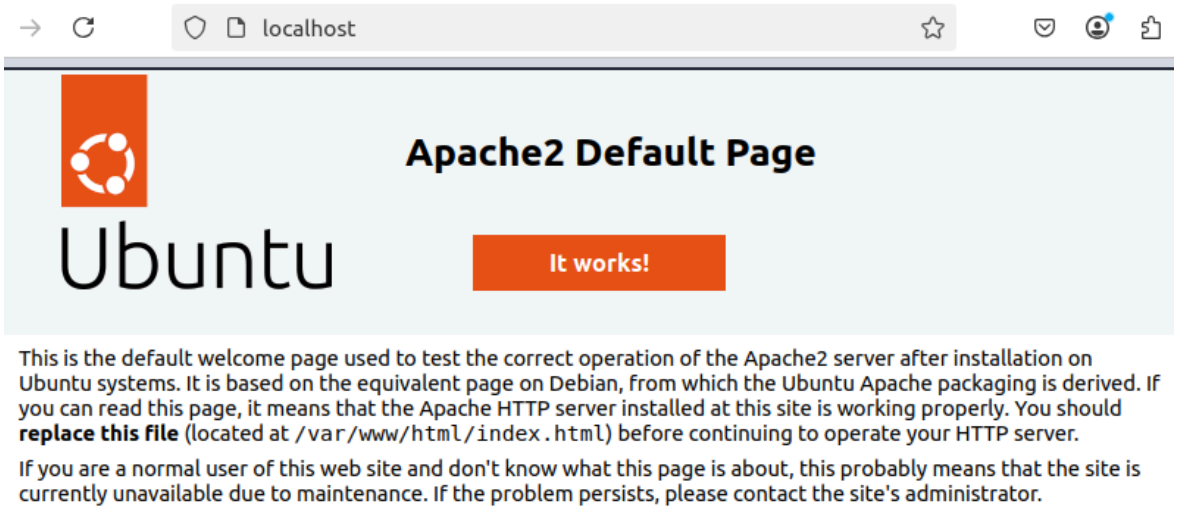
Step 1: Install phpMyAdmin.

```
system@server:~$ sudo apt-get install phpmyadmin
Reading package lists... Done
Building dependency tree... Done
Reading state information... Done
phpmyadmin is already the newest version (4:5.1.1+dfsg1-5ubuntu1).
0 upgraded, 0 newly installed, 0 to remove and 181 not upgraded.
```

Step 2: Verify if the apache2 web server is running.

```
system@server:~$ sudo systemctl status apache2
● apache2.service - The Apache HTTP Server
     Loaded: loaded (/lib/systemd/system/apache2.service; enabled; vendor prese>
     Active: active (running) since Tue 2025-03-04 08:57:25 IST; 8s ago
       Docs: https://httpd.apache.org/docs/2.4/
    Process: 9732 ExecStart=/usr/sbin/apachectl start (code=exited, status=0/SU>
   Main PID: 9737 (apache2)
      Tasks: 55 (limit: 4549)
     Memory: 5.3M
        CPU: 53ms
     CGroup: /system.slice/apache2.service
             ├─9737 /usr/sbin/apache2 -k start
             ├─9738 /usr/sbin/apache2 -k start
             └─9739 /usr/sbin/apache2 -k start

Mar 04 08:57:25 server.example.com systemd[1]: Starting The Apache HTTP Server.>
Mar 04 08:57:25 server.example.com systemd[1]: Started The Apache HTTP Server.
lines 1-16/16 (END)
```

Step3: On browser, open localhost.



Practical 9

NTP

(write the commands)

```
system@server:~$ date
Tuesday 04 March 2025 09:10:07 AM IST
system@server:~$ timedatectl
               Local time: Tue 2025-03-04 09:10:22 IST
           Universal time: Tue 2025-03-04 03:40:22 UTC
                 RTC time: Tue 2025-03-04 03:40:22
                Time zone: Asia/Kolkata (IST, +0530)
System clock synchronized: yes
              NTP service: active
          RTC in local TZ: no
```

```
system@server:~$ sudo timedatectl set-time "2026-05-17"
Failed to set time: Automatic time synchronization is enabled
```

Setting up NTP

Step 1: Install NTP packages.

```
system@server:~$ sudo apt-get install ntp
Reading package lists... Done
Building dependency tree... Done
Reading state information... Done
The following additional packages will be installed:
  libopts25 sntp
Suggested packages:
  ntp-doc
The following packages will be REMOVED:
  systemd-timesyncd
The following NEW packages will be installed:
  libopts25 ntp sntp
0 upgraded, 3 newly installed, 1 to remove and 181 not upgraded.
Need to get 847 kB of archives.
After this operation, 2,247 kB of additional disk space will be used.
Do you want to continue? [Y/n] y
```

Step 2: Configure firewall to allow outgoing UDP traffic for NTP

```
system@server:~$ sudo ufw allow out 123/udp
Rules updated
Rules updated (v6)
```

Step 3: Copy the servers from the NTP Pool Project.

Step 4: Add the servers and comment existing pools in the NTP configuration file.

```
sudo gedit /etc/ntp.conf
```

```
16 # Specify one or more NTP servers.
17 server 0.asia.pool.ntp.org
18 server 1.asia.pool.ntp.org
19 server 2.asia.pool.ntp.org
20 server 3.asia.pool.ntp.org
21
22 # Use servers from the NTP Pool Project. Approved by Ubuntu Technical Board
23 # on 2011-02-08 (LP: #104525). See http://www.pool.ntp.org/join.html for
24 # more information.
25 #pool 0.ubuntu.pool.ntp.org iburst
26 #pool 1.ubuntu.pool.ntp.org iburst
27 #pool 2.ubuntu.pool.ntp.org iburst
28 #pool 3.ubuntu.pool.ntp.org iburst
```

Step 5: Restart the NTP service to apply the configuration.

```
system@server:~$ sudo systemctl restart ntp
system@server:~$ sudo systemctl status ntp
● ntp.service - Network Time Service
     Loaded: loaded (/lib/systemd/system/ntp.service; enabled; vendor preset: e>
     Active: active (running) since Tue 2025-03-04 09:31:02 IST; 17s ago
       Docs: man:ntpd(8)
    Process: 12398 ExecStart=/usr/lib/ntp/ntp-systemd-wrapper (code=exited, sta>
   Main PID: 12404 (ntpd)
      Tasks: 2 (limit: 4549)
     Memory: 1.4M
        CPU: 36ms
     CGroup: /system.slice/ntp.service
             └─12404 /usr/sbin/ntpd -p /var/run/ntpd.pid -g -u 134:140
```

Step 6: Verifying NTP Synchronization.

```
system@server:~$ date
Tuesday 04 March 2025 09:31:57 AM IST
```

Practical 10

File Compression

Q] write a menu driven shell script to:
#create a file, accept filename from the user
#display the content of any file, accept filename from the user
#create the copy of a file, accept name of the file of which the copy is to be created and accept file name to be given to the copy of file
#rename a file, accept the filename of which you want to rename and accept the new filename
#delete a file, accept the filename from the user which file is to be deleted
#run this loop till the user wishes to continue

```
#!/bin/bash

while true
do
     echo "Menu:"
     echo "1.Create a file"
     echo "2.Display content of a file"
     echo "3.Create copy of a file"
     echo "4.Rename a file"
     echo "5.Delete a file"
     read -p "Enter choice:" choice
     case $choice in
     1) read -p "Enter filename:" filename; touch $filename;;
     2) read -p "Enter filename:" filename; cat $filename;;
     3) read -p "Enter current filename:" oldfile; read -p
"Enter new filename:" newfile; cp $oldfile $newfile;;
     4) read -p "Enter current filename:" oldfile; read -p
"Enter new filename:" newfile; mv $oldfile $newfile;;
     5) read -p "Enter filename:" filename; rm $filename;;
     *) echo "Invalid input";;
     esac
     read -p "Do you wish to continue(y/n):" con
     [ $con != "y" ] && break
done
```

Practical 11

Step1: Check and change your hostname

```
hostnamectl
hostnamectl set-hostname server.example.com
```

```
system@server:~$ hostnamectl
 Static hostname: server.example.com
       Icon name: computer-vm
         Chassis: vm
      Machine ID: f147e3cd98fe4988842db7ee783ad1b1
         Boot ID: 0519d15faffa4373b933e2f76fb200af
  Virtualization: vmware
Operating System: Ubuntu 24.04.1 LTS
          Kernel: Linux 6.11.0-19-generic
    Architecture: x86-64
 Hardware Vendor: VMware, Inc.
  Hardware Model: VMware Virtual Platform
Firmware Version: 6.00
   Firmware Date: Thu 2020-11-12
    Firmware Age: 4y 4month 4d
system@Ubuntu:~$ hostnamectl set-hostname server.example.com
```

Step 2: Check your network interface name

```
ifconfig -s
```

```
system@server:~$ ifconfig -s
Iface      MTU    RX-OK RX-ERR RX-DRP RX-OVR    TX-OK TX-ERR TX-DRP TX-OVR Flg
ens33     1500     1955      0      0 0          1359      0      0      0 BMRU
lo       65536      164      0      0 0           164      0      0      0 LRU
```

Step 3: Restart your network interface

```
sudo ifconfig ens33 down
sudo ifconfig ens33 up
```

```
system@server:~$ sudo ifconfig ens33 down
system@server:~$ sudo ifconfig ens33 up
```

Step 4: Install the bind9 packages

```
sudo apt-get update
sudo apt-get install bind9 bind9utils
```

```
system@server:~$ sudo apt-get install bind9 bind9utils
Reading package lists... Done
Building dependency tree... Done
Reading state information... Done
The following additional packages will be installed:
  bind9-utils
```

Step 5: Change directory to the DNS config files path.

```
cd /etc/bind
ls *.conf
```

```
system@server:~$ cd /etc/bind
system@server:/etc/bind$ ls *.conf
named.conf
```

Step 6: Edit the `named.conf.local` file as shown below

```
sudo gedit named.conf.local
```

Open named.conf.local /etc/bind

```
//
// Do any local configuration here
//

// Consider adding the 1918 zones here, if they are not
// organization
//include "/etc/bind/zones.rfc1918";
zone "example.com" IN {
type master;
file "/etc/bind/forward.example.com";
};

zone "43.168.192.in-addr.arpa" IN {
type master;
file "/etc/bind/reverse.example.com";
};
```

Step 7: Copy the `db.local` file as `forward.example.com`

```
sudo cp db.local forward.example.com
```

Step 8: Edit `forward.example.com` as shown below

```
sudo gedit /etc/bind/forward.example.com
```

Open forward.example.com /etc/bind

```
;
; BIND data file for local loopback interface
;
$TTL    604800
@       IN      SOA     localhost. root.localhost. (
                              2         ; Serial
                         604800         ; Refresh
                          86400         ; Retry
                        2419200         ; Expire
                         604800 )       ; Negative Cache
;
@       IN      NS      server.example.com.
@       IN      A       192.168.152.128
server  IN      A       192.168.152.128
host    IN      A       192.168.152.128
client  IN      A       192.168.152.129
www     IN      A       192.168.152.129
```

Step 9: Copy the `forward.example.com` file as `reverse.example.com`

```
sudo cp forward.example.com reverse.example.com
```

Step 10: Edit `reverse.example.com` as shown below

```
sudo gedit /etc/bind/reverse.example.com
```

Open

reverse.example.com
/etc/bind

```
;
; BIND data file for local loopback interface
;
$TTL    604800
@       IN      SOA     localhost. root.localhost. (
                              2         ; Serial
                         604800         ; Refresh
                          86400         ; Retry
                        2419200         ; Expire
                         604800 )       ; Negative Cache
;
@       IN      NS      server.example.com.
@       IN      A       192.168.152.128
server  IN      A       192.168.152.128
host    IN      A       192.168.152.128
client  IN      A       192.168.152.129
www     IN      A       192.168.152.129
128     IN      PTR     server.example.com.
129     IN      PTR     client.example.com.
```

Step 11: Validate the `named.conf` and `named.conf.local` config files

```
sudo named-checkconf -z /etc/bind/named.conf
sudo named-checkconf -z /etc/bind/named.conf.local
```

```
system@server:/etc/bind$ sudo named-checkconf -z /etc/bind/named.conf
zone example.com/IN: loaded serial 2
zone 43.168.192.in-addr.arpa/IN: loaded serial 2
zone localhost/IN: loaded serial 2
zone 127.in-addr.arpa/IN: loaded serial 1
zone 0.in-addr.arpa/IN: loaded serial 1
zone 255.in-addr.arpa/IN: loaded serial 1
system@server:/etc/bind$ sudo named-checkconf -z /etc/bind/named.conf.local
zone example.com/IN: loaded serial 2
zone 43.168.192.in-addr.arpa/IN: loaded serial 2
```

Step 12: Validate the `forward.example.com` and `reverse.example.com` zone files

```
sudo named-checkzone forward /etc/bind/forward.example.com
sudo named-checkzone reverse /etc/bind/reverse.example.com
```

```
system@server:/etc/bind$ sudo named-checkzone forward /etc/bind/forward.example.com
zone forward/IN: loaded serial 2
OK
system@server:/etc/bind$ sudo named-checkzone reverse /etc/bind/reverse.example.com
zone reverse/IN: loaded serial 2
OK
```

Step 13: Start the bind9 service

```
sudo systemctl start bind9
sudo systemctl status bind9
```

```
system@server:/etc/bind$ sudo systemctl start bind9
system@server:/etc/bind$ sudo systemctl status bind9
● named.service - BIND Domain Name Server
     Loaded: loaded (/usr/lib/systemd/system/named.service; enabled; preset: enabled)
     Active: active (running) since Tue 2025-03-18 09:08:28 IST; 41min ago
       Docs: man:named(8)
   Main PID: 5748 (named)
     Status: "running"
      Tasks: 14 (limit: 9382)
     Memory: 8.2M (peak: 9.0M)
```

Step 14: Change permissions of the /bind directory and its contents

```
sudo chmod -R 755 /etc/bind
```

Step 15: Restart the bind9 service

```
sudo systemctl restart bind9
sudo systemctl status bind9
```

```
system@server:/etc/bind$ sudo chmod -R 755 /etc/bind
system@server:/etc/bind$ sudo systemctl restart bind9
system@server:/etc/bind$ sudo systemctl status bind9
● named.service - BIND Domain Name Server
     Loaded: loaded (/usr/lib/systemd/system/named.service; enabled; preset: enabled)
     Active: active (running) since Tue 2025-03-18 09:51:26 IST; 3s ago
       Docs: man:named(8)
   Main PID: 6375 (named)
     Status: "running"
```

Step 16: Edit `/network/interfaces` as shown below

```
sudo gedit /etc/network/interfaces
```

Open

interfaces
/etc/network

```
1 auto lo
2 iface lo iname loopback
3 auto enp0s8
4 iface enp0s8 iname static
5 address 192.168.152.128
6 netmask 255.255.255.0
7 dns-search example.com
8 dns-nameserver 192.168.152.128
```

Step 17: Edit `resolv.conf` as shown below

```
sudo gedit /etc/resolv.conf
```

Open

resolv.conf
/etc

```
1 # This is /run/systemd/resolve/stub-resolv.conf mana
2 # Do not edit.
3 #
4 # This file might be symlinked as /etc/resolv.conf.
5 # /etc/resolv.conf and seeing this text, you have fc
6 #
7 # This is a dynamic resolv.conf file for connecting
8 # internal DNS stub resolver of systemd-resolved. Th
9 # configured search domains.
10 #
11 # Run "resolvectl status" to see details about the u
12 # currently in use.
13 #
14 # Third party programs should typically not access t
15 # through the symlink at /etc/resolv.conf. To manage
16 # different way, replace this symlink by a static fi
17 #
18 # See man:systemd-resolved.service(8) for details ab
19 # operation for /etc/resolv.conf.
20
21 nameserver 192.168.152.128
22 search example.com
```

Step 18: Make sure the firewall is inactive

`sudo ufw status`

Step 19: Perform ping and nslookup

`ping server`

`ping host`

```
system@server:/etc/bind$ sudo ifconfig ens33 down
system@server:/etc/bind$ sudo ifconfig ens33 up
system@server:/etc/bind$ sudo ufw status
WARN: / is world writable!
Status: inactive
system@server:/etc/bind$ ping server
PING server.example.com (172.17.100.74) 56(84) bytes of data.
64 bytes from 172.17.100.74: icmp_seq=1 ttl=64 time=0.070 ms
64 bytes from 172.17.100.74: icmp_seq=2 ttl=64 time=0.033 ms
64 bytes from 172.17.100.74: icmp_seq=3 ttl=64 time=0.032 ms
64 bytes from 172.17.100.74: icmp_seq=4 ttl=64 time=0.031 ms
64 bytes from 172.17.100.74: icmp_seq=5 ttl=64 time=0.040 ms
64 bytes from 172.17.100.74: icmp_seq=6 ttl=64 time=0.035 ms
^C
--- server.example.com ping statistics ---
6 packets transmitted, 6 received, 0% packet loss, time 5118ms
rtt min/avg/max/mdev = 0.031/0.040/0.070/0.013 ms
system@server:/etc/bind$ ping host
PING host.example.com (172.17.100.74) 56(84) bytes of data.
64 bytes from 172.17.100.74: icmp_seq=1 ttl=64 time=0.017 ms
64 bytes from 172.17.100.74: icmp_seq=2 ttl=64 time=0.046 ms
64 bytes from 172.17.100.74: icmp_seq=3 ttl=64 time=0.033 ms
64 bytes from 172.17.100.74: icmp_seq=4 ttl=64 time=0.047 ms
64 bytes from 172.17.100.74: icmp_seq=5 ttl=64 time=0.036 ms
^C
--- host.example.com ping statistics ---
```

nslookup server
nslookup host
nslookup www.google.com