BASH SCRIPTING LOOPS AND CONDITIONS

1. Create a bash script which will take alphabet as input and will check if the input is a consonent or vowel.

`[[ ... ]]` is used for conditional tests.

`${vowels[@]}` is used to expand the `vowels` array into a list of its elements.

`=~` is a pattern matching operator

2. Create a bash script which will check if the apache is installed; if installed it will remove the apache and will install nginx.

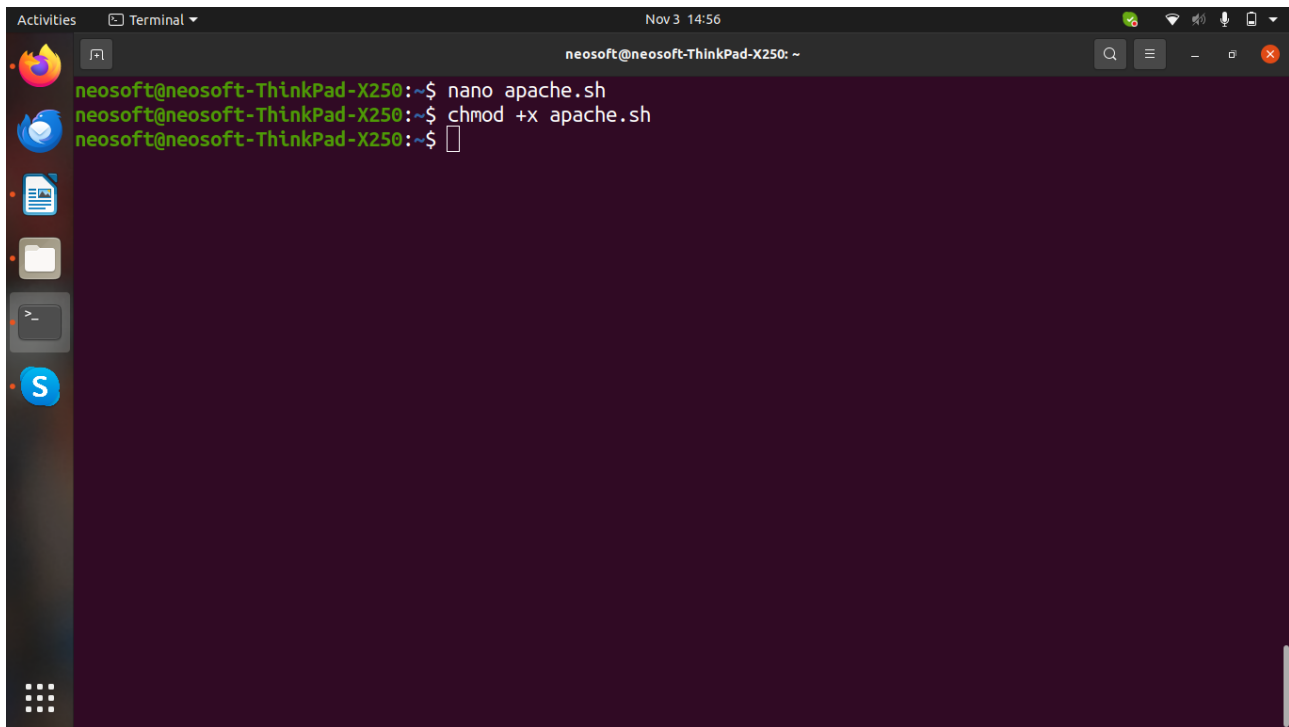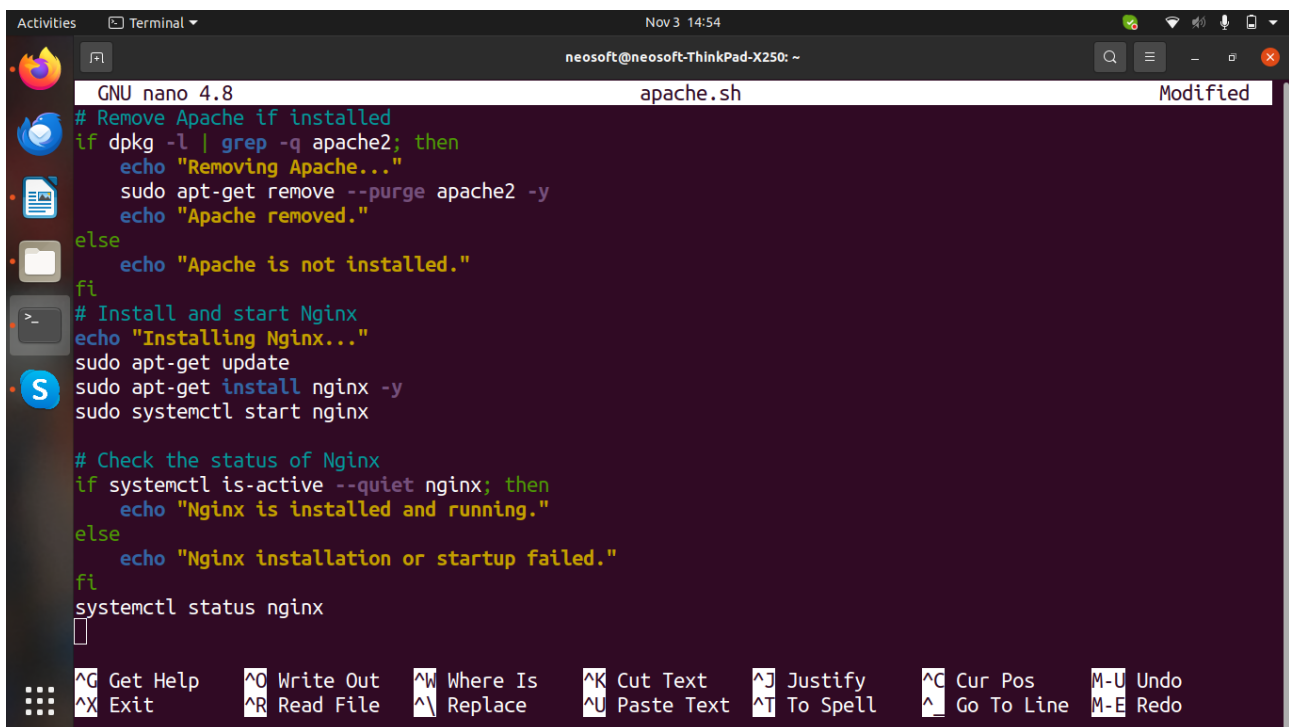```
neosoft@neosoft-ThinkPad-X250:~$ nano apache.sh
neosoft@neosoft-ThinkPad-X250:~$ chmod +x apache.sh
neosoft@neosoft-ThinkPad-X250:~$
```

```
  GNU nano 4.8                         apache.sh                              Modified
# Remove Apache if installed
if dpkg -l | grep -q apache2; then
    echo "Removing Apache..."
    sudo apt-get remove --purge apache2 -y
    echo "Apache removed."
else
    echo "Apache is not installed."
fi
# Install and start Nginx
echo "Installing Nginx..."
sudo apt-get update
sudo apt-get install nginx -y
sudo systemctl start nginx

# Check the status of Nginx
if systemctl is-active --quiet nginx; then
    echo "Nginx is installed and running."
else
    echo "Nginx installation or startup failed."
fi
systemctl status nginx
```

^G Get Help   ^O Write Out   ^W Where Is   ^K Cut Text   ^J Justify   ^C Cur Pos   M-U Undo
^X Exit   ^R Read File   ^\ Replace   ^U Paste Text ^T To Spell   ^_ Go To Line M-E Redo

Explaining the script:

- if dpkg -l | grep -q apache2; then

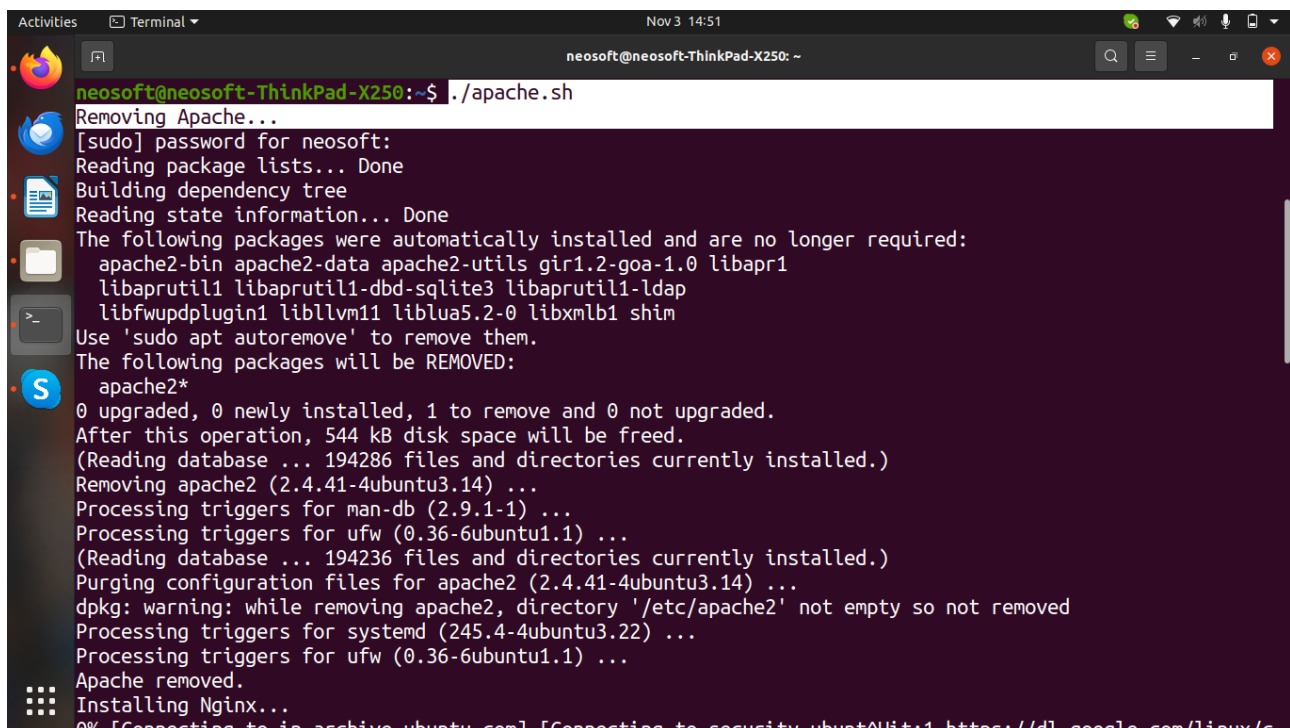 this command is used to list all the packages on the system and finding the 'apache2'

- sudo apt-get remove --purge apache2 -y:

  if apache package is found then with this command it is removed

- After removing the apache message is displayed that Apache is removed and the nginx is installed by the command:

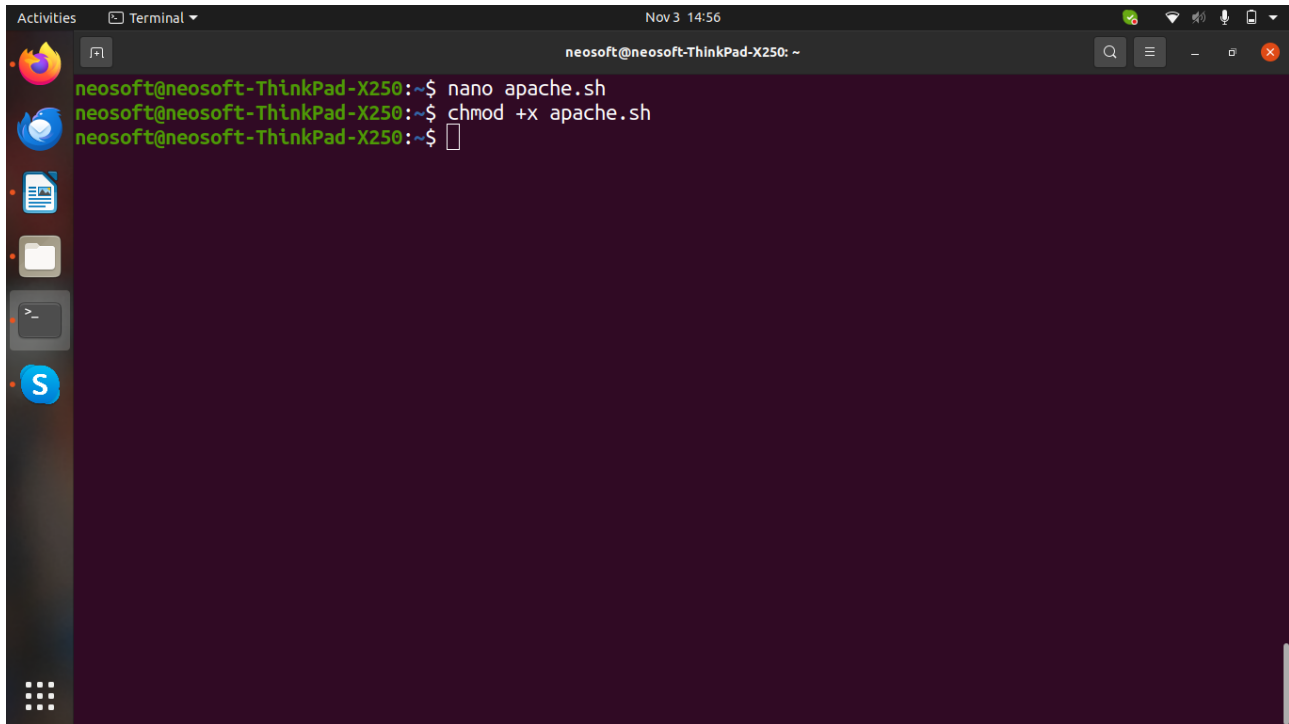 sudo apt-get install nginx -y

  sudo systemctl start nginx

neosoft@neosoft-ThinkPad-X250: ~

```
Apache removed.
Installing Nginx...
0% [Connecting to in.archive.ubuntu.com] [Connecting to security.ubunt^Hit:1 https://dl.google.com/linux/c
hrome/deb stable InRelease
Get:2 http://security.ubuntu.com/ubuntu focal-security InRelease [114 kB]
Get:3 http://security.ubuntu.com/ubuntu focal-security/main i386 Packages [672 kB]
Get:4 http://security.ubuntu.com/ubuntu focal-security/main amd64 Packages [2,564 kB]
Hit:5 http://in.archive.ubuntu.com/ubuntu focal InRelease
Get:6 http://in.archive.ubuntu.com/ubuntu focal-updates InRelease [114 kB]
Get:7 http://in.archive.ubuntu.com/ubuntu focal-backports InRelease [108 kB]
Get:8 http://in.archive.ubuntu.com/ubuntu focal-updates/main i386 Packages [906 kB]
Get:9 http://in.archive.ubuntu.com/ubuntu focal-updates/main amd64 Packages [2,973 kB]
Get:10 http://security.ubuntu.com/ubuntu focal-security/main amd64 DEP-11 Metadata [59.9 kB]
Get:11 http://security.ubuntu.com/ubuntu focal-security/main amd64 c-n-f Metadata [13.2 kB]
Get:12 http://security.ubuntu.com/ubuntu focal-security/restricted i386 Packages [34.7 kB]
Get:13 http://security.ubuntu.com/ubuntu focal-security/restricted amd64 Packages [2,367 kB]
Get:14 http://security.ubuntu.com/ubuntu focal-security/restricted Translation-en [329 kB]
Get:15 http://security.ubuntu.com/ubuntu focal-security/restricted amd64 c-n-f Metadata [552 B]
Get:16 http://security.ubuntu.com/ubuntu focal-security/universe i386 Packages [619 kB]
Get:17 http://security.ubuntu.com/ubuntu focal-security/universe amd64 Packages [899 kB]
Get:18 http://in.archive.ubuntu.com/ubuntu focal-updates/main amd64 DEP-11 Metadata [275 kB]
Get:19 http://in.archive.ubuntu.com/ubuntu focal-updates/main amd64 c-n-f Metadata [17.2 kB]
Get:20 http://in.archive.ubuntu.com/ubuntu focal-updates/restricted i386 Packages [36.2 kB]
Get:21 http://in.archive.ubuntu.com/ubuntu focal-updates/restricted amd64 Packages [2,480 kB]
Get:22 http://security.ubuntu.com/ubuntu focal-security/universe Translation-en [188 kB]
Get:23 http://security.ubuntu.com/ubuntu focal-security/universe amd64 DEP-11 Metadata [96.9 kB]
Get:24 http://security.ubuntu.com/ubuntu focal-security/universe amd64 c-n-f Metadata [19.2 kB]
```

neosoft@neosoft-ThinkPad-X250: ~

```
nginx is already the newest version (1.18.0-0ubuntu1.4).
The following packages were automatically installed and are no longer required:
  apache2-bin apache2-data apache2-utils gir1.2-goa-1.0 libapr1
  libaprutil1 libaprutil1-dbd-sqlite3 libaprutil1-ldap
  libfwupdplugin1 libllvm11 liblua5.2-0 libxmlb1 shim
Use 'sudo apt autoremove' to remove them.
0 upgraded, 0 newly installed, 0 to remove and 1 not upgraded.
Nginx is installed and running.
neosoft@neosoft-ThinkPad-X250:~$ systemctl status nginx
● nginx.service - A high performance web server and a reverse proxy server
     Loaded: loaded (/lib/systemd/system/nginx.service; disabled; vendor preset: enabled)
     Active: active (running) since Fri 2023-11-03 14:10:01 IST; 42min ago
       Docs: man:nginx(8)
   Main PID: 14610 (nginx)
      Tasks: 5 (limit: 9277)
     Memory: 5.9M
     CGroup: /system.slice/nginx.service
             ├─14610 nginx: master process /usr/sbin/nginx -g daemon on; master_process on;
             ├─14611 nginx: worker process
             ├─14612 nginx: worker process
             ├─14613 nginx: worker process
             └─14614 nginx: worker process

Nov 03 14:10:01 neosoft-ThinkPad-X250 systemd[1]: Starting A high performance web server and a reverse pr>
Nov 03 14:10:01 neosoft-ThinkPad-X250 systemd[1]: Started A high performance web server and a reverse pro>
lines 1-16/16 (END)
```
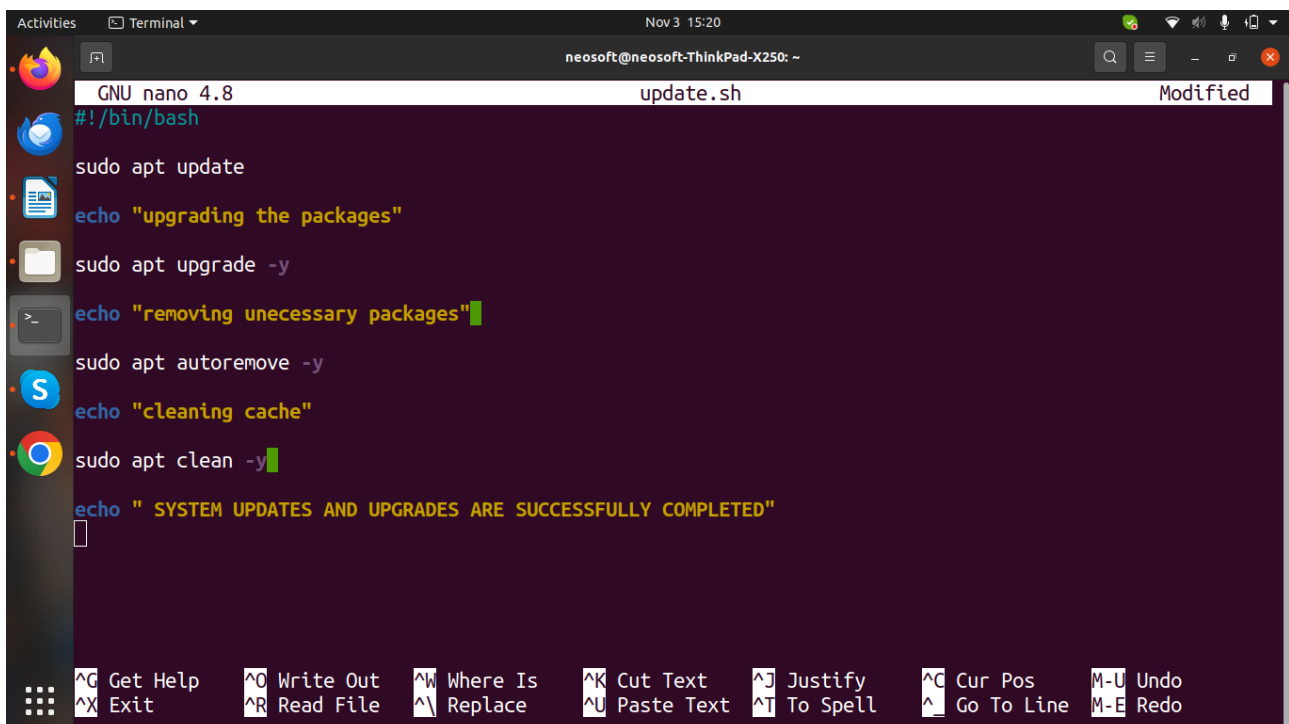
3. Create a bsah script which will check the linux distribution and will execute update commands for the specific distribution.

neosoft@neosoft-ThinkPad-X250: ~

```
neosoft@neosoft-ThinkPad-X250:~$ ./update.sh
[sudo] password for neosoft:
Hit:1 https://dl.google.com/linux/chrome/deb stable InRelease
Get:2 http://security.ubuntu.com/ubuntu focal-security InRelease [114 kB]
Hit:3 http://in.archive.ubuntu.com/ubuntu focal InRelease
Get:4 http://in.archive.ubuntu.com/ubuntu focal-updates InRelease [114 kB]
Get:5 http://security.ubuntu.com/ubuntu focal-security/main amd64 DEP-11 Metadata [59.9 kB]
Get:6 http://in.archive.ubuntu.com/ubuntu focal-backports InRelease [108 kB]
Get:7 http://security.ubuntu.com/ubuntu focal-security/universe amd64 DEP-11 Metadata [97.0 kB]
Get:8 http://security.ubuntu.com/ubuntu focal-security/multiverse amd64 DEP-11 Metadata [940 B]
Get:9 http://in.archive.ubuntu.com/ubuntu focal-updates/main amd64 DEP-11 Metadata [274 kB]
Get:10 http://in.archive.ubuntu.com/ubuntu focal-updates/universe amd64 DEP-11 Metadata [414 kB]
Get:11 http://in.archive.ubuntu.com/ubuntu focal-updates/multiverse amd64 DEP-11 Metadata [944 B]
Get:12 http://in.archive.ubuntu.com/ubuntu focal-backports/main amd64 DEP-11 Metadata [7,976 B]
Get:13 http://in.archive.ubuntu.com/ubuntu focal-backports/universe amd64 DEP-11 Metadata [30.5 kB]
Fetched 1,222 kB in 11s (114 kB/s)
Reading package lists... Done
Building dependency tree
Reading state information... Done
1 package can be upgraded. Run 'apt list --upgradable' to see it.
upgrading the packages
Reading package lists... Done
Building dependency tree
Reading state information... Done
Calculating upgrade... Done
The following packages were automatically installed and are no longer required:
  apache2-bin apache2-data apache2-utils gir1.2-goa-1.0 libapr1 libaprutil1 libaprutil1-dbd-sqlite3
  libaprutil1-ldap libfwupdplugin1 libllvm11 liblua5.2-0 libxmlb1 shim
Use 'sudo apt autoremove' to remove them.
#
# Canonical released microcode updates for both Intel (CVE-2022-40982) and AMD
# (CVE-2023-20593). 'Unattended upgrades' provide security updates by default
```

neosoft@neosoft-ThinkPad-X250: ~

```
Unpacking libsndfile1:amd64 (1.0.28-7ubuntu0.2) over (1.0.28-7ubuntu0.1) ...
Setting up libsndfile1:amd64 (1.0.28-7ubuntu0.2) ...
Processing triggers for libc-bin (2.31-0ubuntu9.12) ...
removing unecessary packages
Reading package lists... Done
Building dependency tree
Reading state information... Done
The following packages will be REMOVED:
  apache2-bin apache2-data apache2-utils gir1.2-goa-1.0 libapr1 libaprutil1 libaprutil1-dbd-sqlite3
  libaprutil1-ldap libfwupdplugin1 libllvm11 liblua5.2-0 libxmlb1 shim
0 upgraded, 0 newly installed, 13 to remove and 0 not upgraded.
After this operation, 91.3 MB disk space will be freed.
(Reading database ... 194070 files and directories currently installed.)
Removing apache2-bin (2.4.41-4ubuntu3.14) ...
Removing apache2-data (2.4.41-4ubuntu3.14) ...
Removing apache2-utils (2.4.41-4ubuntu3.14) ...
Removing gir1.2-goa-1.0:amd64 (3.36.1-0ubuntu1) ...
Removing libaprutil1-dbd-sqlite3:amd64 (1.6.1-4ubuntu2.2) ...
Removing libaprutil1-ldap:amd64 (1.6.1-4ubuntu2.2) ...
Removing libaprutil1:amd64 (1.6.1-4ubuntu2.2) ...
Removing libapr1:amd64 (1.6.5-1ubuntu1) ...
Removing libfwupdplugin1:amd64 (1.5.11-0ubuntu1~20.04.2) ...
Removing libllvm11:amd64 (1:11.0.0-2~ubuntu20.04.1) ...
Removing liblua5.2-0:amd64 (5.2.4-1.1build3) ...
Removing libxmlb1:amd64 (0.1.15-2ubuntu1~20.04.1) ...
Removing shim (15.7-0ubuntu1) ...
Processing triggers for man-db (2.9.1-1) ...
Processing triggers for libc-bin (2.31-0ubuntu9.12) ...
cleaning cache
 SYSTEM UPDATES AND UPGRADES ARE SUCCESSFULLY COMPLETED
neosoft@neosoft-ThinkPad-X250:~$
```

4. Create a bash script which will compress all the files in a specific directory which have .log as extension. Also add a cronjob which will execute at a certain time.

neosoft@neosoft-ThinkPad-X250: ~

```
  GNU nano 4.8                          compress.sh
#!/bin/bash

# Directory containing log files
log_directory=/home/neosoft/logs/

# Check if the directory exists
if [ -d "$log_directory" ]; then
    # Change to the log directory
    cd "$log_directory"

    # Compress all .log files
    for log_file in *.log; do
        if [ -f "$log_file" ]; then
            sudo gzip "$log_file"
            echo "Compressed $log_file"
        fi
    done
else
    echo "Directory not found: $log_directory"
fi
```
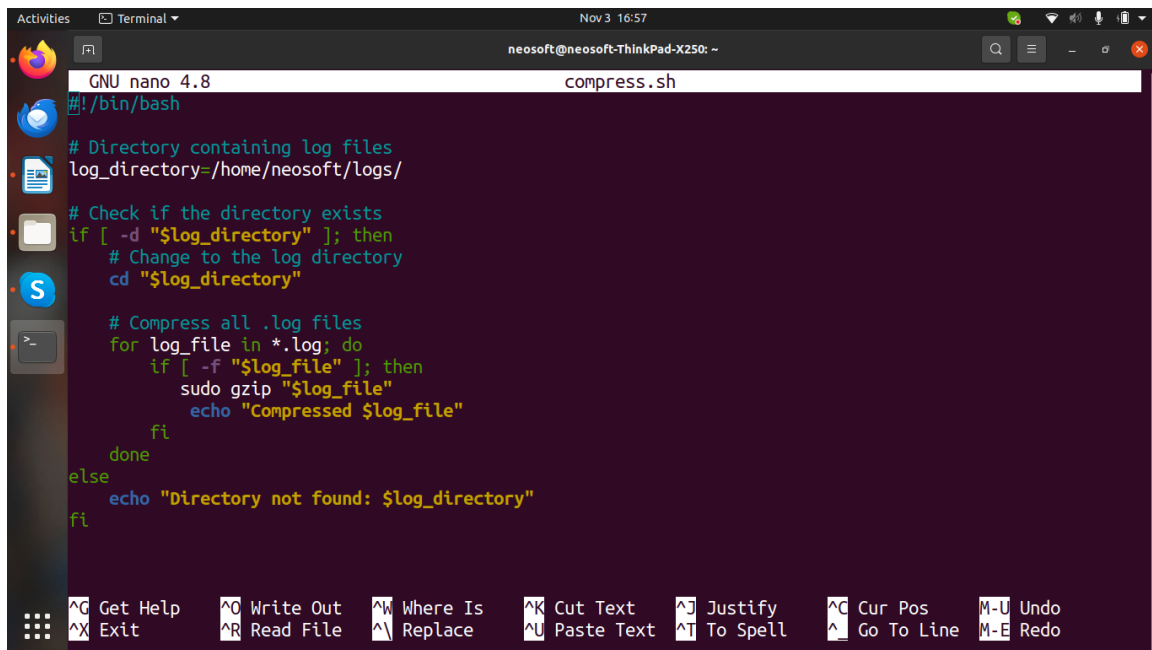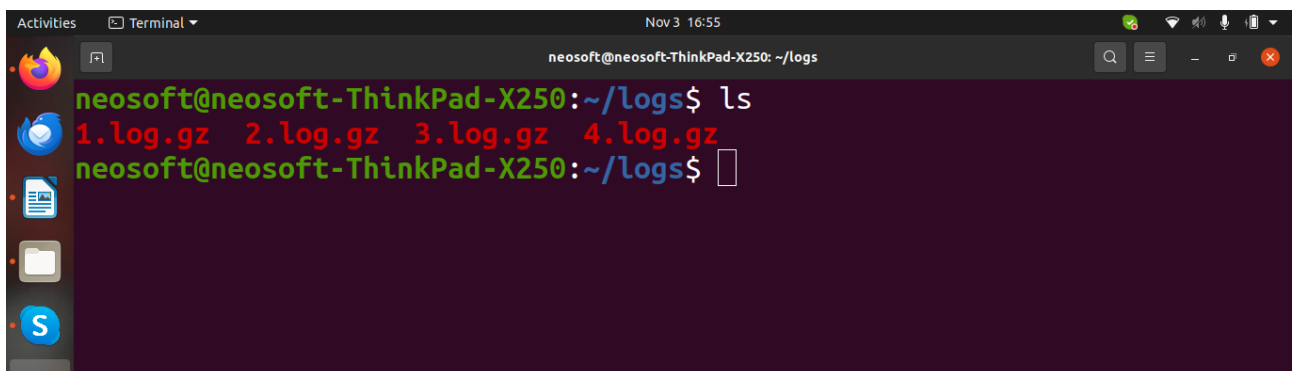
```
^G Get Help    ^O Write Out    ^W Where Is    ^K Cut Text     ^J Justify     ^C Cur Pos      M-U Undo
^X Exit        ^R Read File    ^\ Replace     ^U Paste Text   ^T To Spell    ^  Go To Line   M-E Redo
```

neosoft@neosoft-ThinkPad-X250: ~

```
  GNU nano 4.8                    /tmp/crontab.M9XdiP/crontab
# Edit this file to introduce tasks to be run by cron.
#
# Each task to run has to be defined through a single line
# indicating with different fields when the task will be run
# and what command to run for the task
#
# To define the time you can provide concrete values for
# minute (m), hour (h), day of month (dom), month (mon),
# and day of week (dow) or use '*' in these fields (for 'any').
#
# Notice that tasks will be started based on the cron's system
# daemon's notion of time and timezones.
#
# Output of the crontab jobs (including errors) is sent through
# email to the user the crontab file belongs to (unless redirected).
#
# For example, you can run a backup of all your user accounts
# at 5 a.m every week with:
# 0 5 * * 1 tar -zcf /var/backups/home.tgz /home/
#
# For more information see the manual pages of crontab(5) and cron(8)
#
# m h  dom mon dow   command
57 16 * * * /home/neosoft/compress.sh
```

```
^G Get Help    ^O Write Out    ^W Where Is    ^K Cut Text     ^J Justify     ^C Cur Pos      M-U Undo    M-A Mark Text
^X Exit        ^R Read File    ^\ Replace     ^U Paste Text   ^T To Spell    ^  Go To Line   M-E Redo    M-6 Copy Text
```

Mentioning the time when the script should be executed.

- Specifying the path where the log files are stored
- Compressing the files ending with .log using gzip

The files are compressed using the script and executing at specific time with the help of crontab.