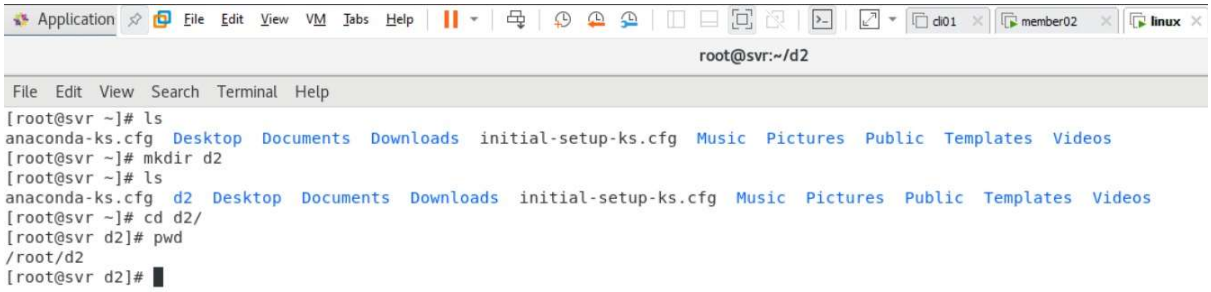
Exploring Linux Commands: File System Operations and Permissions Management

Objective:

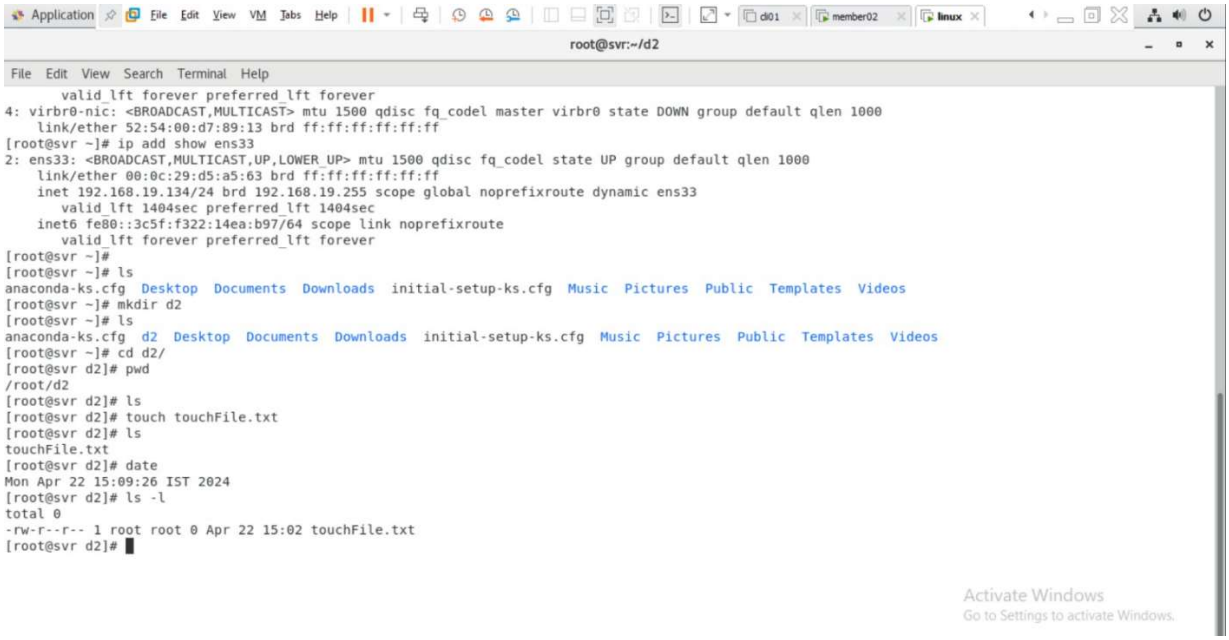
The objective of the commands used in the scenario is to illustrate various file system operations and permissions management in a Linux environment. These commands cover tasks such as navigating directories, creating, modifying, and deleting files, as well as managing file permissions, ownership, and group ownership.

1. **Navigation Commands:**
    - **ls**: Lists files and directories in the current directory.
    - **cd**: Changes the current directory.
    - **pwd**: Prints the current working directory.
    - **mkdir**: Creates a new directory.
    - **touch**: Creates a new file or updates the timestamp of an existing file.
2. **File System Operations:**
    - **mv**: Moves or renames files or directories.
    - **cp**: Copies files or directories.
    - **rm**: Removes (deletes) files or directories.
    - **cat**: Displays the contents of a file or concatenates multiple files.
    - **nano**, **vi**, **vim**, **gedit**: Text editors used to create or edit files.
3. **File Permissions:**
    - **chmod**: Changes file permissions.
    - **chown**: Changes file ownership.
    - **chgrp**: Changes file group ownership.
    - **ls -l**: Lists files and directories in long format, including permissions, ownership, and group.
4. **File Input/Output:**
    - **cat > file**: Creates a new file or overwrites existing content with input from the command line.
    - **history > file**: Redirects command history to a file.

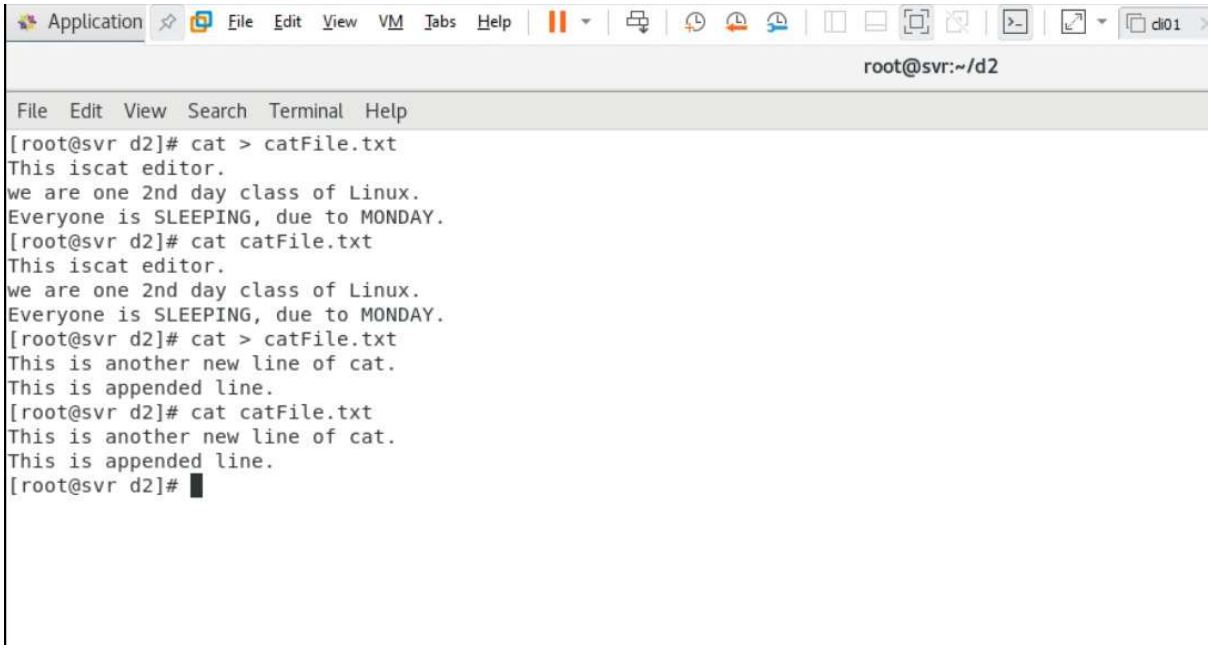
5. **Directory Operations:**
    - **rm -rf**: Removes directories and their contents recursively.
    - **ls -R**: Lists files and directories recursively.
    - **ls -a**: Lists all files and directories, including hidden ones.
6. **Miscellaneous:**
    - **date**: Displays the current date and time.
    - **cal**: Displays a calendar.
    - **wc**: Prints newline, word, and byte counts for each file.
    - **man**: Displays the manual pages for a command.



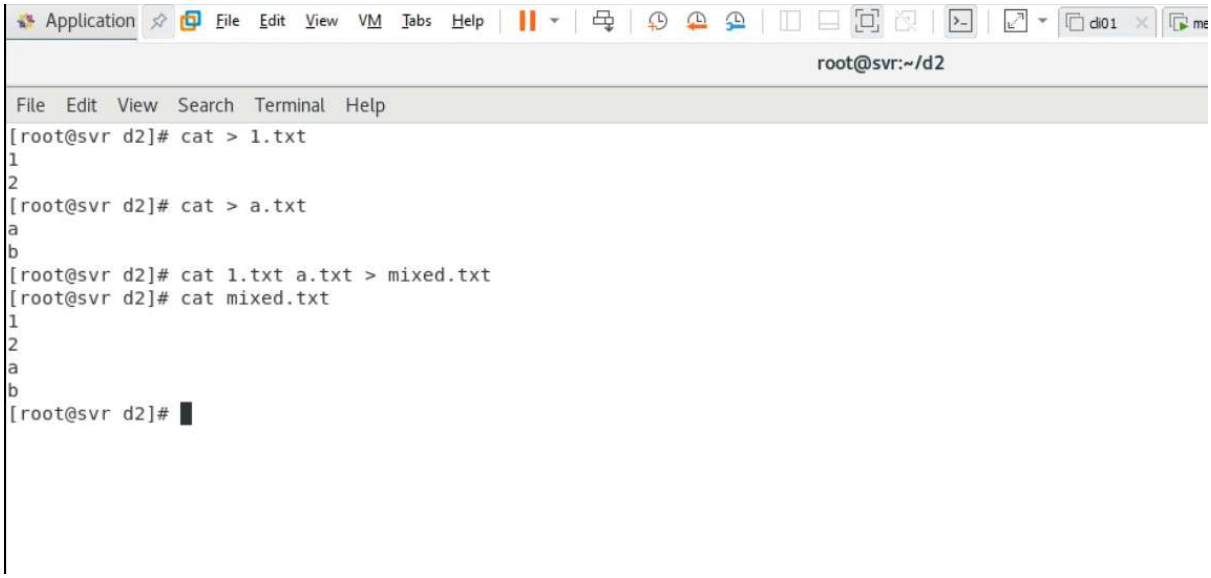
ls	This command lists the files and directories in the current directory.
mkdir d2	This command creates a new directory named d2.
ls	Lists the files and directories again to show the newly created directory d2.
cd d2/	Changes the current directory to d2.
pwd	Prints the current working directory, which should now be d2.



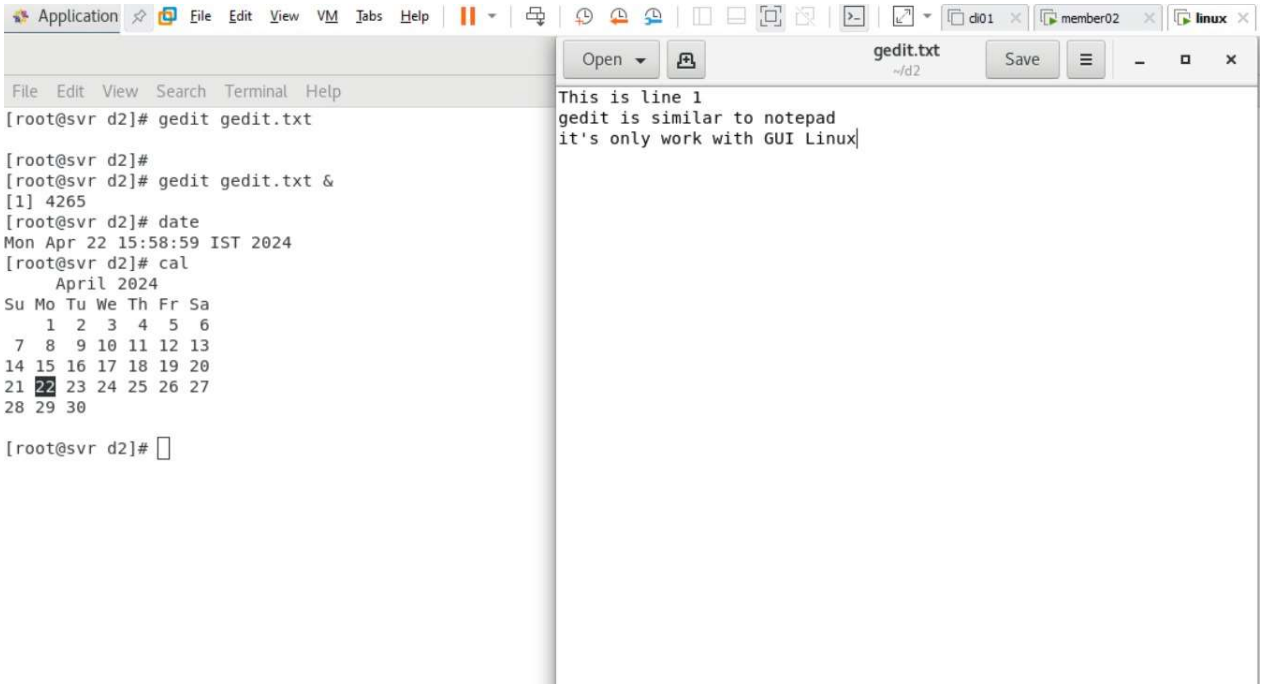
ls	Lists the files and directories in the d2 directory.
touch touchFile.txt	Creates a new empty file named touchFile.txt.
ls	Lists the files and directories again to show the newly created file.
date	Prints the current date and time.
ls -l	Lists the files and directories in long format, providing detailed information about each file or directory.



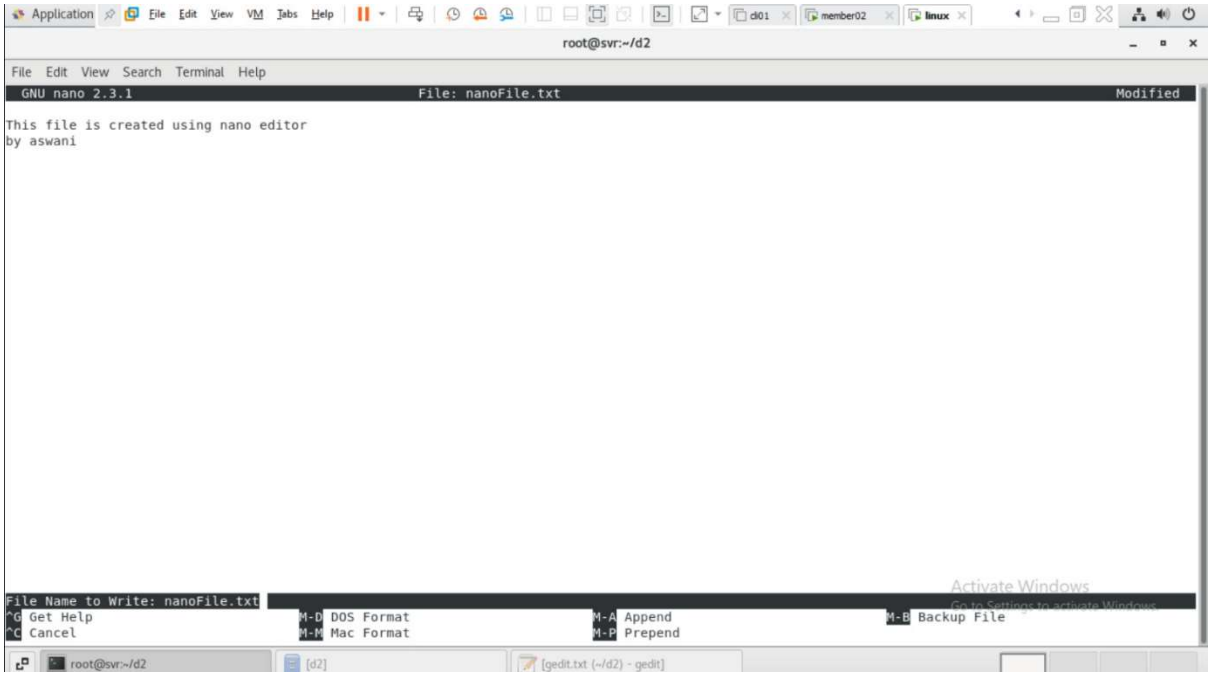
cat > catFile.txt	Begins creating a new file named catFile.txt and allows the user to input text into it. The user can type text into the terminal, and it will be saved to the file until they exit with Ctrl + D.
cat catFile.txt	Displays the contents of the file catFile.txt.
cat /etc/shells	Displays the contents of the /etc/shells file, which typically lists the valid login shells on the system.
cat > catFile.txt	this begins creating a new catFile.txt and allows the user to input text.
cat catFile.txt	Displays the contents of catFile.txt.
cat >> catFile.txt	Appends more text to the existing catFile.txt file.
cat catFile.txt	Displays the contents of catFile.txt after appending text.
ls	Lists the files and directories in the current directory.



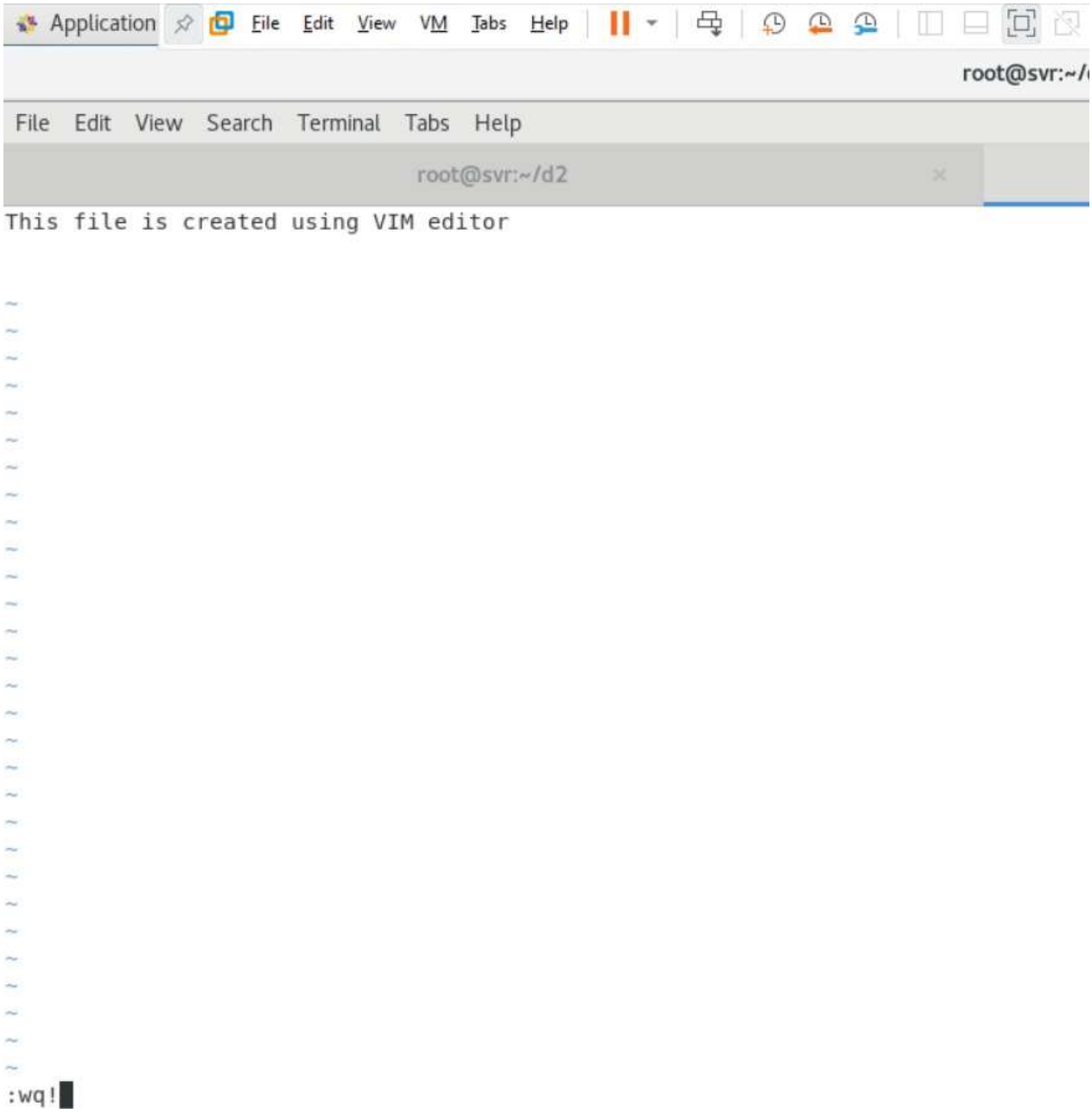
cat 1.txt	Attempts to display the contents of a file named 1.txt, assuming it exists in the current directory.
cat > 1.txt	Starts creating a new file named 1.txt and allows the user to input text.
cat > a.txt	Creates a new file named a.txt and allows the user to input text.
cat 1.txt a.txt > mixed.txt	Concatenates the contents of 1.txt and a.txt into a new file named mixed.txt.
cat mixed.txt	Displays the contents of the file mixed.txt.
cat > mixed.txt	Starts creating a new file named mixed.txt and allows the user to input text.
cat mixed.txt	Displays the contents of the file mixed.txt, which should now contain the newly inputted text.



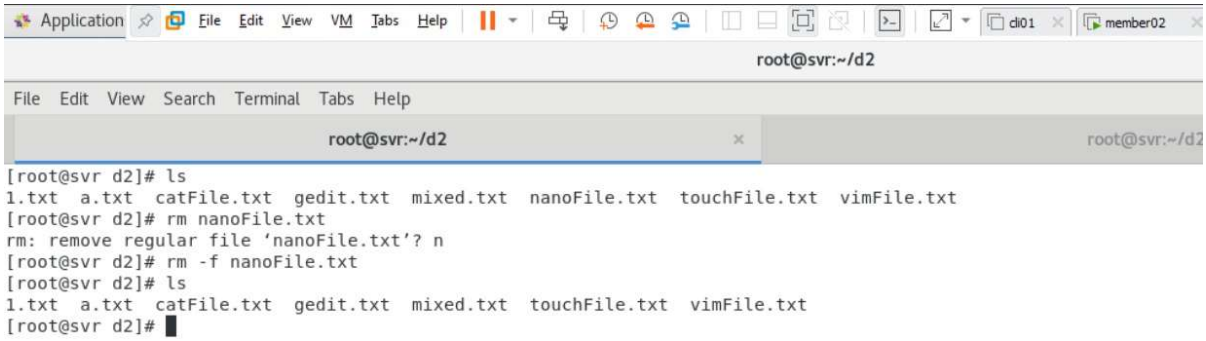
gedit gedit.txt	Opens the text editor Gedit and creates/open a file named gedit.txt for editing. The ampersand (&) at the end allows the command prompt to return immediately for further commands.
date	Prints the current date and time.
cal	Displays a calendar for the current month.



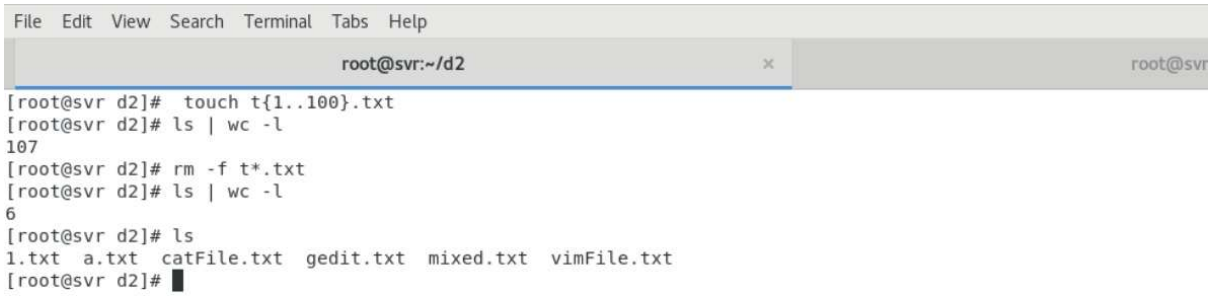
nano nanoFile.txt	Opens the text editor Nano and creates/opens a file named nanoFile.txt for editing.
cat nanoFile.txt	Displays the contents of the file nanoFile.txt.



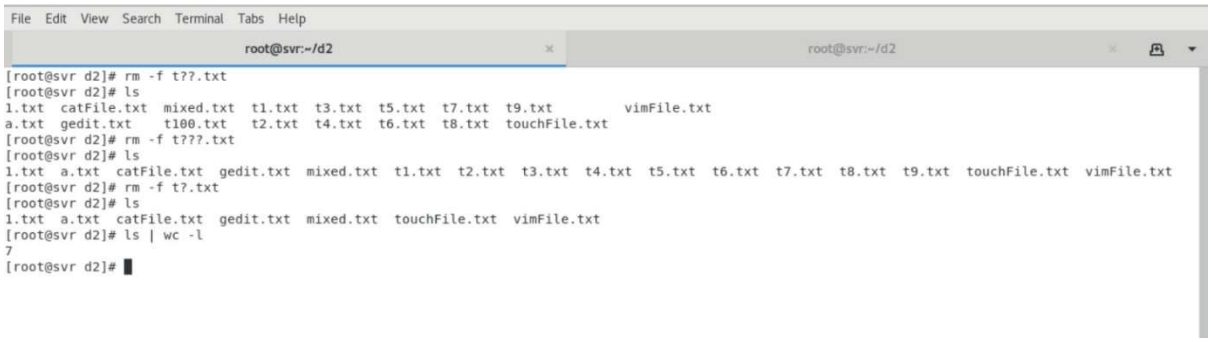
vi /etc/passwd	Opens the file /etc/passwd in the Vi text editor for viewing/editing.
vim vimFile.txt	Opens the file vimFile.txt in the Vim text editor for viewing/editing.
cat vimFile.txt	Displays the contents of the file vimFile.txt.



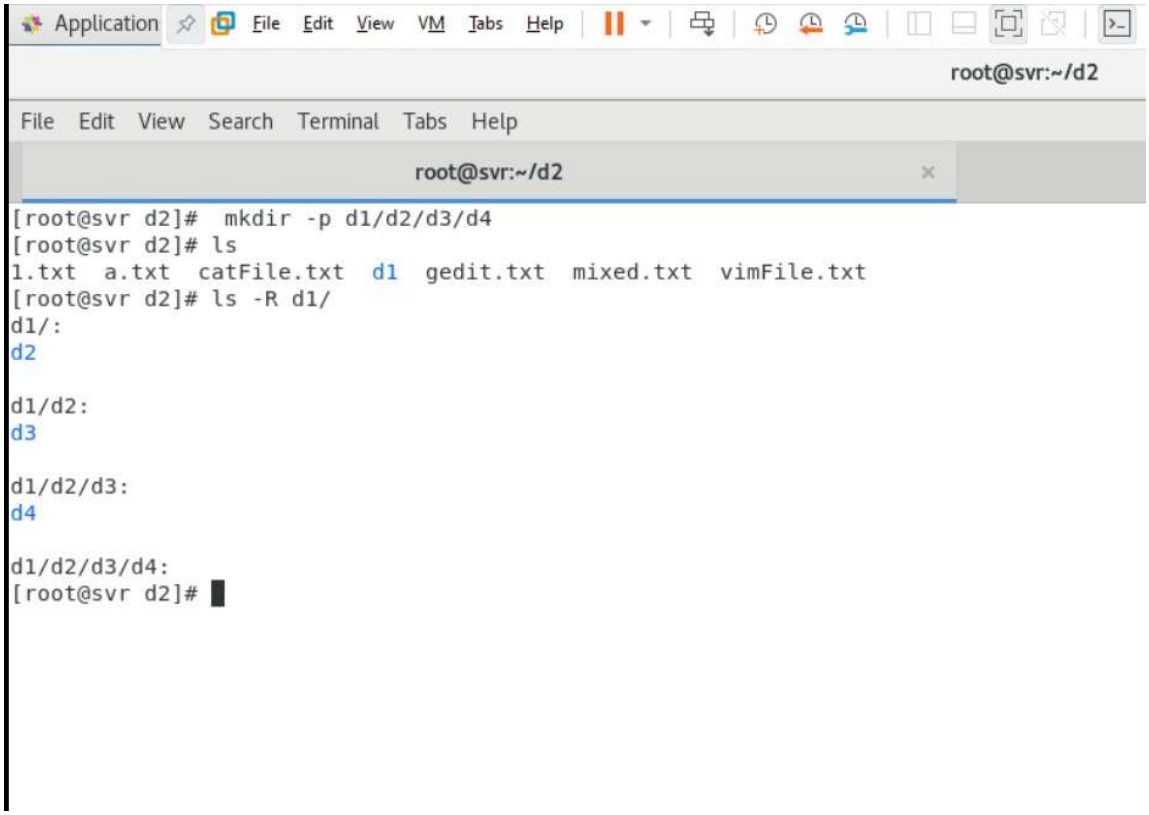
ls	Lists the files and directories in the current directory.
rm nanoFile.txt	Removes (deletes) the file nanoFile.txt.
rm -f nanoFile.txt	Removes (deletes) the file nanoFile.txt forcefully without prompting for confirmation.



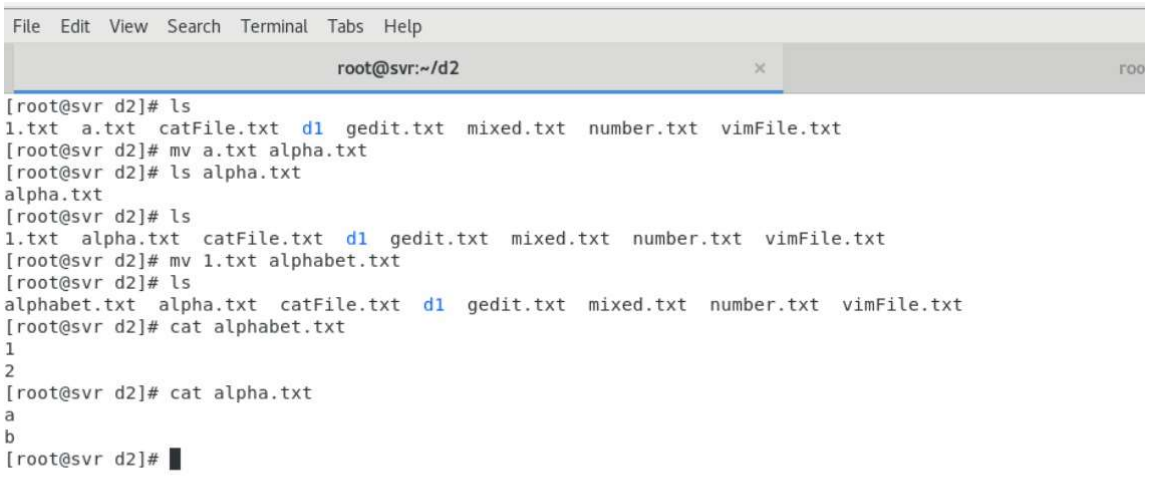
touch t{1..100}.txt	Creates 100 empty files named t1.txt, t2.txt, ..., t100.txt using brace expansion.
ls	Lists the files and directories in the current directory, showing the newly created files.
ls | wc -l	Counts the number of files and directories in the current directory.
ls	Lists the files and directories in the current directory.



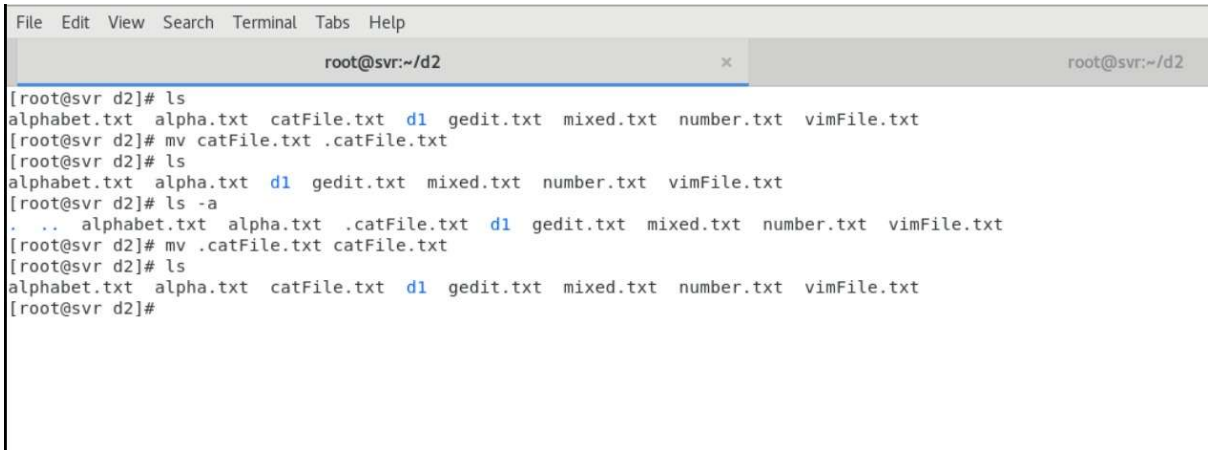
rm -f t?.txt	Removes (deletes) files with a single character wildcard in their name followed by .txt, for example, t1.txt, t2.txt, but not t10.txt.
ls	Lists the files and directories in the current directory after removal.
ls | wc -l	Counts the number of files and directories in the current directory.
ls	Lists the files and directories in the current directory.
rm -f t??.txt	Removes (deletes) files with a double character wildcard in their name followed by .txt, for example, t10.txt, t20.txt, but not t100.txt.
ls | wc -l	Counts the number of files and directories in the current directory.
ls	Lists the files and directories in the current directory.
rm -f t???.txt	Removes (deletes) files with a triple character wildcard in their name followed by .txt, for example, t100.txt, but not t10.txt, t20.txt.
ls	Lists the files and directories in the current directory after removal.



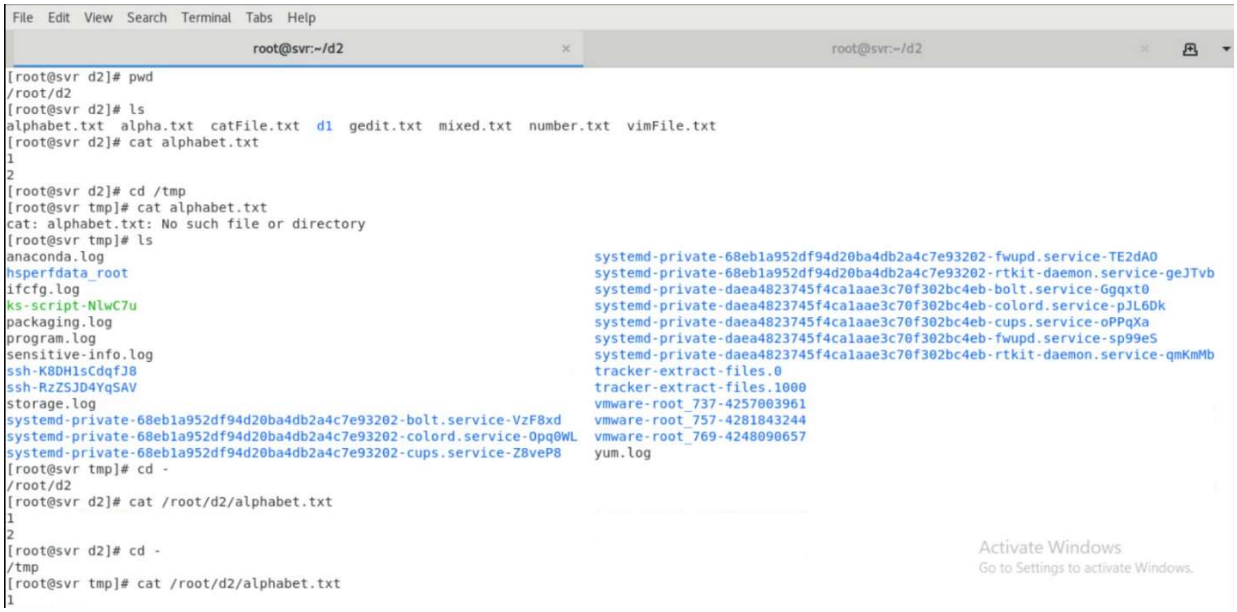
mkdir -p dir1/dir2/dir3/dir4/dir5/dir6	Recursively creates directories as necessary to create the specified directory structure.
ls	Lists the files and directories in the current directory.
ls -R dir1/	Lists the files and directories in the dir1 directory recursively, showing its subdirectories and their contents.

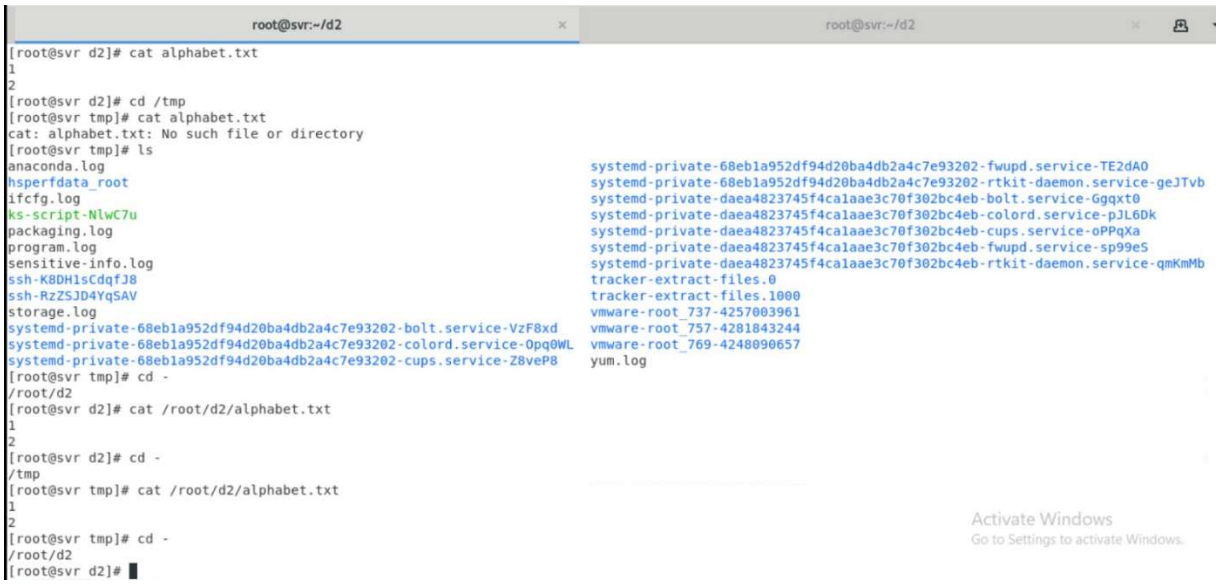


ls	Lists the files and directories in the current directory.
mv a.txt alpha.txt	Renames the file a.txt to alpha.txt.
ls	Lists the files and directories in the current directory after renaming.
cat alpha.txt	Displays the contents of the file alpha.txt.
ls	Lists the files and directories in the current directory.
mv 1.txt alphabet.txt	Renames the file 1.txt to alphabet.txt.
ls	Lists the files and directories in the current directory after renaming.
cat alphabet.txt	Displays the contents of the file alphabet.txt.



ls	Lists the files and directories in the current directory.
mv catFile.txt .catFile.txt	Moves (renames) the file catFile.txt to .catFile.txt. The . at the beginning makes the file hidden.
ls	Lists the files and directories in the current directory after the renaming.
ls -a	Lists all files and directories in the current directory, including hidden ones.
mv .catFile.txt catFile.txt	Renames the hidden file .catFile.txt back to catFile.txt.
ls	Lists the files and directories in the current directory after the renaming.





pwd	Prints the current working directory.
ls	Lists the files and directories in the current directory.
cat alphabet.txt	Displays the contents of the file alphabet.txt.
cd /tmp	Changes the current directory to /tmp.
cat alphabet.txt	Attempts to display the contents of the file alphabet.txt in the /tmp directory.
ls	Lists the files and directories in the current directory (/tmp).
cd -	Changes the current directory to the previous directory.

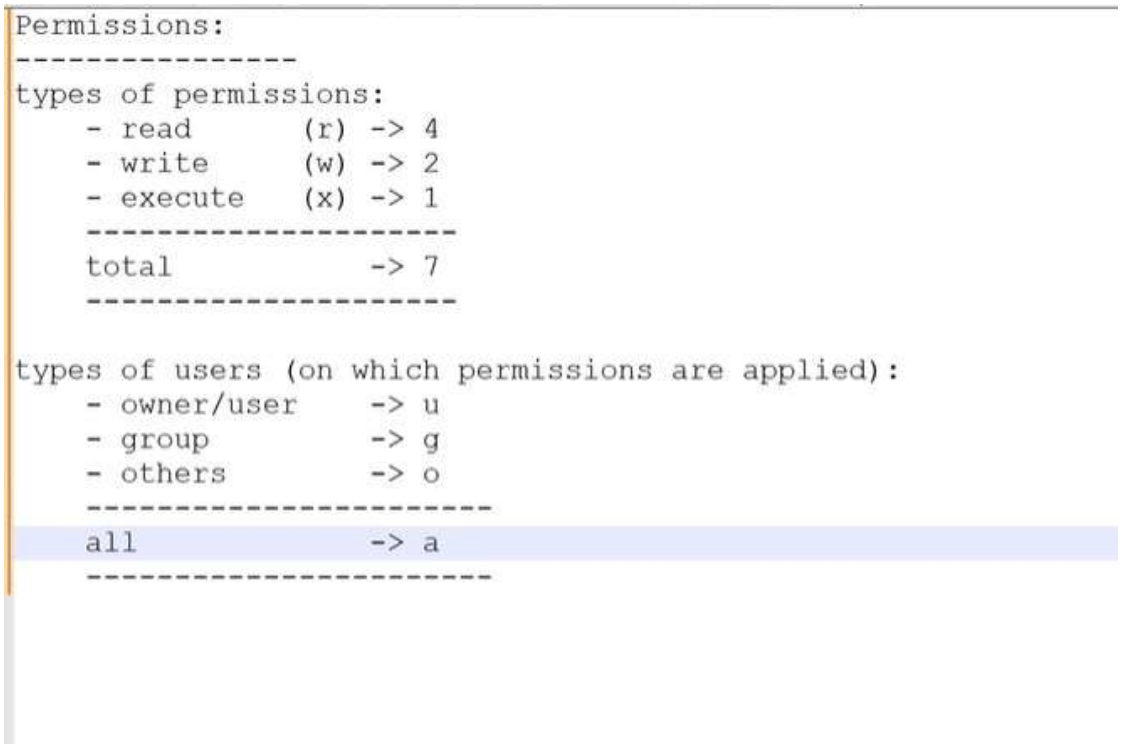
cat /root/d2/alphabet.txt	Attempts to display the contents of the file alphabet.txt located in the /root/d2/ directory.
cd -	Changes the current directory to the previous directory.
history	Displays a list of previously executed commands.

```
104  pwd
105  ls
106  cat alphabet.txt
107  cd /tmp
108  cat alphabet.txt
109  ls
110  cd -
111  cat /root/d2/alphabet.txt
112  cd -
113  cat /root/d2/alphabet.txt
114  cd -
115  history
```

```
[root@svr d2]# touch /day2/d2-commands.txt
[root@svr d2]# ls
alphabet.txt  alpha.txt  catFile.txt  d1  gedit.txt  mixed.txt  number.txt  vimFile.txt
[root@svr d2]# mv /day2/d2-commands.txt  .
[root@svr d2]# ls
alphabet.txt  alpha.txt  catFile.txt  d1  d2-commands.txt  gedit.txt  mixed.txt  number.txt  vimFile.txt
[root@svr d2]#
```

pwd	Prints the current working directory.
mkdir /day2	Attempts to create a directory named day2 in the root directory /.
ls	Lists the files and directories in the current directory.

ls /	Lists the files and directories in the root directory /.
touch /day2/d2-commands.txt	Creates an empty file named d2-commands.txt in the directory /day2.
ls /day2/	Lists the files and directories in the directory /day2.
rm /day2/d2-commands.txt	Removes (deletes) the file d2-commands.txt from the directory /day2.
ls /day2/	Lists the files and directories in the directory /day2.
touch /day2/d2-commands.txt	Creates an empty file named d2-commands.txt in the directory /day2.
ls	Lists the files and directories in the current directory.
mv /day2/d2-commands.txt .	Moves the file d2-commands.txt from the directory /day2 to the current directory (. represents the current directory).
ls	Lists the files and directories in the current directory.



- **Types of Permissions**:
  - **Read ®**: Allows viewing the contents of the file or directory, represented by the number **4**.
  - **Write (w)**: Grants the ability to modify or delete the file or directory, denoted by the number **2**.

- **Execute (x)**: Permits running the file as a program or accessing the directory, indicated by the number **1**.
- **Total**: The sum of these permissions is **7**, which represents full access (read, write, execute).

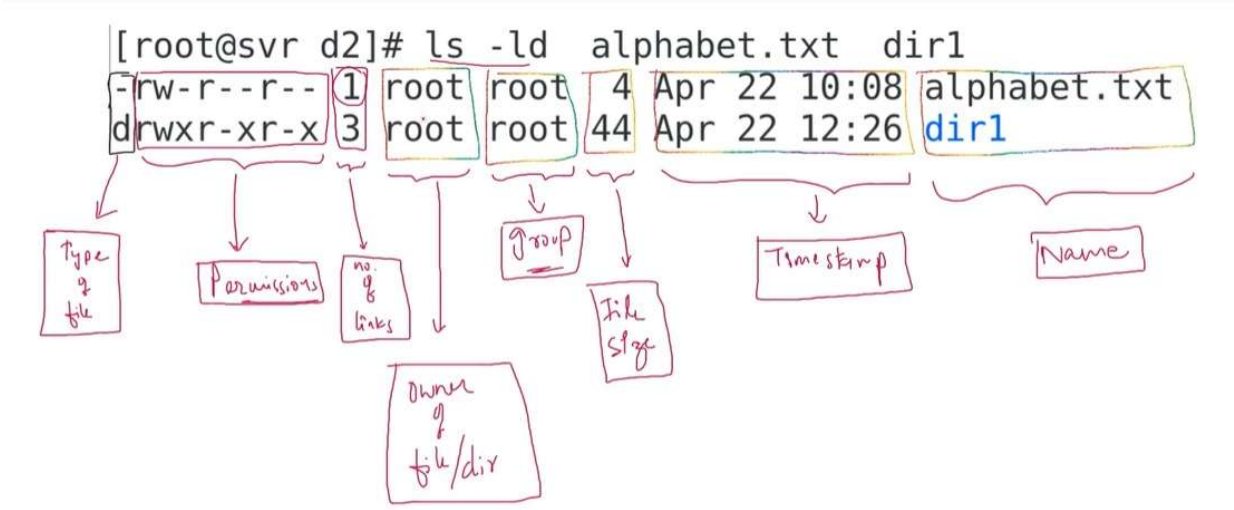
- **Permission Categories**:
  - **Owner/User (u)**: The file or directory creator, who has certain default permissions.
  - **Group (g)**: Users who are part of a group that has specific permissions for the file or directory.
  - **Others (o)**: All other users who have access to the file system.
  - **All (a)**: A shorthand for setting permissions for all categories (owner, group, others).

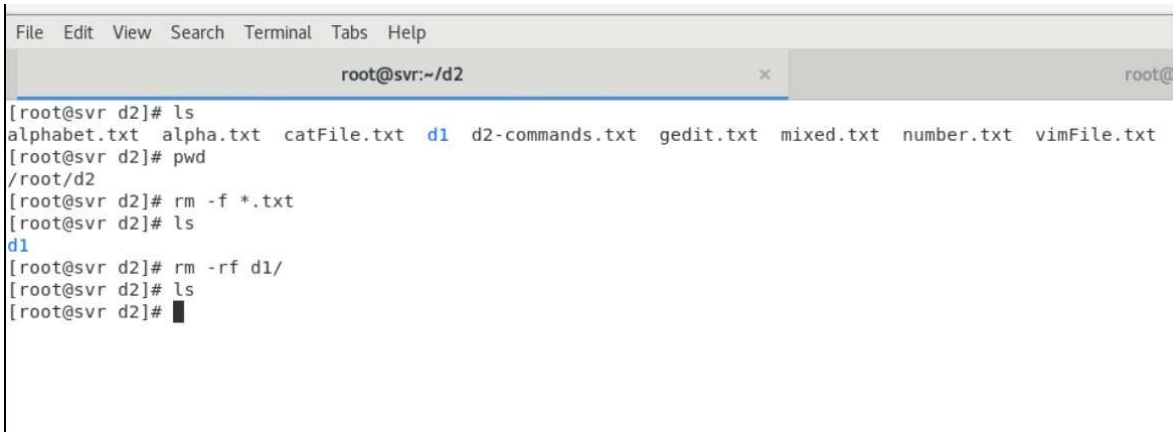
```
types of files in linux:
    1. -    -> file (txt, pdf, mp3, cfg, etc...)
    2. d    -> directory
    3. l    -> link (symobolic link files - hard link & soft link)
    4. c    -> char device file
    5. s    -> local socket file
    6. p    -> named pipe file
    7. b    -> block device file

to list all various types
    # ls -l /dev
```

- **File Types**:
  - **Regular Files (-)**: Standard files like text, PDF, MP3, etc.
  - **Directory (d)**: Folders that contain other files or directories.
  - **Symbolic Link (l)**: Shortcuts or references to other files or directories.
  - **Local Socket (s)**: Endpoints for inter-process communication.
  - **Named Pipe (p)**: Mechanisms for inter-process communication using a pipe.
  - **Character Device (c)**: Files that represent devices and support character-by-character access.
  - **Block Device (b)**: Files that represent devices and support block-oriented access.

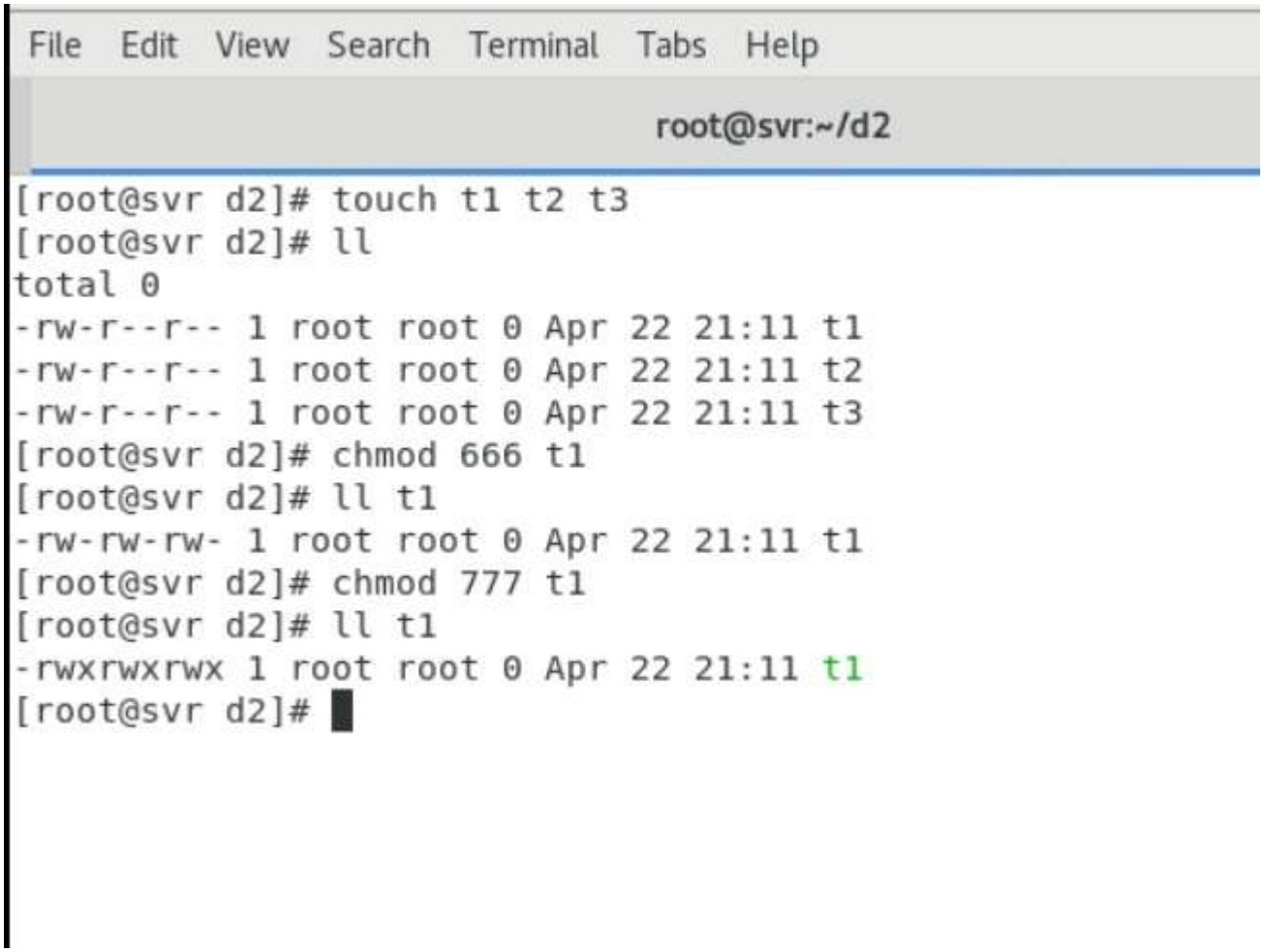
```
[root@svr d2]# ls -l /dev
total 0
crw-rw----  1 root video    10, 175 Apr 22 14:23 agpgart
crw-------  1 root root     10, 235 Apr 22 14:23 autofs
drwxr-xr-x  2 root root         160 Apr 22 14:23 block
drwxr-xr-x  2 root root          80 Apr 22 14:23 bsg
crw-------  1 root root     10, 234 Apr 22 14:23 btrfs-control
drwxr-xr-x  3 root root          60 Apr 22 14:23 bus
lrwxrwxrwx  1 root root           3 Apr 22 14:23 cdrom -> sr0
drwxr-xr-x  2 root root          80 Apr 22 14:23 centos
drwxr-xr-x  2 root root        3080 Apr 22 14:23 char
crw-------  1 root root      5,   1 Apr 22 14:23 console
lrwxrwxrwx  1 root root          11 Apr 22 14:23 core -> /proc/kcore
drwxr-xr-x  4 root root          80 Apr 22 14:23 cpu
crw-------  1 root root     10,  61 Apr 22 14:23 cpu_dma_latency
crw-------  1 root root     10,  62 Apr 22 14:23 crash
drwxr-xr-x  6 root root         120 Apr 22 14:23 disk
brw-rw----  1 root disk    253,   0 Apr 22 14:23 dm-0
brw-rw----  1 root disk    253,   1 Apr 22 14:23 dm-1
crw-rw----+ 1 root audio    14,   9 Apr 22 14:23 dmmidi
drwxr-xr-x  2 root root          80 Apr 22 14:23 dri
crw-rw----  1 root video    29,   0 Apr 22 14:23 fb0
lrwxrwxrwx  1 root root          13 Apr 22 14:23 fd -> /proc/self/fd
crw-rw-rw-  1 root root      1,   7 Apr 22 14:23 full
crw-rw-rw-  1 root root     10, 229 Apr 22 14:23 fuse
crw-------  1 root root    246,   0 Apr 22 14:23 hidraw0
crw-------  1 root root     10, 228 Apr 22 14:23 hpet
drwxr-xr-x  3 root root           0 Apr 22 14:23 hugepages
crw-------  1 root root     10, 183 Apr 22 14:23 hwrng
lrwxrwxrwx  1 root root          25 Apr 22 14:23 initctl -> /run/systemd/initctl/fifo
```

- The command # ls -l /dev is used to list all devices in the /dev directory with detailed information.

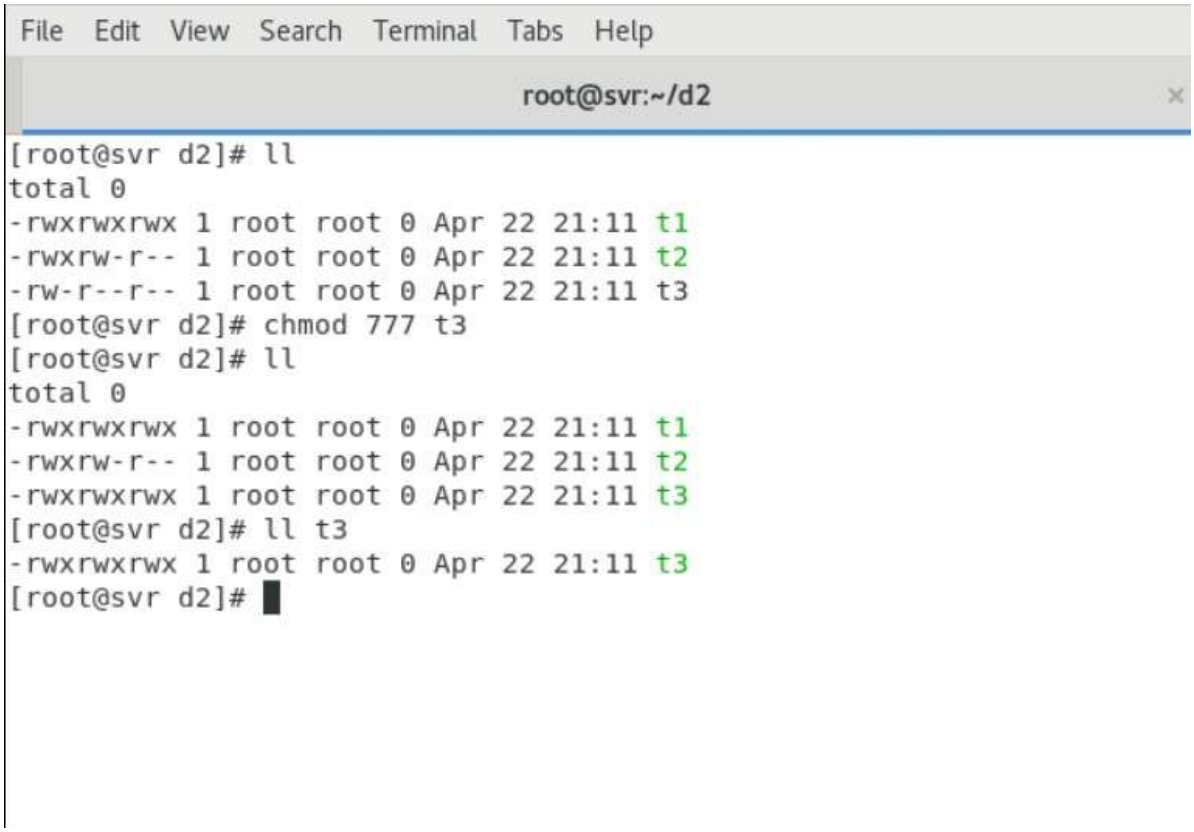




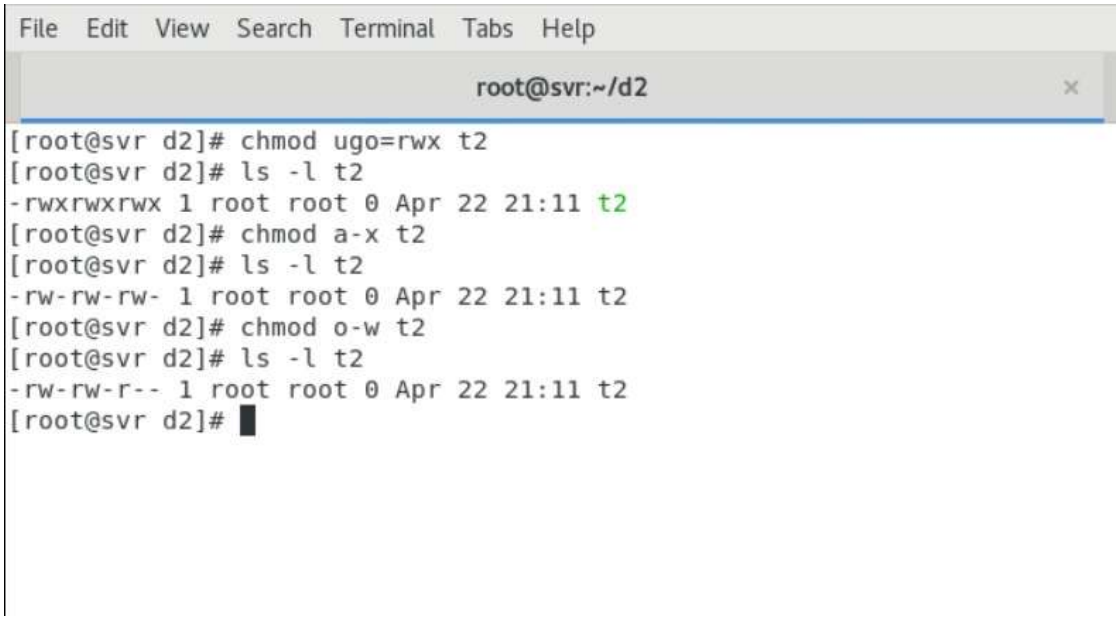
ls	Lists the files and directories in the current directory.
pwd	Prints the current working directory.
rm -f *.txt	Removes (deletes) all files with a .txt extension in the current directory. The -f option forces the removal without asking for confirmation.
ls	Lists the files and directories in the current directory after the removal of .txt files.
rm -rf dir1/ newdir/	Removes (deletes) the directories dir1 and newdir along with their contents. The -r option recursively removes directories and their contents, and the -f option forces the removal without asking for confirmation.
ls	Lists the files and directories in the current directory after the removal of dir1 and newdir.



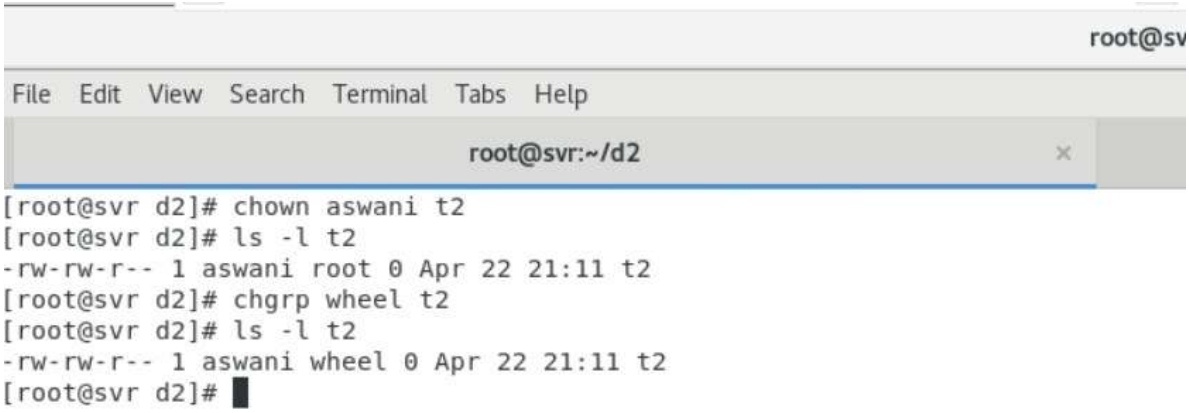
touch t1 t2 t3	Creates three empty files named t1, t2, and t3.
ll	It seems like a typo; ll is not a standard Unix command. It could be a custom alias or shorthand for another command like ls -l which lists files and directories in long format. Assuming it behaves like ls -l, it lists detailed information about the files t1, t2, and t3.
chmod 666 t1	Changes the permissions of the file t1 to allow read and write access for the owner, group, and others.
ll t1	Lists detailed information about the file t1, including its permissions after the chmod command in step 166.
chmod 777 t1	Changes the permissions of the file t1 to allow read, write, and execute access for the owner, group, and others.
ll t1	Lists detailed information about the file t1, including its permissions after the chmod command in step 168.
chmod 764 t1	Changes the permissions of the file t1 to allow read, write, and execute access for the owner, read and write access for the group, and read-only access for others.
ll t1	Lists detailed information about the file t1, including its permissions after the chmod command in step 170.

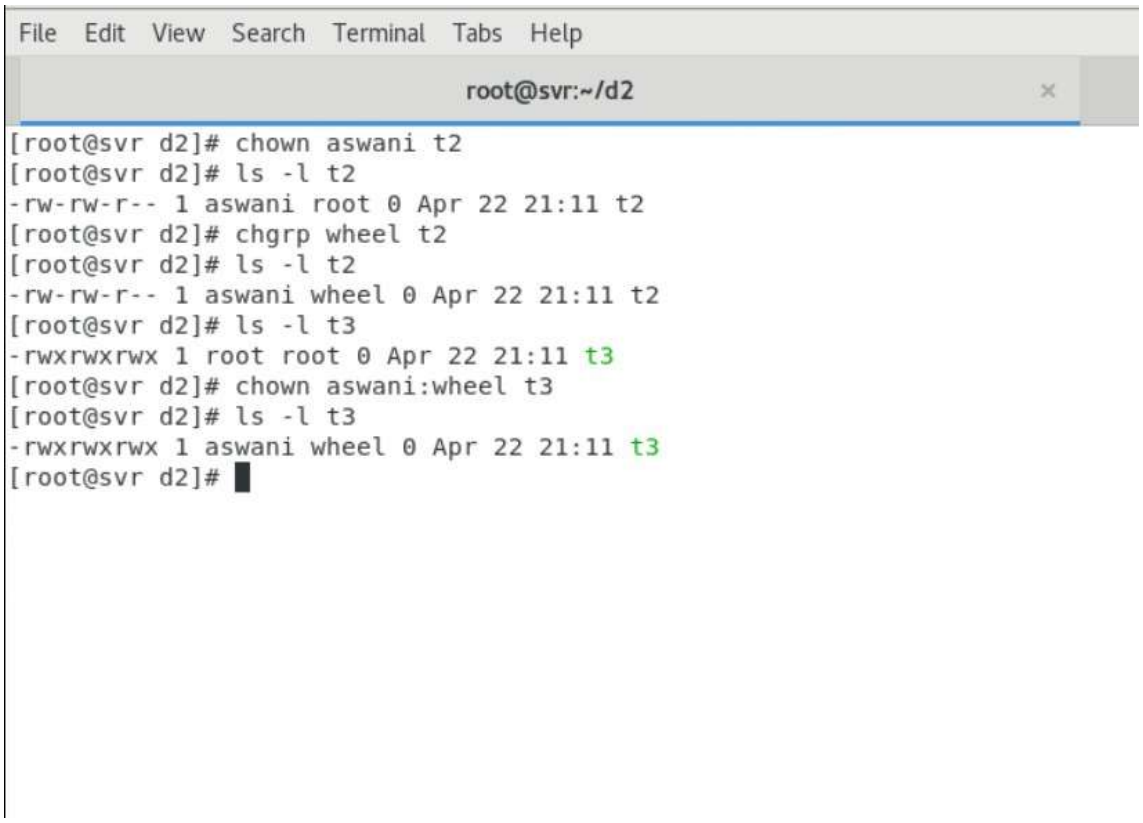


ll	This command seems like a custom alias or shorthand for another command, possibly ls -l, which lists files and directories in long format, including detailed information like permissions, ownership, size, and modification date.
chmod 777 t3	Changes the permissions of the file t3 to allow read, write, and execute access for the owner, group, and others.
ll	Lists the files and directories in the current directory in long format, including the file t3 with its updated permissions after the chmod command in step 2.
ll t3	Lists detailed information about the file t3, including its permissions.



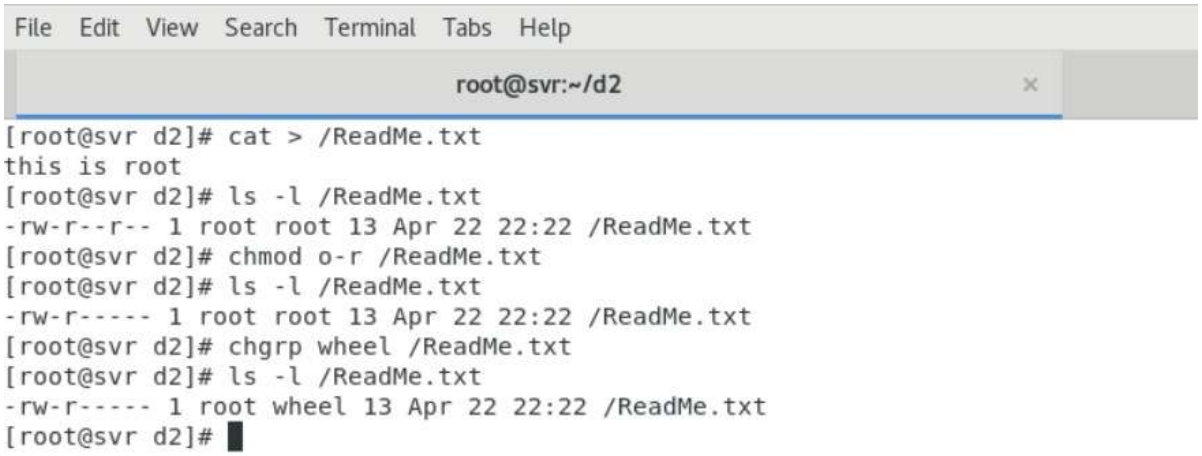
chmod ugo=rwx t2	Changes the permissions of the file t2 to allow read, write, and execute access for the owner, group, and others. The ugo=rwx syntax specifies permissions for user (owner), group, and others.
ls -l t2	Lists detailed information about the file t2, including its permissions.
chmod ugo=rwx t2	Changes the permissions of the file t2 to allow read, write, and execute access for the owner, group, and others. This command repeats the permission change from command 173.
ls -l t2	Lists detailed information about the file t2, including its permissions after the chmod command in step 176.
chmod a-x t2	Removes execute permissions for all users (owner, group, and others) from the file t2.
ls -l t2	Lists detailed information about the file t2, including its permissions after the chmod command in step 178.
chmod o-w t2	Removes write permissions for others (users who are not the owner or in the file's group) from the file t2.
ls -l t2	Lists detailed information about the file t2, including its permissions after the chmod command in step 180.





chown aswani t2	Changes the ownership of the file t2 to the user aswani.
ls -l t2	Lists detailed information about the file t2, including its permissions, ownership, and group.
chgrp wheel t2	Changes the group ownership of the file t2 to the group wheel.
ls -l t2	Lists detailed information about the file t2 after the ownership and group ownership changes.

ls -l t3	Lists detailed information about the file t3, including its permissions, ownership, and group.
chown aswani:wheel t3	Changes the ownership of the file t3 to the user aswani and the group wheel.
ls -l t3	Lists detailed information about the file t3 after the ownership and group ownership changes.



cat > /ReadMe.txt	Creates a new file named ReadMe.txt in the root directory (/) and allows the user to input text.
ls -l /ReadMe.txt	Lists detailed information about the file ReadMe.txt located in the root directory.
chmod o-r /ReadMe.txt	Removes read permissions for others from the file ReadMe.txt.
ls -l /ReadMe.txt	Lists detailed information about the file ReadMe.txt, including its permissions after the chmod command in step 191.
chgrp wheel /ReadMe.txt	Changes the group ownership of the file ReadMe.txt to the group wheel.
ls -l /ReadMe.txt	Lists detailed information about the file ReadMe.txt, including its permissions, ownership, and group after the chgrp command in step 193.