

Linux File Commands

Linux file commands allow you to navigate, create, modify, and manage files and directories. Below are some essential and advanced file-related commands with their practical examples and outputs.

```
$ touch notes.txt
$ ls
notes.txt
```

Creates an empty file named notes.txt. The 'touch' command is often used to create new files or update timestamps.

```
$ echo 'Welcome to Linux File Management' > welcome.txt
$ cat welcome.txt
Welcome to Linux File Management
```

Creates a new file 'welcome.txt' and writes text into it using output redirection (>).

```
$ cp welcome.txt backup.txt
$ ls
backup.txt welcome.txt
```

Copies a file from one name to another or to a different location using 'cp'.

```
$ mv backup.txt /tmp/
$ ls /tmp/
backup.txt
```

Moves a file to another directory. Also used to rename files or directories.

```
$ mkdir projects
$ cd projects
$ touch main.py
$ ls
main.py
```

Creates a directory named 'projects', changes into it, and creates a new file inside.

```
$ rm welcome.txt
$ ls
notes.txt projects
```

Deletes a file permanently. Use with caution — there's no undo.

```
$ rm -r projects/
$ ls
notes.txt
```

Recursively deletes a directory and its contents using '-r'.

```
$ cat > shopping.txt
Milk
Eggs
```

```
Bread
(ctrl+d)
$ cat shopping.txt
Milk
Eggs
Bread
```

Creates a file interactively and writes content directly into it.

```
$ head -n 2 shopping.txt
Milk
Eggs
```

Displays the first 2 lines of a file.

```
$ tail -n 1 shopping.txt
Bread
```

Displays the last line of a file.

```
$ chmod 644 notes.txt
$ ls -l notes.txt
-rw-r--r-- 1 sheetal sheetal 50 Oct 31 notes.txt
```

Changes file permissions to make it readable by everyone but writable only by the owner.

```
$ find . -name '*.txt'
./notes.txt
./shopping.txt
```

Searches for files with a specific pattern recursively starting from the current directory.