

# File Handling Commands

File handling commands can be tedious, but knowing how to use them effectively can save you time and make your work easier. In this presentation, we'll explore the most essential commands with examples to help you master them.

A  
by Abhilash Jadala



# Getting Started

Before we dive into the different file handling commands, let's get familiar with the most basic command: **ls**. The ls command lists all files and directories in the current directory.



## List of Directories

Directories in the current folder are displayed with blue font on screen.

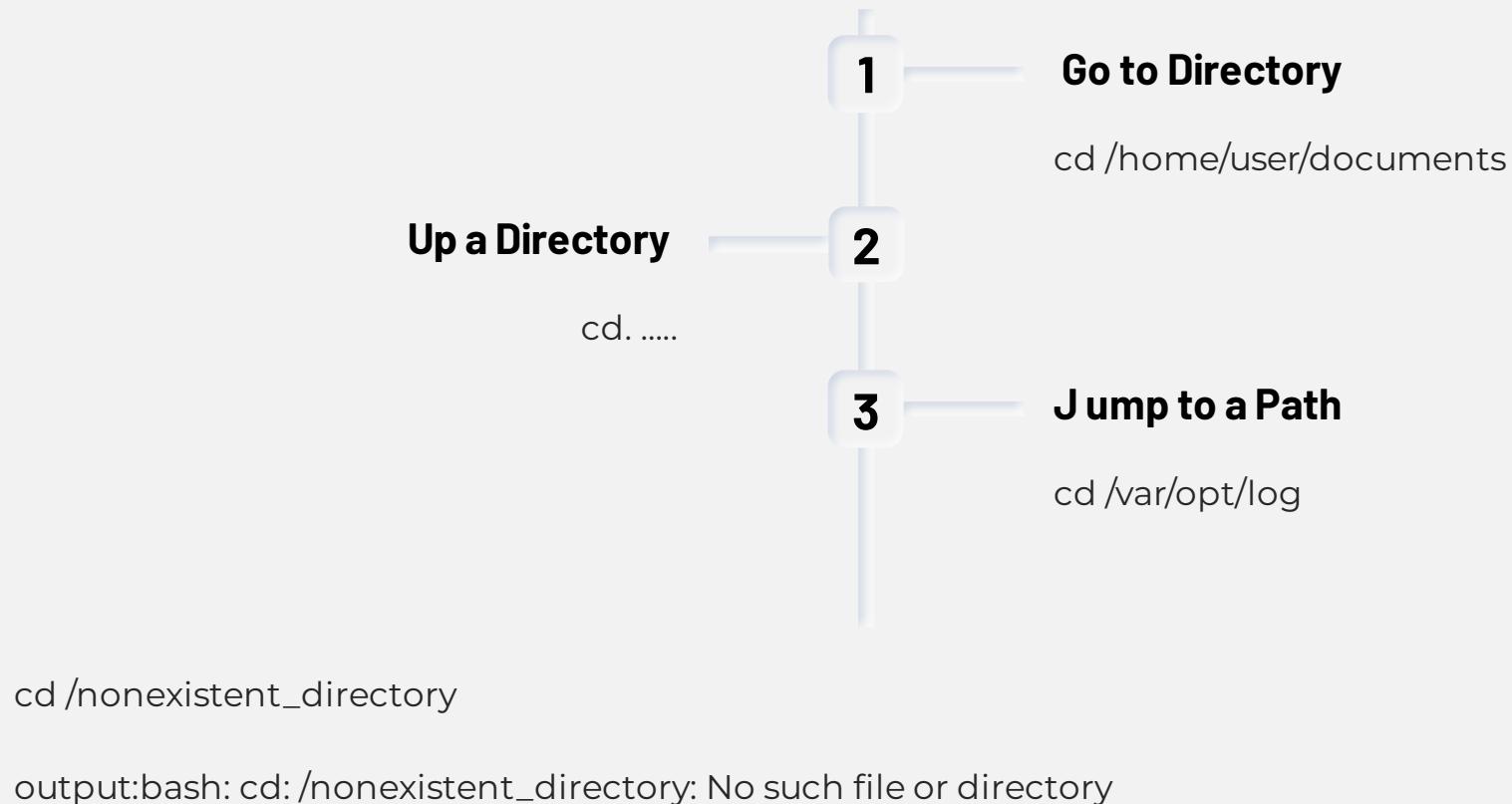
```
drwxr-xr-x 12 root root 4096 Dec 24 21:31 .
drwxr-xr-x  3 root root 4096 Dec 24 21:30 ..
drwxr-xr-x  7 root root 4096 Dec 24 21:31 app
-rw xr-xr-x  1 root root 1686 Dec 24 21:31 artisan
drwxr-xr-x  3 root root 4096 Dec 24 21:31 bootstrap
-rw xr-xr-x  1 root root 1753 Dec 24 21:31 composer.json
-rw xr-xr-x  1 root root 201569 Dec 24 21:31 composer.lock
drwxr-xr-x  2 root root 4096 Dec 24 21:31 config
drwxr-xr-x  5 root root 4096 Dec 24 21:31 database
-rw xr-xr-x  1 root root 521 Dec 24 21:31 .env.example
drwxr-xr-x  8 root root 4096 Dec 24 23:10 .git
-rw xr-xr-x  1 root root 111 Dec 24 21:31 .gitattributes
-rw xr-xr-x  1 root root 146 Dec 24 21:31 .gitignore
-rw xr-xr-x  1 root root 1125 Dec 24 21:31 package.json
-rw xr-xr-x  1 root root 1040 Dec 24 21:31 phpunit.xml
drwxr-xr-x  4 root root 4096 Dec 24 21:31 public
-rw xr-xr-x  1 root root 3703 Dec 24 21:31 readme.md
drwxr-xr-x  5 root root 4096 Dec 24 21:31 resources
drwxr-xr-x  2 root root 4096 Dec 24 21:31 routes
-rw xr-xr-x  1 root root 563 Dec 24 21:31 server.php
drwxr-xr-x  5 root root 4096 Dec 24 21:31 storage
drwxr-xr-x  4 root root 4096 Dec 24 21:31 tests
```

## List of Files

Files in the current folder are displayed with black font on screen.

# Navigating with cd

The **cd** (Change Directory) command is used to move between directories. With this command, you can navigate to a directory, move up a level, or jump to a specific path. If you need to access your files quickly, this is an essential command to know.



# Viewing with cat

If you want to view the contents of a file in the terminal, you can use the **cat** command. This command prints the entire contents of a file to the screen, so you can easily see what's inside.

## Single File view

```
cat file.txt
```

## Multiple File Concatenation

```
cat file1.txt file2.txt
```

## File Creation with cat

```
cat > newfile.txt
```

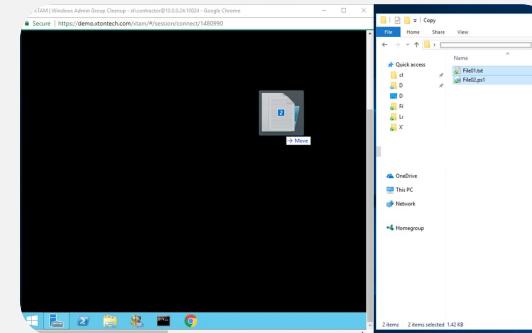
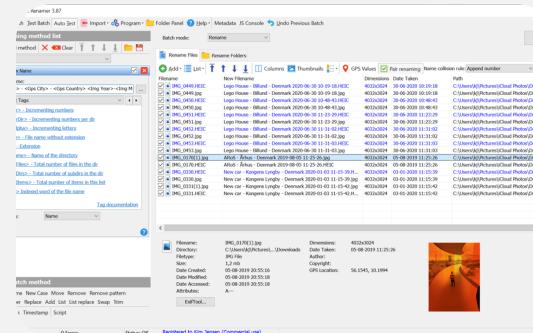
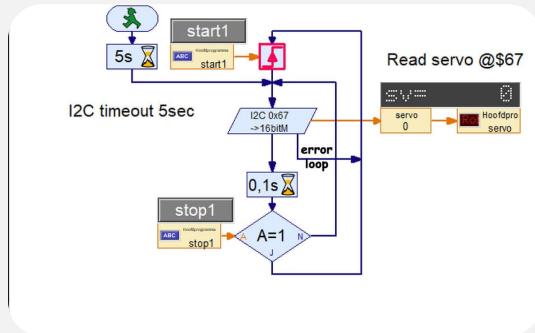
## Append to a File

```
cat >> oldfile.txt
```



# Moving Files with mv

The **mv** command is useful for moving files from one location to another or simply renaming files.



## Move a File

```
mv file1.txt  
/home/user/documents/
```



## Interactive File Movement

```
mv -i file1.txt  
/home/user/documents/
```

## File Renaming

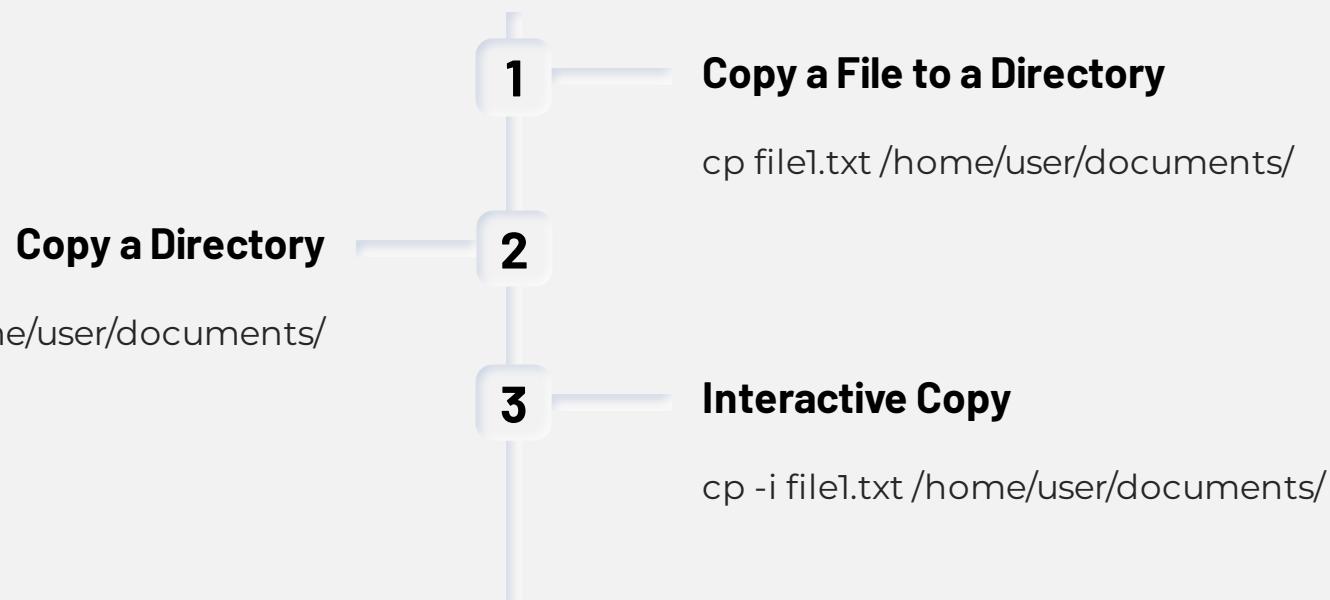
```
mv file1.txt presentation.txt
```

## Multiple File Movement

```
mv file1.txt file2.txt  
/home/user/documents/
```

# Copying Files with cp

The **cp** command is used for copying files from one directory to another. If you want to create a backup for a file or have multiple copies of a file in different directories, this command is your go-to option.



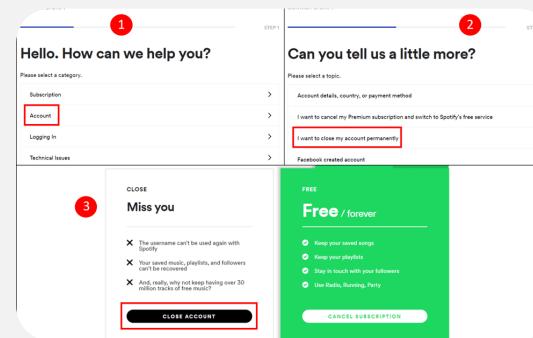
# Deleting Files with rm

The **rm** command is used for deleting files and directories. Be careful with this command as there is no undo or recycling feature. Once you delete a file or directory, it's gone!



## Delete a File

```
rm file.txt
```



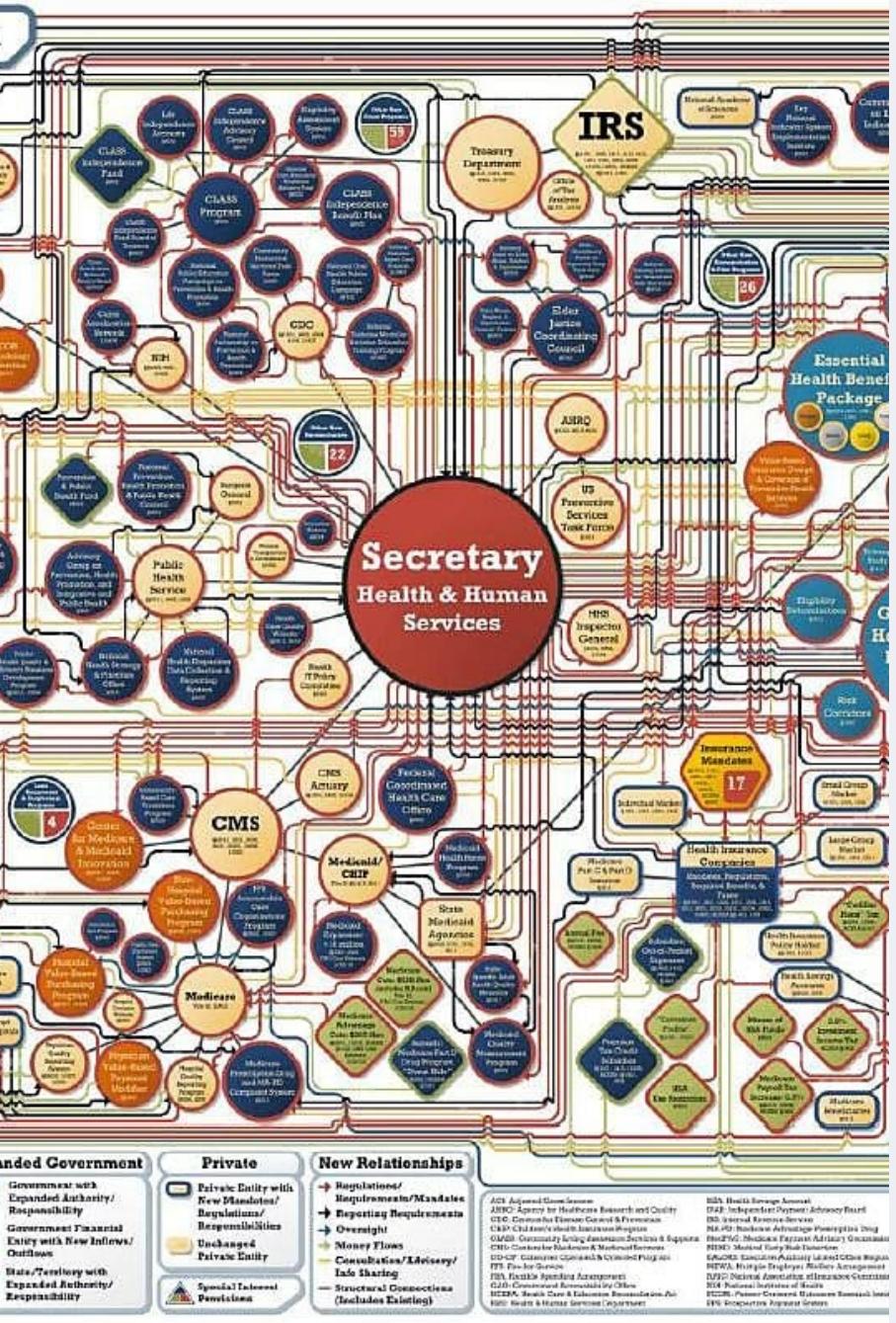
## Delete a Directory Recursively

```
rm -r directory-name
```



## Forcefully Delete

```
rm -f file.txt
```



# Understanding File Permissions

File permissions can be a confusing topic, but they're important to understand when working with files on a Unix-based system. In this card, we'll explore how to view and modify file permissions. You'll learn about the different types of permissions and how to use them to control who can read, write, or execute a file.

# Summing Up

Now you know some of the most important file handling commands and how to use them! It's important to remember to be careful when handling files using the terminal as these commands can wipe data without any warnings.

