Terminal Command Essentials

Hi! I'm **Syed Muzammil**, a front-end developer.

I'm learning new things to achieve my long-term goals and help others along the way.

Source: "Practical Introduction to Command Line" by Stanimira Vlaeva.

Command Control

Use **Ctrl + C** to abort a running command immediately.

Unlike clear, this keeps your previous output visible.

● COMMAND

```
Ctrl + C
```

Shell Identification

Type `echo $0` to verify which shell language (e.g., bash, zsh) is currently active.

● COMMAND

```
echo $0
```

Workspace Hygiene

Clear your terminal view using the clear command or the shortcut.

● COMMAND

```
clear
```

```
Shortcut: Ctrl + L
```

Efficiency / History

Don't retype! Press the **Up Arrow** key to cycle through and execute previous commands.

● COMMAND

```
↑ Arrow Key
```

Handling Special Characters

Spaces in filenames confuse the terminal. You must "escape" them.

Use a backslash \ or wrap the name in quotes.

● OPTION 1: BACKSLASH ESCAPE

```
mkdir My\ Folder
```

OPTION 2: DOUBLE QUOTES

```
mkdir "My Folder"
```

OPTION 3: SINGLE QUOTES

```
mkdir 'My Folder'
```

Orientation

Run **`pwd`** (Print Working Directory) to see exactly where you are in the file system.

● COMMAND

```
pwd
```

Listing Content

Use `ls` to view all files and folders in your current location.

● COMMAND

```
ls
```

Navigation

Use `cd <Folder Name>` to enter a directory.

● COMMAND

```
cd <Folder Name>
```

Reverse Navigation

Use `cd  .` to go back one folder.

● COMMAND

```
cd ..
```

```
cd ../.. (Jump back two levels)
```

Remote Listing

View a folder's contents without entering it by adding the path.

● COMMAND

```
ls <path/to/folder>
```

Auto-Completion

Type the first few letters of a file or folder and hit **Tab**.

Bash will auto-complete the name for you.

● COMMAND

```
Tab Key
```

Root Navigation

Run `cd /` to jump straight to the root directory (base of the system).

● COMMAND

```
cd /
```

Return Home

Typing `cd` (with no arguments) instantly transports you to your user Home directory.

● COMMAND

```
cd
```

Creating Directories

Use `mkdir` to create folders.

COMMAND

```
mkdir <name>
```

```
mkdir folder1 folder2 (Create multiple)
```

Naming Conventions

To create a folder with spaces, don't forget the quotes!

● COMMAND

```
mkdir "My Photos"
```

Creating Files

Use `touch` to create empty files instantly.

● COMMAND

```
touch <filename>
```

Writing Content

Use echo to write text into a file.

> overwrites. >> appends.

● COMMAND

```
echo "Message" > file.txt
```

```
echo "More text" >> file.txt
```

Viewing Content

Use `cat` to display a file's contents directly.

● COMMAND

```
cat <filename>
```

Concatenation

Merge multiple files into one using the redirect symbol.

● COMMAND

```
cat file1 file2 > combined_file
```

< Previous Next >

Detailed View

Use `ls -l` for a "long listing" (permissions, size, date).

● COMMAND

```
ls -l
```

Filtering

Use wildcards (*) to list specific file types.

● COMMAND

```
ls *.jpg
```

Moving/Renaming

Use `mv` to move files or rename them.

● COMMAND

```
mv <source> <destination>
```

Deleting Files

Use `rm` to delete a file.

⚠ WARNING: PERMANENT

● COMMAND

```
rm <filename>
```

Batch Deletion

Use curly braces to delete a specific set of files at once.

● COMMAND

```
rm {file1,file2}.txt
```

Deleting Directories

Use `rmdir` to remove an **empty** folder.

● COMMAND

```
rmdir <directory>
```

Recursive Deletion

Use `rm -r` to remove a folder and all its contents.

● COMMAND

```
rm -r <directory>
```

‹ Previous Finish