



Command-Line Essentials & Beyond

Windows | macOS | Linux

Why Learn the Command Line?

- **Speed** – tasks in seconds vs. clicks.
- **Automation** – scripts & batch jobs.
- **Remote work** – servers often have **no GUI**.
- **Super-powers** – access tools the GUI hides.

Getting a Terminal Window

OS / Editor	Open the terminal
Windows	Start ► <i>cmd</i> , or Win + R → <i>cmd</i> , or Terminal app
macOS	Spotlight (⌘ Space) → <i>Terminal</i>
VS Code	Ctrl/⌘ + ` (back-tick)
Admin	Right-click <i>cmd</i> → <i>Run as administrator</i>

Tip: Power users install **Windows Terminal** for tabs & panes.

Anatomy of a Command

```
<command> [options] [arguments]
```

- **command** – the program you run
- **options / flags** – modify behaviour (often start with `/` or `-`)
- **arguments** – files, folders, URLs, ...

Navigation Commands – Quick Reference

Purpose	Windows (cmd)	macOS / Linux
Show path	cd	pwd
Move into folder	cd Pictures	same
Up one level	cd ..	same
List files	dir	ls -la
Make folder	mkdir Assets	same
Clear screen	cls	clear

Navigation – Example Session

```
C:\> cd Users\Henry\Documents
C:\Users\Henry\Documents> mkdir Assets
C:\Users\Henry\Documents> dir
05/01/25    <DIR>          .
05/01/25    <DIR>          ..
05/01/25    <DIR>          Assets
04/30/25           2,048 notes.txt
C:\Users\Henry\Documents> cd Assets
C:\Users\Henry\Documents\Assets> cd ..
C:\Users\Henry\Documents> cd ..
C:\Users\Henry> cls
C:\Users\Henry>
```

File Operations – Quick Reference

Task	Command
Move file	<code>move src.txt D:\Docs</code>
Copy file	<code>copy src.txt backup.txt</code> (<i>Win</i>) / <code>cp</code> (<i>Unix</i>)
Remove file	<code>del file.txt</code> / <code>rm file.txt</code>
Echo text → file	<code>echo Hello > note.txt</code>
Exit shell	<code>exit</code>

File Operations – Example Session

```
C:\Projects> echo First line > readme.md
C:\Projects> copy readme.md backup.md
           1 file(s) copied.
C:\Projects> move backup.md Docs\
           1 file(s) moved.
C:\Projects> del readme.md
C:\Projects> dir Docs
05/01/25          0  backup.md
```


Visualise Folders with `tree`

```
# Show hierarchy of current dir  
tree
```

```
C:\Projects  
├── app.py  
├── requirements.txt  
└── data  
    ├── raw  
    └── processed
```

```
# Include files  
tree /F
```

Great for documenting project structure.

Download Anything with **wget**

```
# Simple download (Unix)
wget https://example.com/file.zip
```

```
--2025-05-01 12:34:56--  https://example.com/file.zip
Resolving example.com (example.com)... 93.184.216.34
Connecting to example.com (example.com)|93.184.216.34|:443... connected.
HTTP request sent, awaiting response... 200 OK
Length: 10485760 (10M) [application/zip]
Saving to: 'file.zip'
file.zip      100%[=====>]  10.00M  4.20MB/s   in 2.4s
2025-05-01 12:34:59 (4.20 MB/s) - 'file.zip' saved [10485760/10485760]
```

```
# Save with new name & resume if interrupted
wget -c -O latest.zip https://example.com/big.zip
```

wget Power Options – Cheatsheet

Flag	Function	Example
<code>-r</code>	Recursive site grab	<code>wget -r https://blog.com</code>
<code>-b</code>	Background	<code>wget -b URL</code>
<code>--limit-rate=200k</code>	Throttle	<code>wget --limit-rate=200k URL</code>
<code>--user</code> / <code>--password</code>	Auth	<code>wget --user=henry --password=secret URL</code>
<code>--no-check-certificate</code>	Ignore SSL	<code>wget --no-check-certificate URL</code>

Cheatsheet Recap

Navigation	<code>cd, dir/ls, mkdir, cls/clear</code>
Files	<code>copy cp, move mv, del rm, echo</code>
Visualise	<code>tree /F</code>
Download	<code>wget -c URL</code>

90 % of daily CLI work uses these commands – now with examples you can paste & try.

Further Resources

- **Microsoft Docs:** aka.ms/commandline
- **Linux TLDP Bash Guide:** tldp.org
- **Oh My Posh / Oh My Zsh:** prompt themes
- **commandlinefu.com:** crowdsourced one-liners

Ready to Explore!

Try the example sessions on your own system, tweak paths, and watch the terminal respond.

Muscle memory starts **today** – happy hacking!

