Windows CMD Essentials (Windows command line)

A comprehensive cheat sheet for Windows Command Line (CMD), designed for beginners and intermediate users. This guide covers essential commands, file management, system administration, networking, and basic batch scripting, with practical examples and pro tips.

Core Fundamentals: Navigation & System Info

NAVIGATION & FILES

`cd <directory>`	Change Directory. Navigate to a specified folder. **Examples:** `cd C:\Users\Public` `cd ..` (Go up one level) `cd \` (Go to root of current drive)
`dir`	List directory contents. **Examples:** `dir` (List files/folders in current directory) `dir /w` (Wide format) `dir /s` (Subdirectories) `dir *.log` (Files with .log extension)
`tree`	Display directory structure as a tree. **Examples:** `tree` `tree /f` (Display files as well)
`copy <source> <destination>`	Copy files or directories. **Examples:** `copy report.txt C:\Backup` `copy *.doc C:\Documents\Reports` (Wildcard)
`move <source> <destination>`	Move files or directories. **Examples:** `move old_file.txt new_location\new_file.txt` `move C:\Temp\* D:\Archive` (Move all from Temp)
`del <file(s)>`	Delete files. **Examples:** `del unwanted.tmp` `del *.bak /q` (Quiet delete all .bak files) `del /s /q C:\Logs\*.log` (Delete logs recursively)
`ren <old_name> <new_name>`	Rename files or directories. **Example:** `ren report.txt final_report.txt`
Wildcards & Paths	`*` (Any characters) `?` (Any single character) **Absolute Path:** `C:\Program Files\App` (Starts from root) **Relative Path:** `..\folder\file.txt` (Relative to current directory) **Pro Tip:** Use `Tab` for autocompletion of file and directory names.

`systeminfo`	Display detailed system configuration information. **Example:** `systeminfo	more` (Pipe to more for pagination)
`tasklist`	List all running processes. **Examples:** `tasklist` `tasklist /svc` (Show services hosted by processes) `tasklist /fi "IMAGENAME eq chrome.exe"` (Filter by image name)
`taskkill /pid <PID> /f`	Terminate a process by Process ID (PID) or image name. **Examples:** `taskkill /pid 1234 /f` (Force kill process with PID 1234) `taskkill /im notepad.exe /f` (Force kill Notepad)
`hostname`	Display the name of the host computer. **Example:** `hostname`
`echo <message>`	Display messages or turn command echoing on/off in scripts. **Examples:** `echo Hello World!` `echo %PATH%` (Display PATH environment variable)
`set`	Display, set, or remove CMD environment variables. **Examples:** `set` (List all environment variables) `set MYVAR=Hello` (Set a new variable) `echo %MYVAR%` (Display variable)
Viewing Environment Variables	Use `set` to see all variables or `echo %VAR_NAME%` for specific ones. **Example:** `echo %USERNAME%` **Pro Tip:** For persistent environment variable changes, use `System Properties > Advanced > Environment Variables`.

Network & Disk Management

NETWORKING

`ipconfig`	Display current TCP/IP network configuration. **Examples:** `ipconfig` (Basic info) `ipconfig /all` (Detailed info) `ipconfig /release` (Release IP address) `ipconfig /renew` (Renew IP address)

`ping <destination>`	Test network connectivity to a host.
	
	Examples:
	`ping google.com`
	`ping 192.168.1.1`
	`ping -t 8.8.8.8` (Continuous ping, Ctrl+C to stop)

`tracert <destination>`	Trace the route packets take to a network host. **Example:** `tracert google.com`
`netstat -ano`	Display active TCP connections, listening ports, and associated process IDs. **Examples:** `netstat -ano` `netstat -ano	findstr "LISTENING"` (Show only listening ports) `netstat -an	findstr ":80"` (Find connections on port 80)

`nslookup <hostname/IP>`	Query DNS for domain name or IP address mapping.
	Examples:
	`nslookup google.com`
	`nslookup 8.8.8.8`
`net use`	Connects to or disconnects from shared network resources.
	Examples:
	`net use Z: \\SERVER\Share` (Map a network drive)
	`net use Z: /delete` (Disconnect network drive)
	`net use` (List current network connections)
	Pro Tip: `ipconfig /flushdns` can often resolve website loading issues by clearing your local DNS cache.

`chkdsk <drive> /f`	Check a disk for errors and fix them. **Examples:** `chkdsk C: /f` (Fix errors on C: drive, requires reboot if in use) `chkdsk D: /r` (Locate bad sectors and recover readable information)

`diskpart`	Disk partitioning utility. Enters a separate interactive shell.
	
	Basic Steps:
	`list disk`
	`select disk <n>`
	`list partition`
	`exit`
	
	Warning: Use with extreme caution, as incorrect commands can lead to data loss.

`format <drive> /fs:<filesystem>`	Format a disk or partition. **Examples:** `format E: /fs:NTFS` (Format E: drive as NTFS) `format F: /q /fs:FAT32` (Quick format F: as FAT32)
`attrib [+R|-R] [+A|-A] [+S|-S] [+H|-H] <file/folder>`	Display or change file attributes. **Attributes:** `R` - Read-only `A` - Archive `S` - System `H` - Hidden **Examples:** `attrib +h secret.txt` (Hide file) `attrib -h secret.txt` (Unhide file) `attrib +r important.doc` (Make read-only)
`xcopy <source> <destination> /E /I /H /K /O /X`	Advanced file copy utility, supports copying directories and subdirectories. **Common Switches:** `/E` - Copy directories and subdirectories, including empty ones. `/I` - If destination doesn't exist and copying multiple files, assume destination is a directory. `/H` - Copy hidden and system files. `/K` - Copy attributes. `/O` - Copy file ownership and ACL information. `/X` - Copy file audit settings. **Example:** `xcopy C:\MyDocs D:\Backup /E /H`

`robocopy <source> <destination> [options]`	Robust File Copy. More powerful than xcopy, designed for reliable copying, especially over networks.
	
	Common Options:
	`/S` - Copy subdirectories (excluding empty ones).
	`/E` - Copy subdirectories (including empty ones).
	`/MT[:n]` - Multithreaded copy (default 8 threads).
	`/Z` - Restartable mode (for unreliable networks).
	`/MIR` - Mirror a directory tree (delete files in dest that don't exist in source).
	
	Examples:
	`robocopy C:\Data D:\Backup /E /MT:16`
	`robocopy \\Server\Share C:\LocalSync /MIR /Z`
	
	Pro Tip: `robocopy` is your best friend for large syncs or backups, especially `/MIR` for maintaining exact copies.

User Management & Automation

USER & PERMISSIONS

`net user`	Manage user accounts. **Examples:** `net user` (List all local user accounts)

	`net user <username>` (Display info for a specific user)
	`net user JohnDoe password123 /add` (Add new user)
	`net user JohnDoe /delete` (Delete user)
`whoami`	Display the current user's domain and username.
	Example:
	`whoami`
	`whoami /groups` (Show groups the user belongs to)
	`whoami /priv` (Show privileges)
`runas /user:<username> "<command>"`	Run a command or program as a different user.
	Example:
	`runas /user:Administrator "cmd.exe"` (Open CMD as Administrator, prompts for password)
	Pro Tip: To open an administrative CMD directly, search for "cmd" in the Start Menu, right-click, and select "Run as administrator".
Changing User Passwords	To change a local user's password:
	`net user <username> <new_password>`
	Example:
	`net user Guest NewPassword123`
	To force a password change at next login:
	`net user <username> /logonpasswordchg:yes`
Checking User Groups	Use `whoami /groups` to see all groups the current user is a member of.
	To check groups for another user (local):
	`net localgroup` (List all local groups)
	`net localgroup Administrators` (List members of Administrators group)
	Pro Tip: Understanding user permissions and groups is crucial for security. Always follow the principle of least privilege.

BATCH & AUTOMATION

.bat Scripting Basics	Batch files (`.bat` or `.cmd`) are sequences of CMD commands executed line by line.
	
	Syntax:
	• `@echo off` - Prevents commands from being displayed.
	• `REM <comment>` - Add comments.
	• `:: <comment>` - Another way to add comments.
	• `echo <message>` - Display text.
	• `pause` - Pause script execution until a key is pressed.
	
	Example `my_script.bat` :
	
	<pre>@echo off
	REM This is a sample batch script
	echo Hello, User!
	set /p USER_NAME="Enter your name: "
	echo Nice to meet you, %USER_NAME%!
	pause</pre>

`if` statement	Conditional execution.
	
	Syntax:
	`IF [NOT] EXIST <filename> <command>`
	`IF %variable%==<string> <command>`
	`IF ERRORLEVEL <number> <command>`
	
	Examples:
	`IF EXIST C:\file.txt echo File exists.`
	`IF "%OS%"=="Windows_NT" echo Running on Windows.`
	`IF %ERRORLEVEL% NEQ 0 echo An error occurred.`

`for` loop	Iterate over sets of files, directories, or numbers. **Syntax (files):** `FOR %%variable IN (set) DO command` **Examples:** `FOR %%f IN (*.txt) DO echo %%f` (List all .txt files) `FOR /D %%d IN (*) DO echo %%d` (List all directories) `FOR /L %%i IN (1,1,10) DO echo %%i` (Loop from 1 to 10)

`pause`	Pauses the execution of a batch program and displays a message to the user to press any key to continue.
	
	Example:
	
	<pre>echo This part will execute first.
	pause
	echo This part will execute after you press a key.</pre>

`goto <label>`	Directs the batch program to a line identified by a label.

Syntax:

```
:label_name
```

Example:

```
@echo off
goto START

:ERROR_HANDLER
echo An error occurred!
goto END

:START
echo Starting script...
rem Simulate an error
exit /b 1

:END
echo Script finished.
pause
```

Pro Tip: Use `call` to execute another batch script and return to the current one, rather than `goto` which will not return.

`schtasks`	Schedule commands or programs to run at specified times.

Examples:

`schtasks /create /tn "DailyBackup" /tr "C:\scripts\backup.bat" /sc daily /st 03:00` (Create a daily task)

`schtasks /query /tn "DailyBackup" /v` (View task details)

`schtasks /run /tn "DailyBackup"` (Run task immediately)

`schtasks /delete /tn "DailyBackup" /f` (Delete task)

Pro Tip: For complex schedules, the Task Scheduler GUI (`taskschd.msc`) is often easier to use, but `schtasks` is great for scripting and automation.