

\$ cp

Master cp — Interactively

Why the terminal is the fastest way to duplicate files

GO

Instant feedback

Press Enter, see it happen — no compiling.

RE

Learn by tweaking

Change one flag, rerun, and watch it change.

IO

Real files, real power

Duplicate files and folders in seconds.

OK

Zero setup

Built into every Linux system — just open a shell.

```
$ cp report.txt \  
backup.txt
```

Copied!

~10 sec

01

Copy A Single File

Duplicate a file under a new name or location

- `cp` source dest copies the file
- The original file is left untouched
- Overwrites dest if it already exists

SYNTAX

```
cp report.txt backup.txt
```



user@linux: ~

```
$ cp report.txt \  
    backup.txt
```

Copied!



02

Copy A Whole Directory

Duplicate an entire folder and its contents

- -r copies directories recursively
- Every file and subfolder comes along
- Required any time the source is a folder

SYNTAX

```
cp -r site/ site-backup/
```



user@linux: ~

```
$ cp -r site/ \
site-backup/
```

Folder copied



03

Copy Without Overwriting

Play it safe and never lose an existing file

- `-n` skips the copy if dest already exists
- Prevents accidental overwrites
- Combine with `-v` to see what happened

SYNTAX

```
cp -n data.csv backup/
```



user@linux: ~

```
$ cp -n data.csv \  
backup/
```

Skipped (exists)



04

Preserve File Metadata

Keep timestamps, permissions, and ownership intact

- `-p` preserves mode, timestamps, and owner
- Useful for backups and system migrations
- Without it, copies get fresh timestamps

SYNTAX

```
cp -p config.yml backup/
```



user@linux: ~

```
$ cp -p config.yml \  
backup/
```

Metadata kept





```
$ echo "Thanks for watching!"
```

Thank You For Watching

You just learned 4 real cp moves — now go run them yourself.

▶ SUBSCRIBE

@MyWebUniversity

on YouTube — for more Linux & command-line tutorials