# cut Cheatsheet

By Dejan Panovski • Updated on Jun 17, 2026 • Download PDF

Quick reference for extracting fields, bytes, and characters with cut in Linux

The cut command extracts sections from each line of files or standard input. This cheatsheet covers field, byte, and character selection, custom delimiters, and common pipeline usage.

## Select Fields

Pick columns with `-f` (default delimiter is TAB).

| | |
|---|---|
| `cut -f1 file.txt` | Select the first field |
| `cut -f1,3 file.txt` | Select fields 1 and 3 |
| `cut -f1-4 file.txt` | Select fields 1 through 4 |
| `cut -f2- file.txt` | From field 2 to the end of the line |
| `cut -f-3 file.txt` | From the first field through field 3 |

## Delimiters

Set the input delimiter and control output.

| | |
|---|---|
| `cut -d',' -f1 file.csv` | Use a comma as the delimiter |
| `cut -d':' -f1,3 file` | Colon delimiter, fields 1 and 3 |
| `cut -d' ' -f2 file` | Space delimiter, second field |
| `cut -s -d',' -f1 file` | Skip lines with no delimiter |
| `cut --output-delimiter='_' -f1,3 file` | Set a different output delimiter |

## Bytes and Characters

Cut by position instead of field.

| | |
|---|---|
| `cut -c1-10 file` | First 10 characters of each line |
| `cut -c3- file` | From character 3 to the end |
| `cut -c1,4,7 file` | Characters at positions 1, 4, and 7 |
| `cut -b1-5 file` | First 5 bytes of each line |
| `cut -b5,9,13 file` | Bytes at positions 5, 9, and 13 |

## Invert and Combine

Exclude a selection or chain with other tools.

| | |
|---|---|
| `cut -f1,3 --complement file` | Print every field except 1 and 3 |
| `cut -c1-5 --complement file` | Print every character except 1 to 5 |
| `cut -d':' -f1,7 /etc/passwd` | Usernames and login shells |
| `echo "$PATH" \| cut -d':' -f1` | First entry in the `PATH` variable |
| `cut -d',' -f2 data.csv` | Extract the second CSV column |

## Common Pipelines

Practical combinations with other commands.

| | |
|---|---|
| `getent passwd \| cut -d':' -f1` | List all system users |
| `grep "^$USER:" /etc/passwd \| cut -d':' -f7` | Default shell for the current user |
| `echo "one two three" \| tr -s ' ' \| cut -d' ' -f2` | Collapse repeated spaces, then cut |
| `ls -l file.txt \| cut -c1-10` | Permission string for one file |
| `who \| cut -d' ' -f1 \| sort -u` | Unique logged-in users |

## Troubleshooting

Quick checks for common `cut` issues.

| | |
|---|---|
| `Whole line printed unchanged` | Wrong delimiter; the default is TAB, set `-d` to match the input |
| `Lines without a delimiter still show` | Add `-s` to skip lines containing no delimiter |
| `Cannot use -f, -b, -c together` | Use only one selection mode per command |
| `Multibyte characters look garbled` | Use `-c` to cut by character, not `-b` (bytes) |
| `Repeated spaces split oddly` | `cut` treats each delimiter literally; normalize with `tr -s` or use `awk` |

## Related Guides

Use these guides for full text-processing workflows.

| | |
|---|---|
| cut Command in Linux | Full `cut` guide with examples |
| awk Command in Linux | Pattern-based field extraction |
| tr Command in Linux | Translate and squeeze characters |
| sort Command in Linux | Order lines from files or input |
| grep Command in Linux | Search and filter matching lines |