

corewire

Linux

Navigation & Textverarbeitung

Folien-Hinweis

- Space, Page down: Nächste Folie
- Page up: Vorherige Folie
- ESC, o: Übersicht

[Zur Kapitelübersicht](#)

Arbeiten mit Dateien und Verzeichnissen

Das Dateisystem

- **Hierarchische Struktur** - Baum von Verzeichnissen
- **Root-Verzeichnis** `/` - Wurzel des Dateisystems
- **Absolute Pfade** - Beginnen mit `/` (z.B. `/home/user/docs`)
- **Relative Pfade** - Relativ zum aktuellen Verzeichnis

Navigations-Befehle

`pwd` - Print Working Directory

```
$ pwd  
/home/user
```

`cd` - Change Directory

```
$ cd /home/user/documents # Absoluter Pfad  
$ cd documents            # Relativer Pfad  
$ cd ..                   # Übergeordnetes Verzeichnis  
$ cd ~                    # Home-Verzeichnis  
$ cd -                    # Letztes Verzeichnis
```

Verzeichnisinhalte anzeigen - ls

Grundlegende Verwendung

```
$ ls           # Aktuelle Verzeichnisinhalte
$ ls /home    # Bestimmtes Verzeichnis
$ ls documents # Relativer Pfad
```

Wichtige Optionen

```
$ ls -l      # Detaillierte Ausgabe (long)
$ ls -a      # Alle Dateien (inkl. versteckte)
$ ls -la     # Kombinierte Optionen
$ ls -lh     # Human-readable Größen
$ ls -t      # Nach Zeit sortieren
$ ls -r      # Umgekehrte Reihenfolge
```

ls -l Ausgabe verstehen

```
$ ls -l
drwxr-xr-x 2 user group 4096 Jan 15 10:30 documents
-rw-r--r-- 1 user group 1024 Jan 14 15:45 readme.txt
```

- **d** = Verzeichnis, **-** = Datei
- **rwxr-xr-x** = Berechtigungen (Owner, Group, Others)
- **2** = Anzahl Links
- **user group** = Besitzer und Gruppe
- **4096** = Größe in Bytes
- **Jan 15 10:30** = Datum/Zeit der letzten Änderung

Dateien und Verzeichnisse erstellen

mkdir - Verzeichnisse erstellen

```
$ mkdir mein_ordner           # Einzelnes Verzeichnis
$ mkdir ordner1 ordner2       # Mehrere Verzeichnisse
$ mkdir -p pfad/zu/verzeichnis # Elternverzeichnisse erstellen
$ mkdir -p projekt/{src,docs,tests} # Mehrere Unterverzeichnisse
```

touch - Leere Dateien erstellen oder Zeitstempel ändern

```
$ touch datei.txt           # Neue leere Datei
$ touch datei1.txt datei2.txt # Mehrere Dateien
$ touch existing.txt        # Zeitstempel aktualisieren
```


Dateien und Verzeichnisse kopieren - cp

Dateien kopieren

```
$ cp quelle.txt ziel.txt           # Datei kopieren  
$ cp datei.txt /home/user/backup/ # In anderes Verzeichnis  
$ cp datei1.txt datei2.txt zielordner/ # Mehrere Dateien
```

Verzeichnisse kopieren

```
$ cp -r quellordner zielordner # Rekursiv kopieren  
$ cp -r ordner1/ ordner2/      # Mit Inhalt kopieren  
$ cp -a ordner backup/         # Archiv-Modus (Rechte erhalten)
```

Daten verschieben - mv

Dateien/Verzeichnisse verschieben

```
$ mv datei.txt neuer_ordner/      # Datei verschieben  
$ mv alter_name neuer_name        # Umbenennen  
$ mv ordner1/ ordner2/            # Verzeichnis verschieben
```

Mehrere Dateien verschieben

```
$ mv datei1.txt datei2.txt zielordner/ # Mehrere Dateien  
$ mv *.txt textordner/                 # Alle .txt Dateien
```

Dateien und Verzeichnisse löschen - rm

Dateien löschen

```
$ rm datei.txt           # Einzelne Datei
$ rm datei1.txt datei2.txt # Mehrere Dateien
$ rm *.tmp               # Wildcards verwenden
```

Verzeichnisse löschen

```
$ rm -r ordner_mit_inhalt # Rekursiv löschen
$ rm -rf ordner           # Force + Rekursiv (Vorsicht!)
```

Wildcards und Pattern Matching

Grundlegende Wildcards

```
$ ls *.txt           # Alle .txt Dateien
$ ls datei?.txt      # ? = ein beliebiges Zeichen
$ ls *test*          # Enthält "test"
$ ls [abc]*           # Beginnt mit a, b oder c
$ ls [0-9]*           # Beginnt mit Ziffer
```

Praktische Beispiele

```
$ cp *.jpg bilder/    # Alle JPEG Dateien kopieren
$ rm temp*            # Alle temporären Dateien
$ ls backup_202[0-3]* # Backups von 2020-2023
```

Inhalte lesen und bearbeiten

Dateiinhalte anzeigen - cat

Ganze Dateien anzeigen

```
$ cat datei.txt           # Datei ausgeben  
$ cat datei1.txt datei2.txt # Mehrere Dateien verketten  
$ cat *.txt              # Alle .txt Dateien
```

Nützliche Optionen

```
$ cat -n datei.txt        # Mit Zeilennummern  
$ cat -s datei.txt        # Mehrfache Leerzeilen reduzieren
```

Dateianfang und -ende - head/tail

head - Erste Zeilen anzeigen

```
$ head datei.txt           # Erste 10 Zeilen
$ head -n 20 datei.txt     # Erste 20 Zeilen
$ head -5 datei.txt        # Erste 5 Zeilen
$ head *.log               # Von mehreren Dateien
```

tail - Letzte Zeilen anzeigen

```
$ tail datei.txt           # Letzte 10 Zeilen
$ tail -n 15 datei.txt     # Letzte 15 Zeilen
$ tail -f logdatei.log     # Follow-Modus (live updates)
$ tail -F rotierendes.log  # Follow mit Rotation
```

Text suchen - grep

Grundlegende Suche

```
$ grep "suchtext" datei.txt      # Zeilen mit "suchtext" finden
$ grep suchtext *.txt           # In mehreren Dateien suchen
$ grep -r "pattern"verzeichnis/ # Rekursiv in Verzeichnis
```

Wichtige Optionen

```
$ grep -i "text" datei.txt      # Case-insensitive
$ grep -n "pattern" datei.txt   # Mit Zeilennummern
$ grep -c "pattern" datei.txt   # Anzahl Treffer zählen
$ grep -l "pattern" *.txt       # Nur Dateinamen zeigen
```


Erweiterte grep-Funktionen

Regular Expressions

```
$ grep "^Start" datei.txt      # Zeilen die mit "Start" beginnen
$ grep "Ende$" datei.txt      # Zeilen die mit "Ende" enden
$ grep "[0-9]" datei.txt      # Zeilen mit Ziffern
$ grep "ba." datei.txt        # ba + ein beliebiges Zeichen
$ grep "ba*" datei.txt        # b + null oder mehr a
```

Kontextsuche

```
$ grep -A 3 "pattern" datei.txt # 3 Zeilen nach Treffer
$ grep -B 2 "pattern" datei.txt # 2 Zeilen vor Treffer
$ grep -C 2 "pattern" datei.txt # 2 Zeilen vor und nach Treffer
```

Text editieren - nano

Nano starten

```
$ nano datei.txt          # Datei öffnen/erstellen  
$ nano                   # Leere Datei
```

Text editieren - nano

Wichtige Shortcuts in nano:

- `Ctrl+O` - Speichern (Write Out)
- `Ctrl+X` - Beenden
- `Ctrl+K` - Zeile ausschneiden
- `Ctrl+U` - Einfügen
- `Ctrl+W` - Suchen
- `Ctrl+G` - Hilfe anzeigen
- `Alt+U` - Rückgängig machen
- `Alt+E` - Wiederholen

Zur Kapitelübersicht

- Vorheriges Kapitel: [Einführung](#)
- Nächstes Kapitel: [Shell](#)