



1.103.3 Grundbefehle



www.lpi.org





Syntax:

```
ls [options] file1 [file2]
```

Anzeigen der Verzeichniseinträge.

Häufige Optionen (Schalter)

- l long, auch Rechte, Eigentümer, Gruppe, Dateigröße, ... werden angezeigt
- i inode, Inodes werden angezeigt
- a all, auch versteckte Dateien und Verzeichnisse werden angezeigt (Beginnen mit einem Punkt)



Shellbefehle können mit Optionen (Schalter) in ihrer Funktion verändert werden.

Dabei gibt es 3 Formen. Schalter wird mit

- kenntlich gemacht (Kurzform),
- kenntlich gemacht (Langform)
- ohne Zeichen (selten).

Beispiele:

- | | |
|---------------|---|
| ls | Anzeige des Verzeichnisinhalts |
| ls -l | Anzeige des Verzeichnisinhalts inkl. Rechten,
Dateigröße,... |
| ls --all | Anzeige des Verzeichnisinhalts inkl. versteckte Dateien
(entspricht ls -a) |
| tar xzf DATEI | gezipptes Tape-Archiv entpacken |

Sehr oft kann der Schalter -r bzw. -R oder --recursive verwendet werden. Dann wird der Befehl auch für alle Unterverzeichnisse ausgeführt.



Syntax:

`cd directory`

Gehe in das Verzeichnis *directory*.

Besondere Verzeichnisse:

- `.` Aktuelles Verzeichnis
- `..` Ein Verzeichnis höher
- `~` Homeverzeichnis des angemeldeten Users
- `~ab` Homeverzeichnis des Users *ab*.



Syntax:

`pwd`

Anzeigen des aktuellen Pfads.



Syntax:

```
cp [options] file1 file2
```

```
cp [options] files directory    (directory muss existieren!)
```

Kopiert Dateien oder Verzeichnisse. Achtung: Standardmäßig wird überschrieben (außer `-i` wird als Schalter verwendet; oft als alias).

Häufige Optionen (Schalter):

- `-f` force, Ziel wird ohne Nachfrage überschrieben
- `-i` interactively, ist Ziel vorhanden, wird nachgefragt
- `-p` preserve, erhalte Rechte, Eigentümer und Zeitstempel
- `-r,-R` recursiv, auch Unterverzeichniss mitkopieren
- `-v` verbose, Anzeige des Dateinamens beim Kopieren



Die Shell (!) verwendet automatisch Wildcards (\Rightarrow globbing).

- * Alle Zeichen ($x^* \Rightarrow x, x1, xabcdef, \dots$)
- ? Genau ein Zeichen ($x? \Rightarrow x1, xa, xg, \dots$, aber nicht x)
- [zeichen] Ein Zeichen welches innerhalb von [] steht ($x[yz] \Rightarrow xy, xz$)
- [!zeichen] Ein Zeichen welches **nicht** innerhalb von [] steht ($x[!yz] \Rightarrow xa, xb, \dots$, aber nicht xy)
- [a-z] Ein Zeichen zwischen a und z, sonst wie oben ($x[a-c] \Rightarrow xa, xb, xc$)
- [!a-z] Ein Zeichen welches nicht zwischen a und z liegt ($x[!x-z] \Rightarrow xa, xb, xm, \dots$)

Beispiel

?[ae][iy]er* \Rightarrow Mayer Lüdenscheid, Heier, ...

Möchte man das **Globbering unterbinden**, muss der Text in “ “ stehen.



Syntax:

`mkdir [options] directory`

Erzeugt ein Verzeichnis (sofern im Verzeichnis das Schreibrecht existiert).

Häufige Optionen (Schalter):

`-m mode` mode, Verzeichnis mit diesen Rechten erzeugen

`-p` parent, erzeuge auch die übergeordneten Verzeichnisse



Syntax:

`mv [options] quelle ziel`

Verschieben oder **umbenennen** von Dateien oder Verzeichnissen. Im selben Dateisystem sehr schnell.

Ziel existiert nicht => *quelle* wird in *ziel* umbenannt

Ziel existiert => *ziel* ist eine Datei => *ziel* wird überschrieben

=> *ziel* ist ein Verzeichnis => *quelle* wird in *ziel* abgelegt

Häufige Optionen (Schalter)

-f force, Ziel wird ohne Nachfrage überschrieben

-i interactively, ist Ziel vorhanden, wird nachgefragt



Syntax:

`rm [options] files`

Dateien löschen (sofern das Recht im Verzeichnis besteht). Es werden mit Schalter `-d`, `-r`, `-R` auch Verzeichnisse gelöscht.

Häufige Optionen (Schalter):

- `-i` interactively, ist Ziel vorhanden, wird nachgefragt
- `-f` force, auch schreibgeschützte Dateien ohne Nachfrage entfernen
- `-r,-R` recursiv, auch Unterverzeichnis mitlöschen



Syntax:

```
rmdir [options] directories
```

Verzeichnisse löschen (müssen leer sein!)

Häufige Optionen (Schalter):

-p parent, auch darüber liegende Verzeichnisse löschen, sofern diese leer sind

Achtung: Auch mit `rm -R` kann man Verzeichnisse löschen, selbst wenn sie nicht leer sind.



Syntax:

`touch [options] files`

Ändert die Zugriffs/Veränderungszeit einer Datei. Existiert die Datei nicht, wird sie angelegt.

Häufige Optionen (Schalter):

- a access, Nur die Zugriffszeit wird verändert
- m modification, nur die Änderungszeit wird verändert
- t time setzt die Zeit auf time ([[CC]YYMMDDhhmm[.ss] => 17.01.2006 13:07 -> 200601171307).



Syntax:

`file <FILE>`

Zeigt den Typ von <FILE> an.

Gibt an, um welche Art von Dteie es sich handelt, z.B.

- directory
- Unicode text
- Bash-Shellscrip executable
- ...



Syntax:

```
gzip <FILE1> <FILE2> ...  
gunzip <ZIP-FILE>
```

Packt <FILE1> <FILE2> .. bzw. entpackt nach dem Lemp-ZIV-Verfahren (LZ77).

Beim Packen wird i.d.R. die Endung .gz angehängt.

Gunzip kann auch zip, compress gpackte Dateien wieder entpacken.

Syntax:

```
bzip2 <FILE1> ...  
bunzip2 <ZIP-File>
```

Packt <FILE1> <FILE2> .. bzw. entpackt nach einem noch besseren Kompressionsalgorithmus.

Beim Packen wird i.d.R. die Endung .bz2 angehängt.



Syntax:

`cat file`

Listet den Dateiinhalt der Datei *file* und verbindet Dateiinhalte.

Seitenweise Ausgabe:

`cat | more`

RETURN	Weiterschaltung Zeilenweise
Leertaste	Weiterschaltung Seitenweise



Syntax:

less file

Dateibrowser zum Anzeigen der Datei *file*.

Wichtige less-Commandos:

h Hilfe

q quit, beendet less

?text sucht nach *text*. Mit Shift-N bzw. n kommt man von Treffer zu Treffer.

g Springe zum Dateianfang

Shift-G Springe zum Dateiende



Syntax:

man befehl.

Anzeige der Manual-Seiten des Befehls *befehl*.

Die Bedienung entspricht less (abhängig von der Umgebungsvariablen *PAGER*).



ls	Anzeigen von Dateien, Verzeichnissen
cd	Verzeichnispfad wechseln
pwd	Anzeige des aktuellen Pfads
cp	Kopieren von Dateien
mkdir	Verzeichnisse erstellen
mv	Dateien, Verzeichnisse umbenennen und verschieben
rm	Dateien und Verzeichnisse löschen
rmdir	Verzeichnisse löschen
touch	leere Datei erzeugen, Zugriffszeiten ändern
file	Anzeige des File-Typs
gzip	packen nach LZ77
gunzip	entpacken
bzip2	packen (besser als gzip)
bunzip2	entpacken von bzip2-gepackten Dateien
cat	Dateiinhalte auflisten
less	Dateiinhalte browsieren
man	Manualseiten anzeigen



Kontrollieren Sie jeweils das Ergebnis Ihrer Arbeit, bei folgenden Aufgaben:

1. Legen Sie das Verzeichnis */root/test* an
2. Legen Sie in diesem neuen Verzeichnis die leeren Dateien *test1* und *test2* an
3. Kopieren Sie das Verzeichnis *test* in das Verzeichnis */root/neu*
4. Nennen Sie die Datei */root/neu/test1* um in */root/neu/neu1*
5. Löschen Sie im Verzeichnis */root/test* die Dateien *test1* und *test2*
6. Löschen Sie das Verzeichnis */root/test*
7. Löschen Sie das Verzeichnis */root/neu* und seinen gesamten Inhalt mit einem Befehl