

# Variable **PATH** und Kommando **which**



# Inhaltsverzeichnis

- [Klassifizierung von Kommandos](#)
- [Aufruf von externen Kommandos](#)
- [Umgebungsvariable](#) PATH
- [Kommando](#) which
- PATH [-Variable von](#) root
- [Anpassung der](#) PATH [-Variablen](#)

# Klassifizierung von Kommandos

Die `bash` kennt drei verschiedene Arten von Kommandos:

- **Aliase:** Benutzer-definierte Kommando-Kürzel (werden später behandelt)
- **Funktionen:** Benutzer-definierte Kommandos, denen man auch Parameter übergeben kann (sind nicht Gegenstand dieses Kurses)
- **Interne Kommandos:** Kommandos, die in der `bash` selbst implementiert sind (werden später behandelt)

- **Externe Kommandos:** Alle anderen Kommandos, die in separaten Programm-Dateien im Dateisystem (auf einem Datenträger) vorliegen. Diese Kommandos können auf zwei Arten aufgerufen werden:
  - durch Angabe des (absoluten oder relativen) Pfades zur Programm-Datei (dazu muss das Kommandowort mindestens einen `/` enthalten)
  - ohne Pfad-Angabe, wenn das Kommandowort in einem Verzeichnis liegt, das in der Umgebungsvariablen `PATH` enthalten ist

# Aufruf von externen Kommandos

## Aufruf mit absolutem Pfad:

```
hermann@debian:~$ /usr/bin/ls  
Bilder Desktop Dokumente Downloads Musik Öffentlich Videos Vorlagen
```

```
hermann@debian:~$ /bin/ls  
Bilder Desktop Dokumente Downloads Musik Öffentlich Videos Vorlagen
```

Das Kommando `ls` ist offenbar in den Verzeichnissen `/usr/bin` und `/bin` zu finden.

## Aufruf mit relativem Pfad:

```
hermann@debian:~$ ../../usr/bin/ls  
Bilder Desktop Dokumente Downloads Musik Öffentlich Videos Vorlagen
```

## Aufruf ohne Pfad:

```
hermann@debian:~$ ls  
Bilder Desktop Dokumente Downloads Musik Öffentlich Videos Vorlagen
```

In diesem Fall wird von der Shell zum Auffinden des Kommandos `ls` die Umgebungsvariable `PATH` herangezogen.

# Umgebungsvariable **PATH**

Die Umgebungsvariable **PATH** enthält eine durch Doppelpunkte getrennte Liste von Verzeichnissen, in denen die Shell von links nach rechts nach externen Kommandos sucht, falls diese ohne Pfad-Angabe aufgerufen werden.

```
hermann@debian:~$ echo $PATH  
/usr/local/bin:/usr/bin:/bin:/usr/local/games:/usr/games
```

Wie wir gesehen haben, ist das Kommando `ls` in den Verzeichnissen `/usr/bin` und `/bin` zu finden.

Beim Aufruf von `ls` ohne Pfad-Angabe wird zuerst im Verzeichnis `/usr/local/bin` gesucht, dann in `/usr/bin`, dann in `/bin`, dann in `/usr/local/games` und schließlich in `/usr/games`. Der erste Treffer (von links nach rechts) wird ausgeführt.

Also wird beim Aufruf von `ls` die Programm-Datei `/usr/bin/ls` ausgeführt.



# Kommando `which`

- Das Kommando `which` zeigt den Pfad der Programm-Datei an, die bei einem Aufruf ohne Pfad-Angabe ausgeführt wird. Es durchsucht dazu (wie die Shell) von links nach rechts die Verzeichnisse, die in der Umgebungsvariablen `PATH` enthalten sind.

```
hermann@debian:~$ which ls
/usr/bin/ls
hermann@debian:~$ which wc
/usr/bin/wc
```

```
hermann@debian:~$ which cowsay  
/usr/games/cowsay  
hermann@debian:~$ which fortune  
/usr/games/fortune
```

# PATH-Variable von root

Für den Benutzer **root** gelten in der Regel andere Voreinstellungen für die Umgebungsvariable **PATH**.

```
hermann@debian:~$ echo $PATH
/usr/local/bin:/usr/bin:/bin:/usr/local/games:/usr/games
hermann@debian:~$ sudo -i
root@debian:~# echo $PATH
/usr/local/sbin:/usr/local/bin:/usr/sbin:/usr/bin:/sbin:/bin
root@debian:~# exit
Abgemeldet
```

- Der root-PATH enthält zusätzlich die Verzeichnisse, in denen sich Systemverwaltungs-Kommandos befinden können:  
`/usr/local/sbin`, `/usr/sbin` und `/sbin`.
- Die Verzeichnisse `/usr/local/games` und `/usr/games` sind im root-PATH hingegen nicht enthalten.
- Die Verzeichnisse `/usr/local/sbin`, `/usr/sbin` und `/sbin` sind im Benutzer-PATH nicht enthalten. Deswegen werden viele Systemverwaltungs-Kommandos von normalen Benutzern nicht gefunden, z.B. `shutdown`, `reboot`, `useradd` etc.

```
hermann@debian:~$ which shutdown
hermann@debian:~$ sudo which shutdown
/usr/sbin/shutdown
```

```
hermann@debian:~$ which useradd
hermann@debian:~$ sudo which useradd
/usr/sbin/useradd
```

- Andere Systemverwaltungs-Kommandos wie `apt`, `mount` etc. sind hingegen auch für normale Benutzer im PATH enthalten.
- Dies ist meist dann der Fall, wenn der Benutzer die Kommandos zur Anzeige von Informationen nutzen kann. Der Superuser kann diese Kommandos auch für administrative Eingriffe ins System nutzen.
- Sollen Kommandos vom Superuser und von normalen Benutzern verwendet werden, dann sind sie in Verzeichnissen abgelegt, die im root-PATH und im Benutzer-PATH enthalten sind: `/usr/bin` und `/bin`.

```
hermann@debian:~$ which apt
/usr/bin/apt
hermann@debian:~$ sudo which apt
/usr/bin/apt
```

```
hermann@debian:~$ apt list --upgradable
...
hermann@debian:~$ apt update
Paketlisten werden gelesen... Fertig
E: Sperrdatei /var/lib/apt/lists/lock konnte nicht geöffnet werden. - open (13: Keine Berechtigung)
...
hermann@debian:~$ sudo apt update
OK:1 http://deb.debian.org/debian bookworm InRelease
Holen:2 http://security.debian.org/debian-security bookworm-security InRelease [48,0 kB]
...
```

# Anpassung der **PATH**-Variablen

Die Umgebungsvariable **PATH** kann auch vom Benutzer an seine eigenen Bedürfnisse angepasst werden.

Dies behandeln wir jedoch an anderer Stelle.