* **Linux (Bash):**
  + - Wie bei Windows, werden Unterordner markiert. Hier werden diese mit einem **„/“** markiert. (Beispiel: „/home/cindy/Desktop“ zum Vergleich, bei Windows würde es so aussehen: „C:\Benutzer\cindy\Desktop“)
    - **Ordner:**
    - /bin = Darin werden wichtige Binärdateien oder Programme gespeichert.
    - /etc = Hier befinden sich wichtige Systemkonfigurationsdateien.
    - /home = Ein persönliches Verzeichnis für Nutzer. (Dokumente, Fotos, etc.)
    - /proc = Informationen zu aktuell laufenden Prozessen.
    - /user = Enthält die vom Nutzer installierten Softwares.
    - /var = Darin speichern wir Systemprotokolle und alle Dateien, die ständig verändert werden.
    - Wie bei Windows kann man sich mit einem Flag weitere Befehlsoptionen angeben lassen.
    - Man kann auch mehrere Flags gleichzeitig benutzen (z.B. „ls –l –a/“), die Reihenfolge der Flags, bestimmt die Reihenfolge der Ergebnisse.
  + **Linux: Ändern von Verzeichnissen in Bash:**
    - Die PowerShell-Befehle funktionieren hier genauso.
    - Auch hier gibt es die Vervollständigung mit der TAB-Taste, nur, dass hier die möglichen Dateinamen und Verzeichnisse alle gleichzeitig angezeigt werden und nicht nach einander.
  + **Linux: Erstellen Sie Verzeichnisse in Bash:**
    - Ebenfalls mit dem Befehl „mkdir“
    - Statt Leerzeichen schreibt man „\“
    - Also: mkdir Projekt\ Trainee
    - Oder auch: mkdir „Projekt Trainee“
  + **Linux: Befehlsverlauf:**
    - Gleichen Befehle wie in Windows
  + **Linux: Kopieren von Dateien und Verzeichnissen:**
    - Wie bei Windows
    - Ordner mit Inhalt kopieren: cp –r „Name“ ~/(Zielort)
  + **Linux: Verschieben und Umbenennen von Dateien, Verzeichnissen:**
    - Wie bei Windows
    - Mv red\_document.txt blue\_documents.txt
    - Mv \*\_document.txt ~/(Zielordner)
  + **Linux: Entfernen von Dateien und Verzeichnissen:**
    - Wie bei Windows
    - Rm Dateiname
    - Bei wichtigen Dateien, haben nur autorisierte Nutzer haben die Berechtigung zu löschen.
    - Wenn ich ein ganzes Verzeichnis löschen möchte, fragt Bash nach, ob man sich sicher ist. Oder man benutzt **„-R**“, dann weiß Bash, dass man sich dem bewusst ist und fragt nicht mehr nach.
  + **Notizen:**
    - > = Umleitungsoperator. Mit ihm lässt sich die Standartausgabe weiterleiten.
    - < = Umleitungsoperator. Mit ihm lässt sich die Standarteingabe ändern.
    - | = Pipe-Operator. Die Ausgabe eines Befehls an die Eingabe eines anderen Befehls senden. (Ausgabeumleitung)
    - Stderr = I/O-Umleiter. (Input/Output-Umleiter).
    - Alle Ausgabestreams werden nummeriert:

1. Stdout – für Standartausgabe (the output)
2. Stderr – Standartfehler/ Fehlermeldungen (the error)
   * + Mit 2> werden Standartfehler umgeleitet.
     + Wie bei Windows mit „$null“, kann bei Linux mit „/dev/null“ eine Fehlermeldung ins nichts sozusagen ins schwarze Loch umgeleitet werden.
     + „less/ var/log/syslog 2> /dev/null“

* **Übersicht Befehle für Linux:**

|  |  |
| --- | --- |
| Befehl | Bedeutung |
| Ls / | Dateien und Verzeichnisse auflisten |
| „—help“ Flag | Weitere Informationen zum Befehl |
| „man (Befehl)“ | Man pages (Handbuch) hier sieht man noch mehr Details zum Befehl. |
| Ls –L / | Das L steht für Long. Hier sieht man mehr Informationen zum Verzeichnis und den darin gespeicherten Dateien und Ordnern. Man sieht: (v. links n. rechts) Dateiberechtigungen, Anzahl der Verknüpfungen der Datei, Dateieigentümer und Dateigruppen, Dateigröße, das Datum der letzten Änderung und den Datei-oder Verzeichnisnamen. |
| „-a“ oder „--all“ | Damit werden auch die versteckten Dateien angezeigt |
| Pwd oder print working directory | Gibt an, in welchem Verzeichnis man sich gerade befindet |
| Cd oder change directory | Wenn man ein anderes Verzeichnis aufrufen möchte |
| Cd .. | Man geht in das übergeordnete Verzeichnis. (z.B. von „C:\User\Sarah\Dokumente“ in „C:\Unser\Sarah“) |
| Cd ~ | Verweis auf Homeverzeichnisses. |
| Mkdir | Neues Verzeichnis erstellen |
| History | Zeigt mir eine Liste von den zuvor eingegebenen Befehlen an |
| STRG + R oder # | Den Verlauf Ihrer bisher verwendeten Befehle durchsuchen. |
| Clear | Damit wird nicht der Verlauf, sondern nur die Bildschirmausgabe gelöscht. |
| CP oder Copie Paste | Kopieren |
| \* | Irgendwas. |
| -R | Führt den Verzeichnisinhalt auf |
| Mv | Zum Verschieben oder umbenennen meiner Verzeichnisse oder Dateien |
| Rm oder remove | Löschen von Dateien oder Verzeichnissen |
| Cat | Inhalt einer Datei anzeigen lassen (unkomfortabel) |
| Less | „Less Dateiname“ Ähnlich wie „more“ bei Windows, aber „less“ bietet mehr Funktionen. Mit Pfeiltasten scrollen, mit „g“ kommt man an den Anfang der Datei, „G“ kommt man ans Ende der Datei, /wortsuche, „q“ verlassen von „less“ |
| More | Wie bei Windows (wird aber bei Bash, kaum noch genutzt weil „less“ angenehmer ist) |
| Head | Standartmäßig sieht man die ersten 10 Zeilen einer Datei. (head Dateiname) |
| Tail | Zeigt die letzten 10 Zeilen einer Datei an. (tail Dateiname) |
| Nano | Öffnet den Editor Nano. (nano Dateiname) |
| Grep | Sucht nach Wörtern in Dateien oder Verzeichnissen (gleiche wie sls in PowerShell) |
| Echo | Wie bei Windows |
| Sudo (Super User DO) | Damit erhält man die Rechte eines Rootusers (Superuser) |
| Su (Substitute User) | Damit wechselt man zu einem anderen User |
| Exit | Ausloggen vom Nutzer |
| Passwd | Passwort ändern |
| -e (expire) | Einen anderen Nutzer dazu bringen, sein Passwort zu ändern |
| Useradd | Hinzufügen eines Nutzers |
| Userdel | User entfernen |
| Chmod (change mode) | Berechtigungen ändern |
| Dpkg | Ein Debian-Paket installieren (dpkg –i) |
| -i | Install |
| 7z –e (extract) | Extrahieren einer Datei mit 7-Zip |
| Sudo apt install | Paket installieren |
| Sudo apt remove | Paket deinstallieren |
| Apt update | Die Liste der Pakete in den Repositories wird aktualisiert |
| Apt upgrade | Alle Pakete werden automatisch aktualisiert |
| Uname | Liefert Systeminformationen |
| -r (release) | Welche Kernel Version (uname –r) |
| Sudo apt full-upgrade | Aktualisiert auch die Kernel Version (Betriebssystem) |
| Parted –L | Zeigt an mit welchen Laufwerken der Computer verbunden ist |
| Sudo parted /dec/sdb | Man kommt ins Parted Tool und kann dort mehr Befehle eingeben |
| Print | Festplattenstatus ansehen |
| Mklabel | Festplattenbezeichnung festlegen |
| Mkpart (Name, FS, Anfangspunkt MiB, Endpunkt GiB) | Festplatte in Partitions unterteilen |
| Mkfs (sudo mkfs –t ext4 /dev/sdb1) | Partition mit den FS formatieren |
| Quit | Verlassen des Parted Tools |
| Sudo mount | Festplatte mounten (Windows macht dies automatisch) |
| Umount | FS unmounten |
| Sudo blkid | UUID unserer Geräte aufrufen |
| Ln –s (s=softlinks) | Softlink erstellen |
| Ln | Hardlink erstellen |
| Fsck (file system check) | Dateisystem manuell reparieren |
| Ps –x | Man sieht eine Liste aller auf dem System ausgeführten Prozesse. (l. n. r. > **PID** = Prozess-ID; **TTY** =das mit dem Prozess verknüpfte Terminal; **STA**=Prozessstatus(R=running, T=stopped, S=interruptible sleep); **TIME**=die gesamte CPU-Zeit, die ein Prozess bisher in Anspruch genommen hat; **COMMAND**=Name des Befehls, welcher ausgeführt wird.) |
| Ps –ef | „e“= werden alle Prozesse abgerufen, auch die von anderen Nutzern. „f“ zeigt die gesamten Details zu einem Prozess. (l. n. r. **UID**=Nutzer-ID der Person, die den Prozess gestartet hat; **PID**=Prozess-ID; **PPID**=die ID des bergeordneten Prozessen; **C**=Anzahl der diesem Prozess untergeordneten Prozesse; **STIME**=ist die Startzeit des Prozesses; **TTY**=das mit dem Prozess verknüpfte Terminal; **TIME**=die gesamte CPU-Zeit, die ein Prozess bisher in Anspruch genommen hat; **CMD**=Name des ausgeführten Befehls.) |
| Ps –ef I grep | Ob ein Prozess ausgeführt wird |
| SIG (SIGINT) | Signal senden. Abbruch. STRG +C |
| Kill + PID | Prozess beenden |
| SIGKILL | Prozess stirbt. Der Prozess wird komplett beendet, es bleibt keine Zeit um Dateien zu bereinigen |
| Kill –kill PID | Notlösung um ein Prozess zu beenden |
| SIGTSTP | Terminal stop. Ein Prozess wird pausiert. Kill –Tstp |
| SIGCONT | Fortsetzen des Prozesses. Kill -Cont |
| SIGTERM |  |
| Top | Der Verbrach von Systemressourcen in Echtzeit |
| Uptime | Infos zur aktuellen Uhrzeit, Wie lange läuft ihr System bereits, wie viele Nutzer sind angemeldet, welche durchschnittliche Last wirkt auf ihren Computer |
| Lsof | Prozessverwaltung |
| /var/log | Hier werden Protokolle gespeichert |
| /var/log/syslog | Protokoll mit nahezu allen Ereignissen in Ihrem System |
| /var/log/syslog | grep (Wortsuche) |  |