



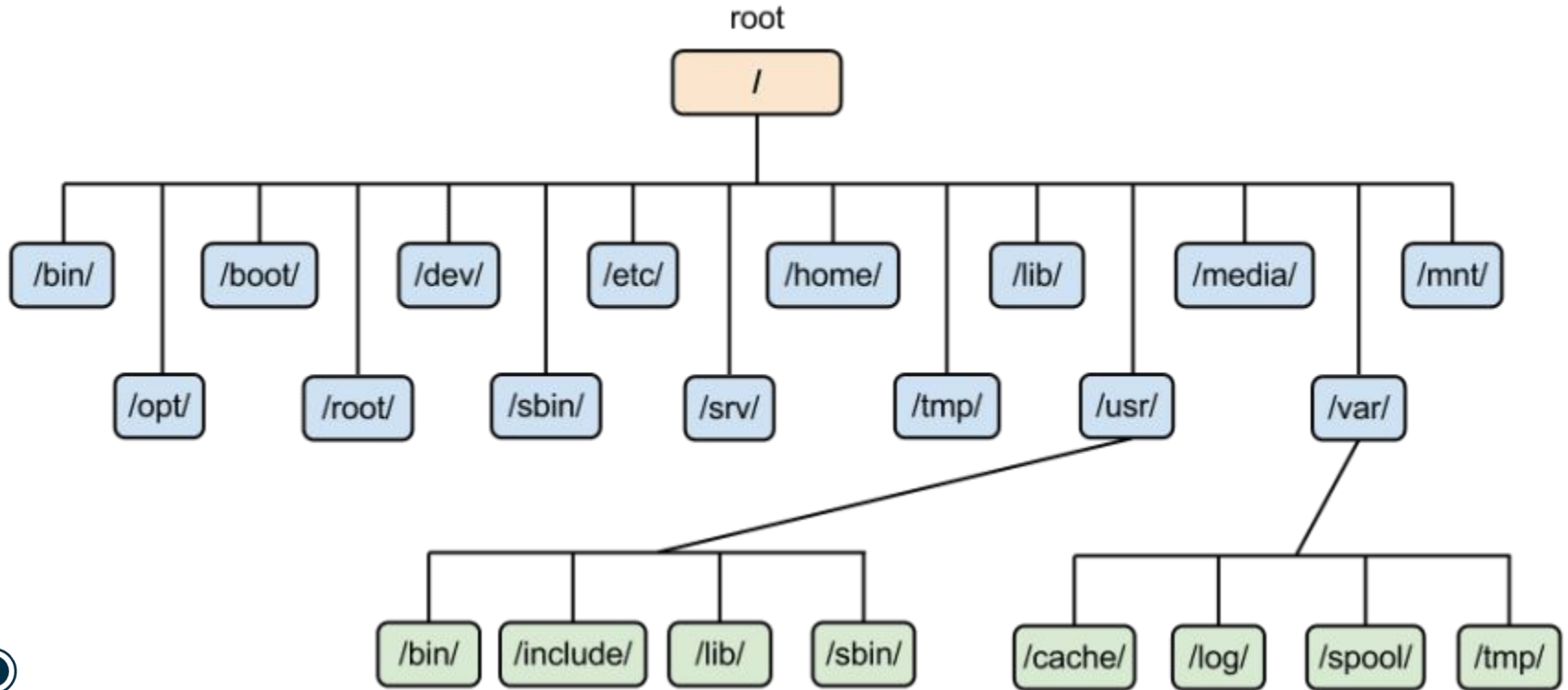
**TECH
STARTER**

Linux-
Tag 2

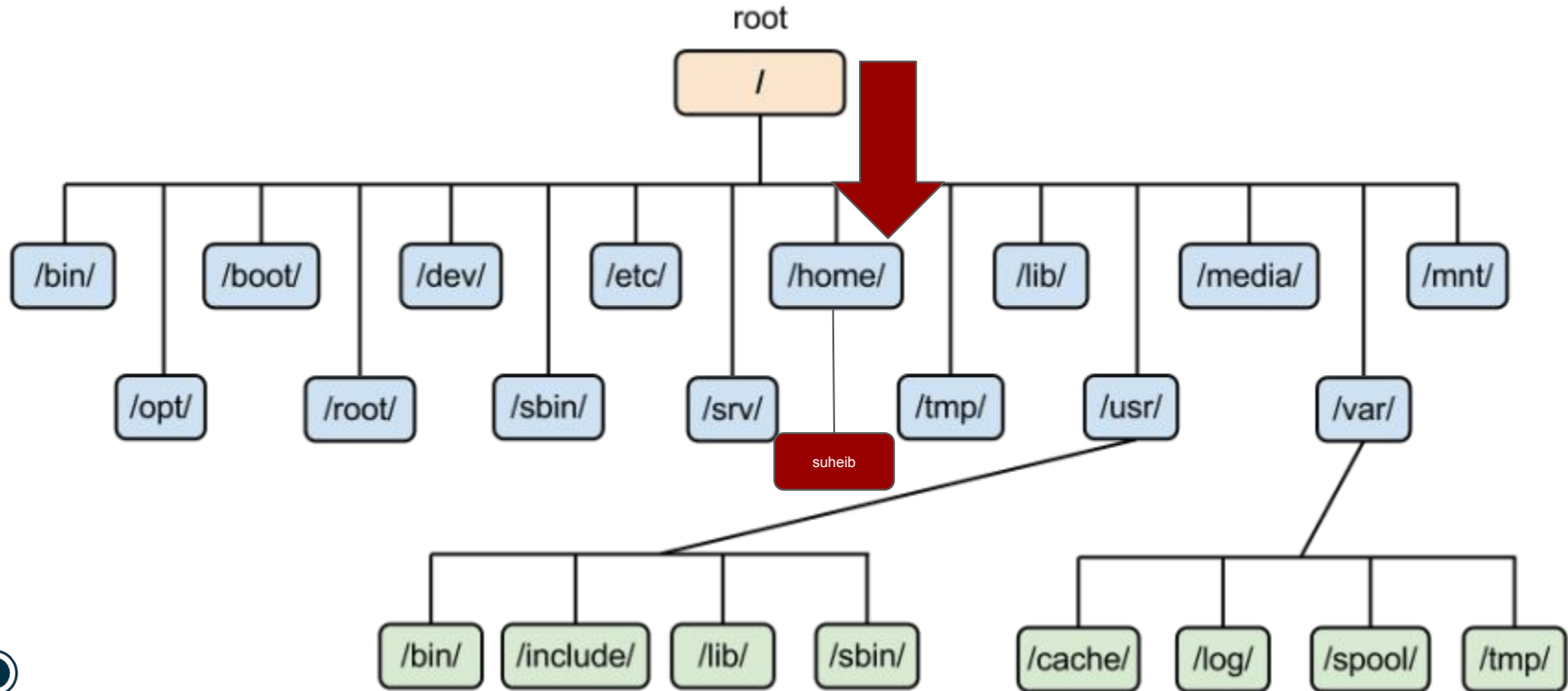
Lernziele

- Grundbefehle in Linux ausführen können
- Linux Dateisystem Aufbau grundlegend kennen
- Absoluter vs. Relativer Pfad

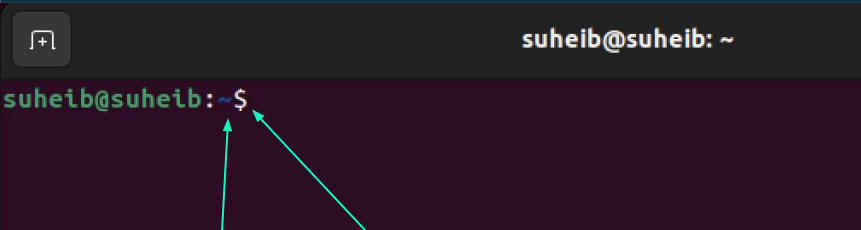
Linux Dateisystem



Wo befinden sich das Home-Verzeichnis von dem Nutzer suheib?



Home-Verzeichnis von suheib



```
suheib@suheib: ~  
suheib@suheib:~$
```

Wir sind als Nutzer
suheib angemeldet

Homeverzeichnis:
/home/suheib

\$ bedeutet keine root Berechtigung
würde root Berechtigung bedeuten (root engl. für Wurzel)

~ entspricht dem
Home-Verzeichnis für den
angemeldeten Nutzer

Absoluter vs. relativer Pfad

Absoluter Pfad

- Pfad beginnt am Anfang der Wurzel (root)
Startet mit /
- **/home/suheib** ist der absolute Pfad für das Home-Verzeichnis von dem Benutzer tom

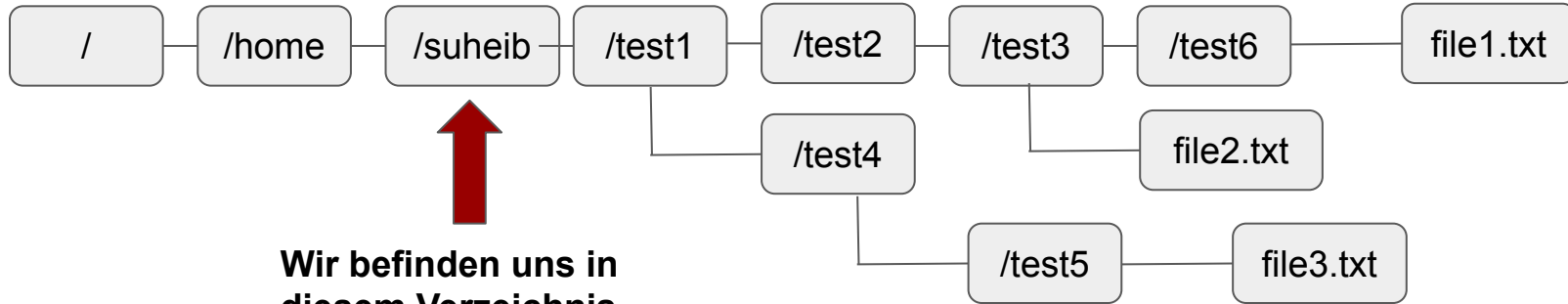
→

Relativer Pfad

- Ausgangspunkt ist der Pfad, indem sich der Nutzer aktuell befindet

Beispiel 1: Absoluter vs. relativer Pfad

Folgender Ausschnitt ist gegeben:



Wir befinden uns in diesem Verzeichnis

file3.txt:

Absolut: `/home/suheib/test1/test4/test5/file3.txt`

Relativ: `./test1/test4/test5/file3.txt`

file2.txt:

Absolut: `/home/suheib/test1/test2/test3/file2.txt`

Relativ: `./test1/test2/test3/file2.txt`

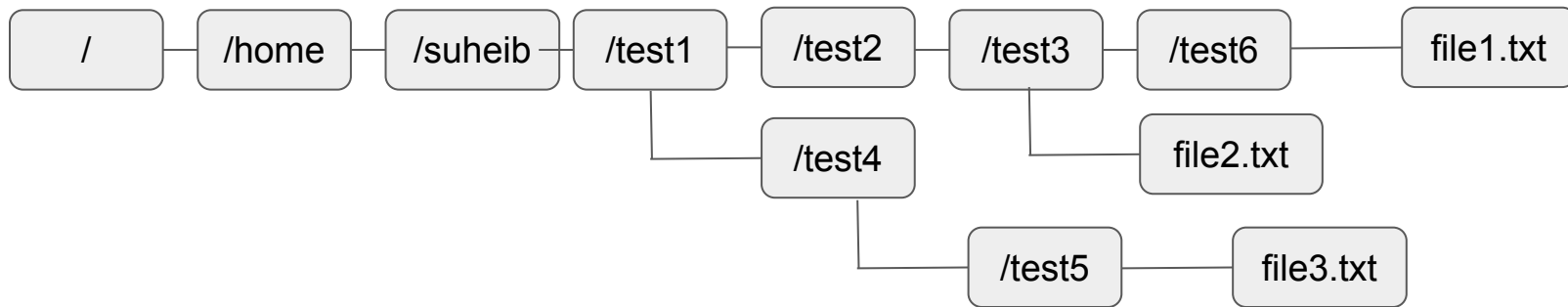
file1.txt:

Absolut: `/home/suheib/test1/test2/test3/test6/file1.txt`

Relativ: `./test1/test2/test3/test6/file1.txt`

Beispiel 2: Absoluter vs. relativer Pfad

Folgender Ausschnitt ist gegeben:



file3.txt:

Absolut: /home/suheib/test1/test4/test5/file3.txt

Relativ: ./file3.txt

**Wir befinden uns in
diesem Verzeichnis**

file1.txt:

Absolut: /home/suheib/test1/test2/test3/test6/file1.txt

Relativ: ../../test2/test3/test6/file1.txt

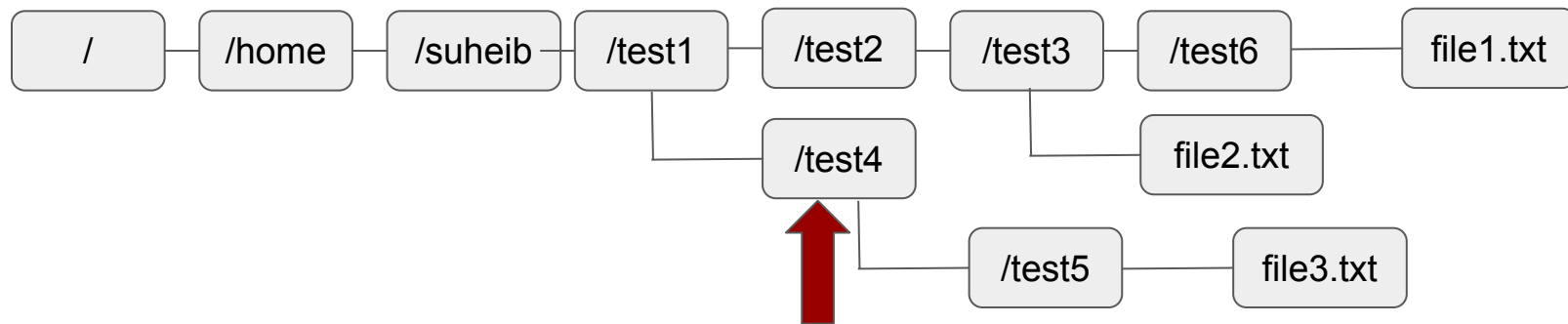
file2.txt:

Absolut: /home/suheib/test1/test2/test3/file2.txt

Relativ: ../../test2/test3/file2.txt

Beispiel 2: Absoluter vs. relativer Pfad

Folgender Ausschnitt ist gegeben:



file3.txt:

Absolut: `/home/suheib/test1/test4/test5/file3.txt`

Relativ: `./test5/file3.txt`

**Wir befinden uns in
diesem Verzeichnis**

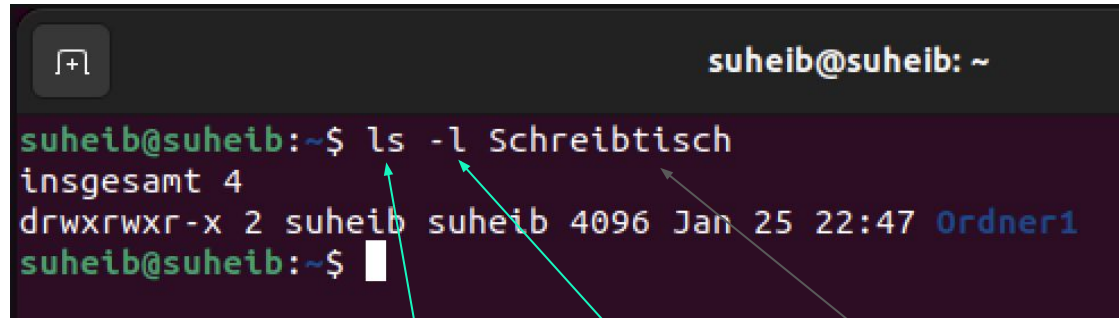
Befehle



**TECH
STARTER**

Befehle-Grundlagen am Beispiel von “ls”

- Befehle werden bei Konvention klein geschrieben
- ls steht für list (engl. für auflisten)



```
suheib@suheib:~$ ls -l Schreibtisch
insgesamt 4
drwxrwxr-x 2 suheib suheib 4096 Jan 25 22:47 Ordner1
suheib@suheib:~$
```

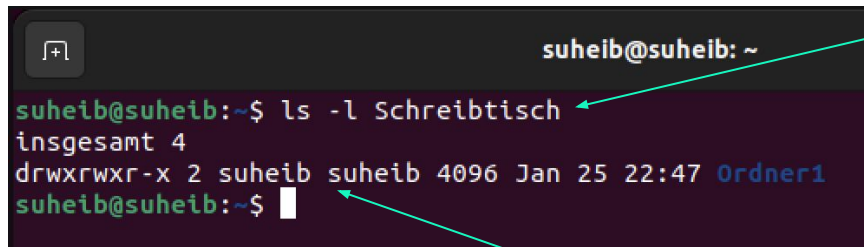
Befehl

Option

Argument
(Der Inhalt des
Ordnerns, der
aufgelistet
werden soll)

Mehr zur Bash-Shell

Bash ist die standard Shell von Linux

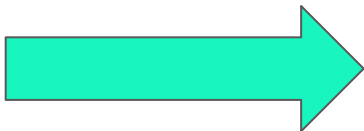


```
suheib@suheib: ~  
suheib@suheib:~$ ls -l Schreibtisch  
insgesamt 4  
drwxrwxr-x 2 suheib suheib 4096 Jan 25 22:47 Ordner1  
suheib@suheib:~$
```

The screenshot shows a terminal window with the prompt 'suheib@suheib: ~'. The user enters the command 'ls -l Schreibtisch'. The output shows 'insgesamt 4' followed by a directory listing for 'Ordner1'. A red arrow points from the first step of the list to the command, and another red arrow points from the second step to the output.

1. **Parsing:** Shell interpretiert den Befehl und trennt ihn in Teile auf (Befehl, Option, Argument)
2. **Befehlsaufruf:** sucht nach ausführbaren Dateien, die über die \$PATH Variable registriert wurden (Mehr dazu kommt) → Gib **echo \$PATH** ein
3. **Ausführung:** Übergibt die Teile an den ausführbaren Befehl

1. Ausgabe

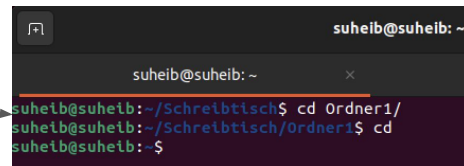


Bash Shell übergibt den Befehl an den Kernel, welcher die erforderlichen Operationen durchführt (Zuweisen von Ressourcen)

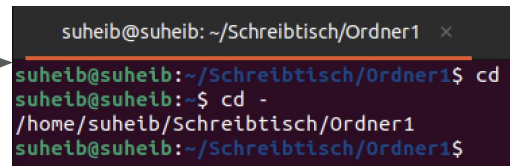
Befehl - cd

Bedeutung: change directory (engl. für Verzeichnis wechseln)

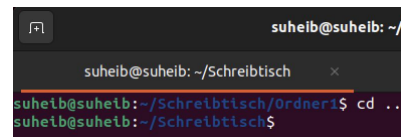
- **cd** ohne Argument bringt dich ins home-Verzeichnis des Nutzers → Am Beispiel von suheib ist das absolut **/home/suheib**
- **cd -** bringt dich in das zuvor aufgerufene Verzeichnis
- **cd [Pfad]** nimmt einen **absoluten** oder **relativen** Pfad als Argument
- **cd ..** bringt dich in den übergeordneten Ordner



```
suheib@suheib: ~  
suheib@suheib: ~/Schreibtisch$ cd Ordner1/  
suheib@suheib: ~/Schreibtisch/Ordner1$ cd  
suheib@suheib: ~$
```



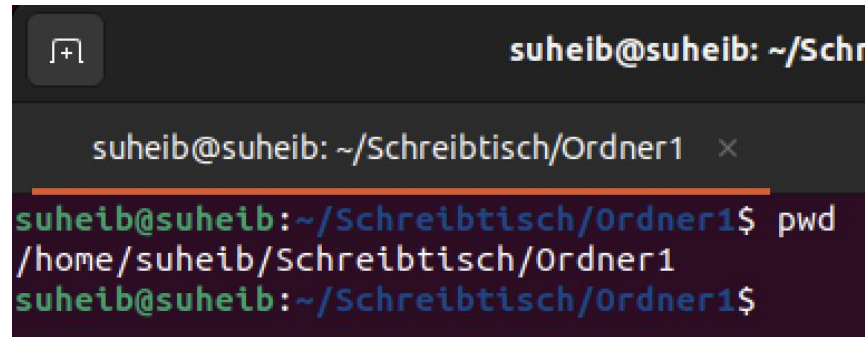
```
suheib@suheib: ~/Schreibtisch/Ordner1  
suheib@suheib: ~/Schreibtisch/Ordner1$ cd  
suheib@suheib: ~$ cd -  
/home/suheib/Schreibtisch/Ordner1  
suheib@suheib: ~/Schreibtisch/Ordner1$
```



```
suheib@suheib: ~/Schreibtisch  
suheib@suheib: ~/Schreibtisch$ cd ..  
suheib@suheib: ~$
```

Befehl - pwd

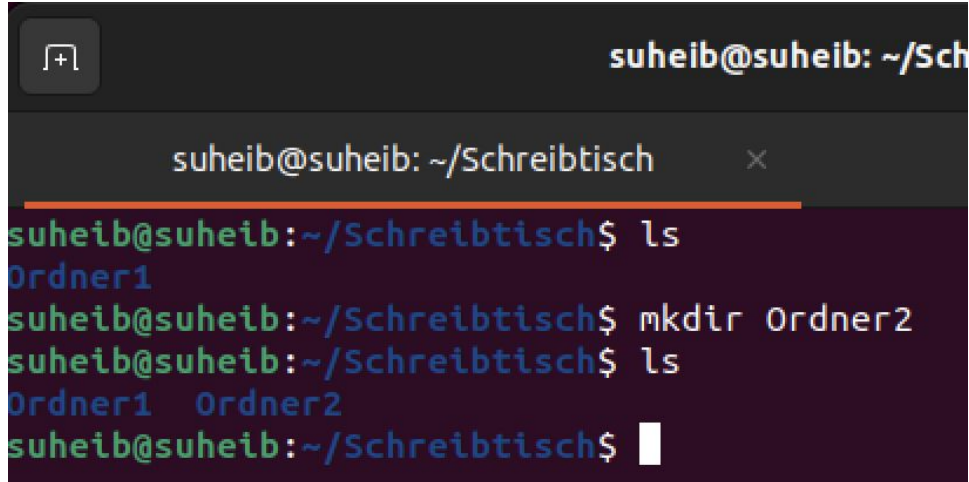
Bedeutung: print working directory zeigt das aktuelle Verzeichnis an (**absoluter** Pfad)

A terminal window with a dark background. The title bar shows 'suheib@suheib: ~/Schre'. The terminal content shows the user 'suheib@suheib' in the directory '~/Schreibtisch/Ordner1' running the 'pwd' command. The output is '/home/suheib/Schreibtisch/Ordner1'.

```
suheib@suheib: ~/Schre  
suheib@suheib: ~/Schreibtisch/Ordner1 x  
suheib@suheib:~/Schreibtisch/Ordner1$ pwd  
/home/suheib/Schreibtisch/Ordner1  
suheib@suheib:~/Schreibtisch/Ordner1$
```

Befehl - mkdir

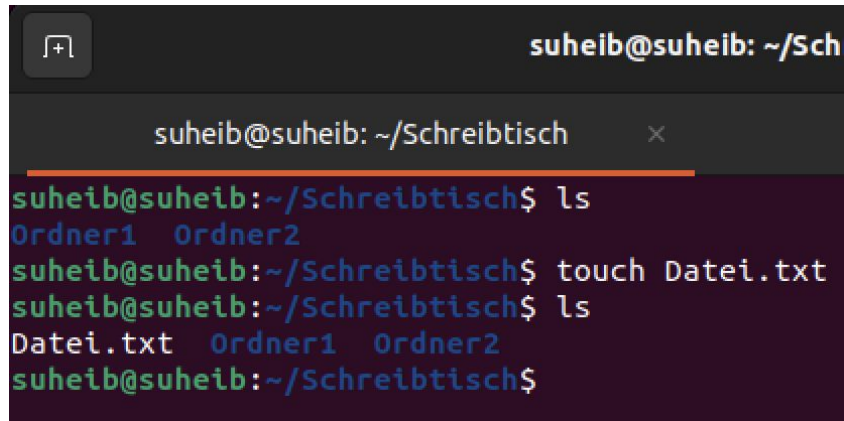
Bedeutung: make directory (engl. für erstelle ein Verzeichnis)



```
suheib@suheib: ~/Schreib  
suheib@suheib: ~/Schreibtisch  
suheib@suheib:~/Schreibtisch$ ls  
Ordner1  
suheib@suheib:~/Schreibtisch$ mkdir Ordner2  
suheib@suheib:~/Schreibtisch$ ls  
Ordner1  Ordner2  
suheib@suheib:~/Schreibtisch$
```

Befehl - touch

Bedeutung: legt eine neue Datei an



```
suheib@suheib: ~/Sch  
suheib@suheib: ~/Schreibtisch  
suheib@suheib:~/Schreibtisch$ ls  
Ordner1  Ordner2  
suheib@suheib:~/Schreibtisch$ touch Datei.txt  
suheib@suheib:~/Schreibtisch$ ls  
Datei.txt  Ordner1  Ordner2  
suheib@suheib:~/Schreibtisch$
```


Mehr Optionen findet ihr

- **man** [Befehl]
- [Befehl] **--help**

Geschafft! :)

Zeit für eine Demo

Weitere Befehle

- **rm** [Option]... [Argument]... → Löschen
- **cat** [Option]... [Argument]... → Auslesen
- **mv** [Option]... [Argument]... → Umbenennen/Verschieben
- **who** → Wer ist angemeldet
- **whoami** → Als welcher Nutzer bin ich gerade aktiv
- **clear** → Macht die CLI frei
- **date** → Gibt das Datum aus
- **which** [Befehl] → Zeigt wo der Befehl abgespeichert ist
- **history** → Zeigt eine Historie der aufgerufenen Befehle an
- ...