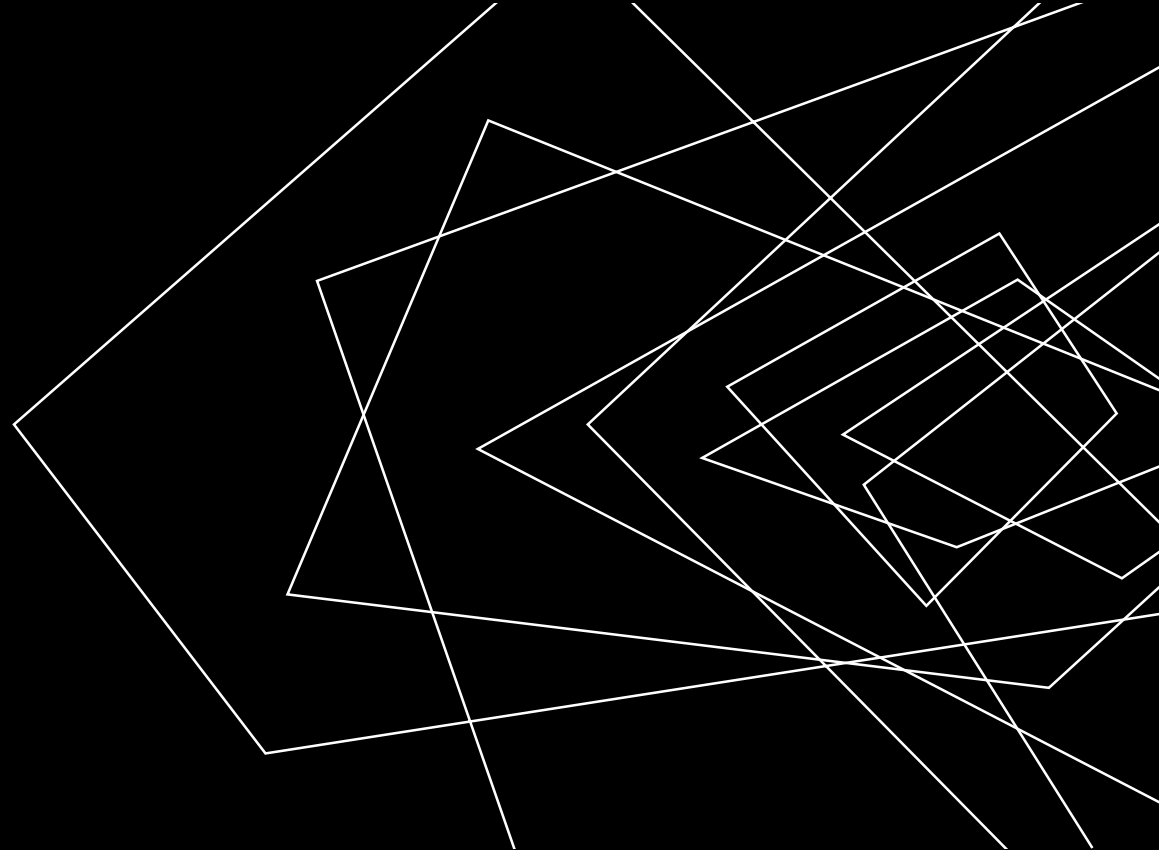
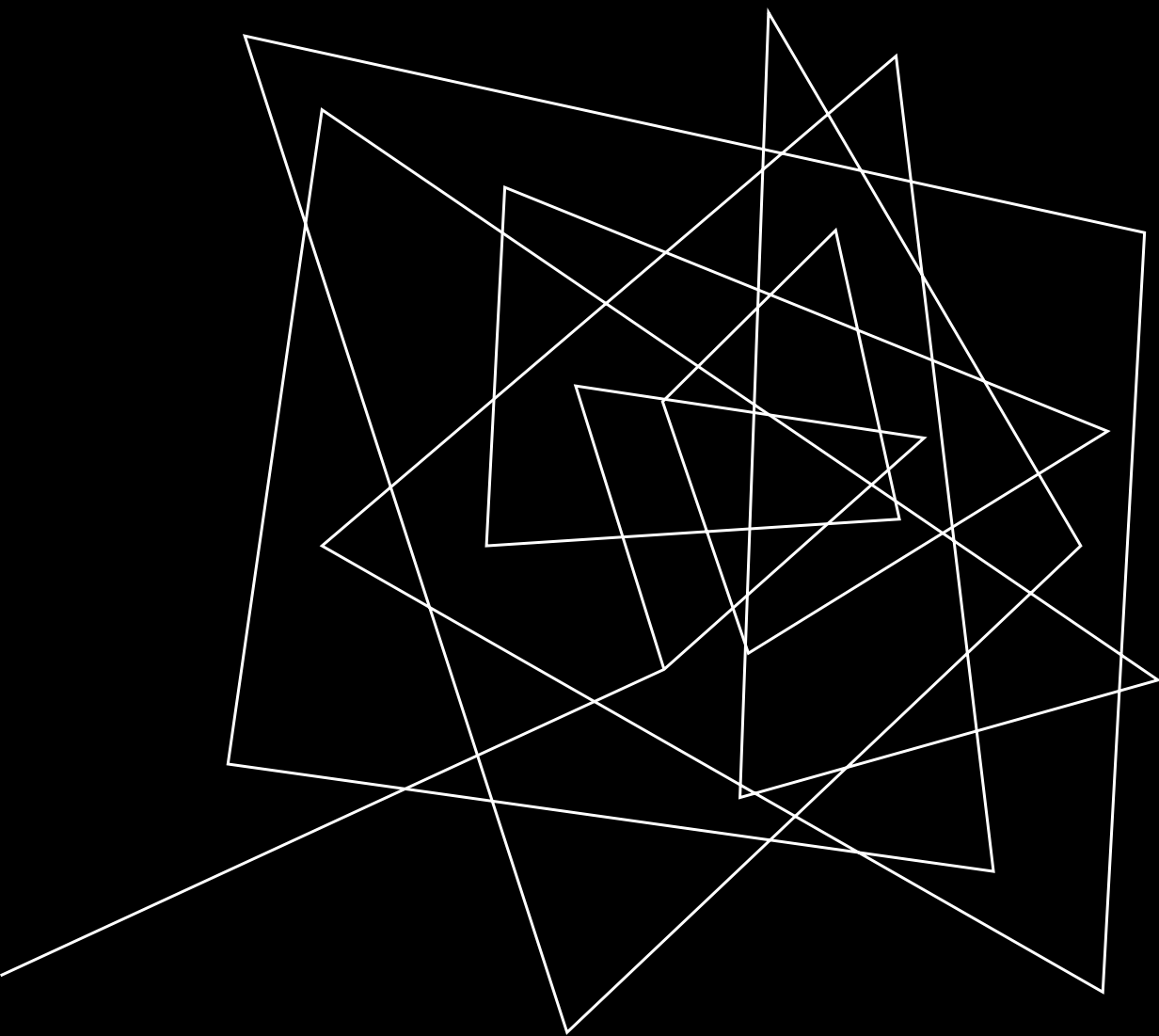


LINUX - TAG 1

# AGENDA

1. Einführung in Linux
2. Installation und Setup von Linux
3. Einführung in die Kommandozeile
4. Navigation im Dateisystem





# 1. EINFÜHRUNG IN LINUX

Was ist Linux? Geschichte, Distributionen, Linux vs.  
andere Betriebssysteme

# 1. EINFÜHRUNG IN LINUX

- 1.1. Was ist Linux?
- 1.2. Geschichte von Linux
- 1.3. Distributionen
- 1.4. Linux vs. Andere Betriebssysteme
- 1.5. Vorteile
- 1.6. Anwendungen

## 1.1. WAS IST LINUX?

- Definition: Linux ist ein freies, Open-Source-Betriebssystem, das auf dem Linux-Kernel basiert.
- Kernel: Der zentrale Bestandteil des Betriebssystems, der Hardware und Software verbindet.
- GNU/Linux: Oft wird Linux zusammen mit GNU-Software verwendet, daher der Name GNU/Linux.

## 1.2. GESCHICHTE VON LINUX

- **1991: Linus Torvalds veröffentlicht die erste Version von Linux (0.01).**
- **1992: Linux wird als Open Source veröffentlicht.**
- **1994: Veröffentlichung von Linux 1.0.**
- **2000er: Wachstum und Verbreitung in Server- und Desktop-Märkten.**

## 1.3. LINUX-DISTRIBUTIONEN

- **Was ist eine Distribution?:**

Verschiedene Versionen von Linux, die auf dem Kernel basieren, aber unterschiedliche Softwarepakete und Management-Tools enthalten.

- **Beispiele:**

- Ubuntu: Benutzerfreundlich, weit verbreitet.
- Debian: Stabil, große Software-Repositories.
- Fedora: Neueste Technologien, von Red Hat unterstützt.
- Arch Linux: Minimalistisch, für erfahrene Benutzer.

## 1.4.1. LINUX VS. ANDERE BETRIEBSSYSTEME

- **Windows:**

- **Lizenzierung:** Proprietär vs. Open Source.
- **Benutzeroberfläche:** Windows GUI vs. verschiedene Linux GUIs (z.B., GNOME, KDE).
- **Sicherheit:** Linux gilt als sicherer wegen seiner Open-Source- Natur und Community-Prüfungen.



## 1.4.2. LINUX VS. ANDERE BETRIEBSSYSTEME

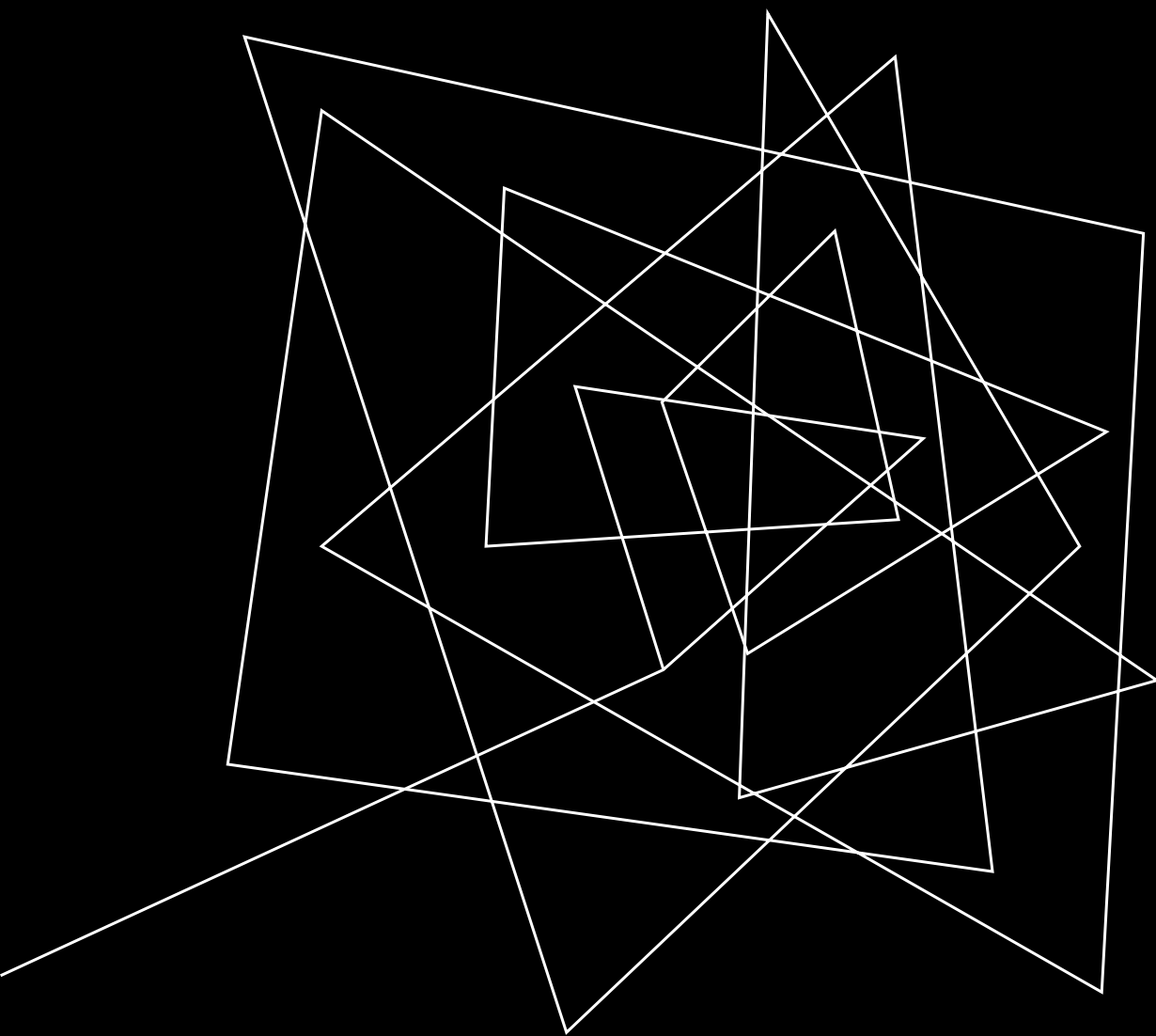
- **macOS:**

- **Basis:** macOS basiert auf Unix, wie Linux.
- **Kompatibilität:** Unterschiede in Software und Hardware-Support.
- **Ähnlichkeiten:** Terminal-Zugriff, UNIX-Befehle.

## 1.4.3. LINUX VS. ANDERE BETRIEBSSYSTEME

- **Unix:**

- **Verwandtschaft:** Linux ist von Unix inspiriert.
- **Unterschiede:** Linux ist Open Source, viele Unix-Systeme sind proprietär.



## 2. INSTALLATION UND SETUP VON LINUX

Installation einer Linux-Distribution (z.B. Ubuntu) in  
einer virtuellen Maschine oder auf einem Rechner,  
Grundlegende Konfiguration

## 2. INSTALLATION UND SETUP VON LINUX

- 2.1. Einführung
- 2.2. Voraussetzungen
- 2.3. Vorbereitung
- 2.4. Installation in einer virtuellen Maschine
- 2.5. Installation auf einem physischen Rechner
- 2.6. Grundlegende Konfiguration
- 2.7. Erste Schritte nach der Installation

## 2.1. EINFÜHRUNG

Installation einer Linux-Distribution (z.B. Ubuntu) in einer virtuellen Maschine oder auf einem Rechner,  
Grundlegende Konfiguration

## 2.2. VORAUSSETZUNGEN

- **Hardware:**

- **Für virtuelle Maschinen: Mindestens 2 GB RAM, 20 GB freien Speicher.**
- **Für physischen Rechner: Empfohlene Mindestanforderungen sind höher.**

- **Software:**

- **Virtualisierungssoftware (z.B., VirtualBox, VMware) für virtuelle Maschinen.**
- **Ubuntu ISO-Datei von der offiziellen Webseite.**

## 2.3. VORBEREITUNG

- **Download:** Ubuntu ISO-Datei von [ubuntu.com](http://ubuntu.com).
- **Virtualisierungssoftware:** Installation und Konfiguration von VirtualBox oder VMware.
  - Download und Installation der Software.
  - Erstellen einer neuen virtuellen Maschine.

## 2.4.1. INSTALLATION IN EINER VIRTUELLEN MASCHINE

### **1. Erstellen der virtuellen Maschine:**

- Auswahl von „Neues VM erstellen“.
- Zuweisen von RAM und Festplattenspeicher.
- Auswahl des ISO-Images als Boot-Medium.



## 2.4.2. INSTALLATION IN EINER VIRTUELLEN MASCHINE

### **2. Installation von Ubuntu:**

- Starten der virtuellen Maschine.
- Auswahl der Installationssprache.
- Auswahl von „Ubuntu installieren“.
- Festlegen der Installationsoptionen  
(z.B., normale Installation, Updates herunterladen).

## 2.4.3. INSTALLATION IN EINER VIRTUELLEN MASCHINE

### **3. Partitionierung:**

- Automatische Partitionierung wählen  
oder manuell konfigurieren (für fortgeschrittene Benutzer).

## 2.4.4. INSTALLATION IN EINER VIRTUELLEN MASCHINE

### **4. Benutzerkonfiguration:**

- Eingabe von Benutzername und Passwort.
- Einrichtung der Zeitzone und Tastaturlayouts.

## 2.5.1. INSTALLATION AUF EINEM PHYSISCHEN RECHNER

### **1. Booten vom Installationsmedium:**

- Erstellen eines bootfähigen USB-Sticks oder DVDs mit der Ubuntu ISO-Datei.
- Booten vom Installationsmedium.

## 2.5.2. INSTALLATION AUF EINEM PHYSISCHEN RECHNER

### **2. Installation von Ubuntu:**

- Auswahl der Installationssprache.
- Auswahl von „Ubuntu installieren“.
- Festlegen der Installationsoptionen (z.B., parallele Installation neben Windows).

## 2.5.3. INSTALLATION AUF EINEM PHYSISCHEN RECHNER

### **3. Partitionierung:**

- Automatische Partitionierung oder manuelle Anpassung (bei Dual-Boot-Systemen).

## 2.5.4. INSTALLATION AUF EINEM PHYSISCHEN RECHNER

### **4. Benutzerkonfiguration:**

- Eingabe von Benutzername und Passwort.
- Einrichtung der Zeitzone und Tastaturlayouts.

## 2.6. GRUNDLEGENDE KONFIGURATION

- **System-Updates:**

- Durchführung von System-Updates nach der Installation  
(`sudo apt update && sudo apt upgrade`).

- **Installieren von Software:**

- Verwendung von `apt-get` oder `snap` zum Installieren zusätzlicher Software.

- **Netzwerk-Konfiguration:**

- Überprüfung der Netzwerkeinstellungen.
- Verbindung zu WLAN oder Ethernet konfigurieren.

- **Benutzerverwaltung:**

- Hinzufügen neuer Benutzer (`sudo adduser benutzername`).
- Verwalten von Benutzerrechten.



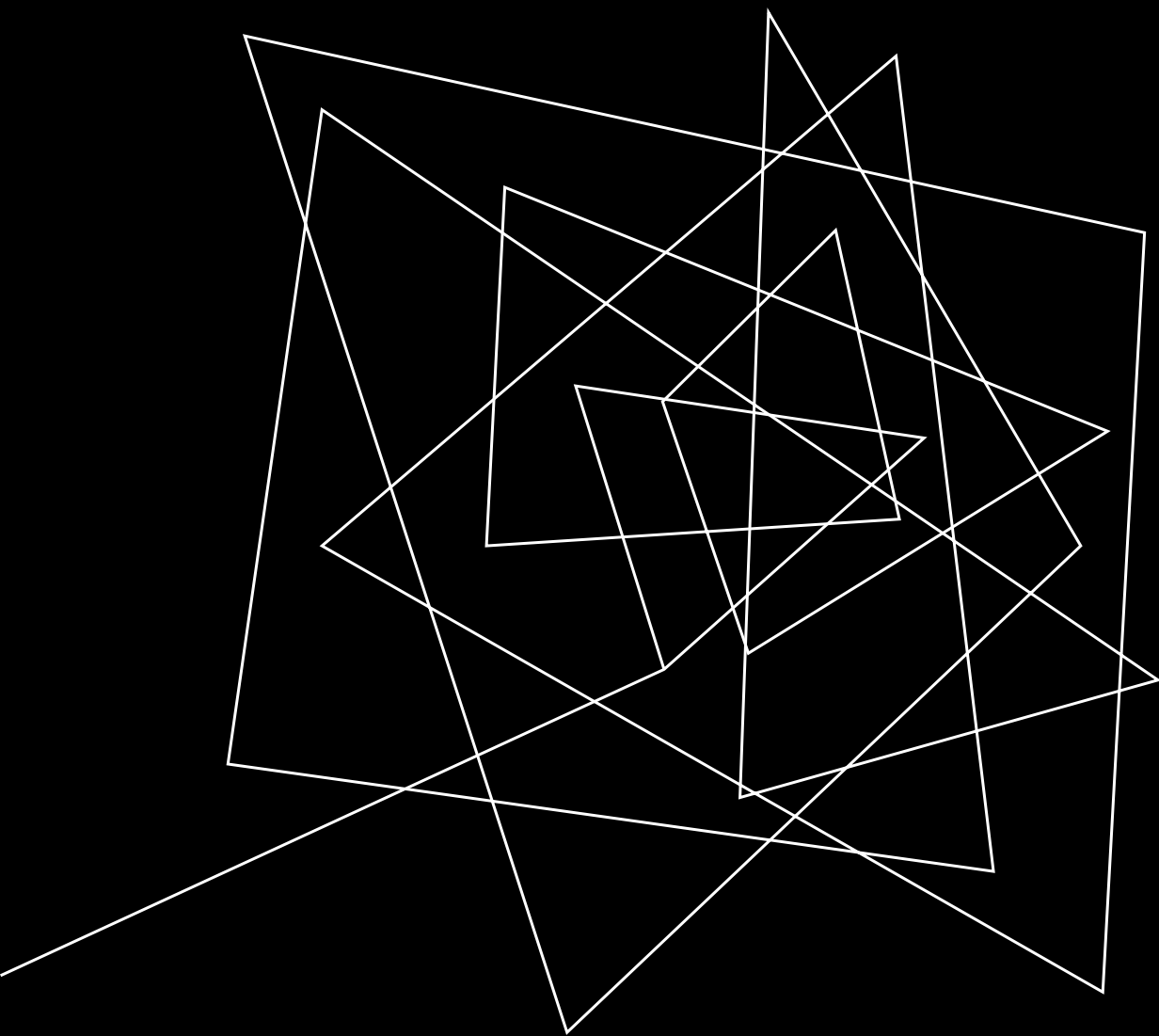
## 2.7. ERSTE SCHRITTE NACH DER INSTALLATION

- **Systemüberprüfung:**

- Überprüfen der Hardware-Komponenten (z.B., Drucker, Grafikkarten).
- Anpassen der Systemeinstellungen nach Bedarf.

- **Systempflege:**

- Regelmäßige Updates und Wartung.
- Backup-Strategien und System-Wiederherstellung.



# 3. EINFÜHRUNG IN DIE KOMMANDOZEILE

Grundlagen der Shell, wichtige Befehle,  
Verzeichnisstruktur

# 3. EINFÜHRUNG IN DIE KOMMANDOZEILE

- 3.1. Einführung in die Shell
- 3.2. Grundlegende Shell-Befehle
- 3.3. Verzeichnisstruktur
- 3.4. Tipps für die Arbeit mit der Kommandozeile
- 3.5. Zusammenfassung

# 3.1. EINFÜHRUNG IN DIE SHELL

- **Was ist eine Shell?**

- Ein Programm, das Benutzereingaben (Befehle) interpretiert und an das Betriebssystem weitergibt.
- Beispiele: Bash (Bourne Again Shell), Zsh, Fish.

- **Shell vs. Terminal**

- Shell: Die eigentliche Befehlszeilenumgebung.
- Terminal: Das Programm, das die Shell ausführt (z.B., GNOME Terminal, Konsole).

## 3.2.1. GRUNDLEGENDE SHELL-BEFEHLE

---

- **Navigation:**

- **pwd** – Zeigt das aktuelle Verzeichnis an.
- **cd** [Verzeichnis] – Wechselt in das angegebene Verzeichnis.

- **Verzeichnisinhalt anzeigen:**

- **ls** – Listet Dateien und Verzeichnisse im aktuellen Verzeichnis auf.
- **ls -l** – Zeigt detaillierte Informationen (Dateigröße, Berechtigungen).

## 3.2.2. GRUNDLEGENDE SHELL-BEFEHLE

---

- **Datei- und Verzeichnismanagement:**

- **cp [Quelle] [Ziel]** – Kopiert Dateien oder Verzeichnisse.
- **mv [Quelle] [Ziel]** – Verschiebt oder benennt Dateien oder Verzeichnisse um.
- **rm [Datei]** – Löscht eine Datei.
- **rm -r [Verzeichnis]** – Löscht ein Verzeichnis und seinen Inhalt rekursiv.

- **Erstellung von Dateien und Verzeichnissen:**

- **touch [Dateiname]** – Erstellt eine neue Datei oder ändert das Änderungsdatum einer Datei.
- **mkdir [Verzeichnisname]** – Erstellt ein neues Verzeichnis.
- **rmdir [Verzeichnisname]** – Löscht ein leeres Verzeichnis.

## 3.3. VERZEICHNISSTRUKTUR

- **Wurzelverzeichnis (/):**

- Das oberste Verzeichnis im Dateisystem.

- **Wichtige Verzeichnisse:**

- **/home** – Benutzerverzeichnisse.
- **/etc** – Systemkonfigurationsdateien.
- **/var** – Variablen Daten (z.B., Logdateien).
- **/usr** – Benutzerspezifische Programme und Daten.
- **/tmp** – Temporäre Dateien.

- **Verzeichnisstruktur verstehen:**

- Hierarchische Struktur.
- Jeder Pfad im System wird relativ zur Wurzel definiert.

## 3.4. TIPPS FÜR DIE ARBEIT MIT DER KOMMANDOZEILE

- **Verwendung von Wildcards:**

- **\*** – Platzhalter für eine beliebige Anzahl von Zeichen (z.B., \*.txt für alle Textdateien).
- **?** – Platzhalter für ein einzelnes Zeichen (z.B., file?.txt).

- **Pipes und Redirection:**

- **|** – Leitet die Ausgabe eines Befehls als Eingabe für einen anderen Befehl weiter.
- **>** – Leitet die Ausgabe eines Befehls in eine Datei um, überschreibt den Inhalt.
- **>>** – Fügt die Ausgabe eines Befehls an eine Datei an.

- **Befehl-Historie:**

- **history** – Zeigt eine Liste der zuletzt ausgeführten Befehle.
- **!n** – Wiederholt den Befehl an Position n in der Historie.



## 3.5. ZUSAMMENFASSUNG

---

- **Shell Grundlagen:** Die Shell interpretiert Befehle und ermöglicht die Interaktion mit dem System.
- **Wichtige Befehle:** Navigation, Datei- und Verzeichnismanagement.
- **Verzeichnisstruktur:** Verständnis der hierarchischen Struktur des Dateisystems.



## 4. NAVIGATION IM DATEISYSTEM

Befehle wie ls, cd, pwd, mkdir, rmdir, Umgang mit  
Pfadern

# 4. NAVIGATION IM DATEISYSTEM

- 4.1. Einführung in die Dateisystem-Navigation
- 4.2. pwd – Aktuelles Verzeichnis anzeigen
- 4.3. cd – Verzeichnis wechseln
- 4.4. ls – Verzeichnisinhalt anzeigen
- 4.5. mkdir – Verzeichnis erstellen
- 4.6. rmdir – Verzeichnis löschen
- 4.7. Umgang mit Pfaden
- 4.8. Tipps und Tricks

## 4.1. EINFÜHRUNG IN DIE DATEISYSTEM-NAVIGATION

---

- **Ziel:** Verstehen der grundlegenden Befehle zur Navigation und Verwaltung von Dateien und Verzeichnissen.
- **Dateisystem-Hierarchie:** Wie das Dateisystem strukturiert ist und wie man sich darin bewegt.

## 4.2. PWD – AKTUELLES VERZEICHNIS ANZEIGEN

---

- **Befehl:** `pwd` (print working directory)
- **Verwendung:** Zeigt den vollständigen Pfad des aktuellen Verzeichnisses an.
- **Beispiel:**

```
$ pwd  
/home/benutzername/dokumente
```

## 4.3. CD – VERZEICHNIS WECHSELN

- **Befehl:** `cd` [Verzeichnis]
- **Verwendung:** Wechselt in das angegebene Verzeichnis.
- **Beispiele:**

`cd /home/benutzername`

– Wechselt zum Benutzerverzeichnis.

`cd ..`

– Geht ein Verzeichnis nach oben.

`cd ~`

– Geht zum Heimatverzeichnis des Benutzers.

## 4.4. LS – VERZEICHNISINHALT ANZEIGEN

- **Befehl:** `ls`
- **Verwendung:** Listet Dateien und Verzeichnisse im aktuellen Verzeichnis auf.
- **Optionen:**
  - **`ls -l`** – Detaillierte Liste (zeigt Berechtigungen, Eigentümer, Dateigröße).
  - **`ls -a`** – Zeigt auch versteckte Dateien (die mit `.` beginnen).
  - **`ls -lh`** – Detaillierte Liste mit menschenlesbaren Größen.
- **Beispiel:**

```
$ ls -l
drwxr-xr-x 2 benutzername benutzername 4096 Sep 10 09:00 beispiel
-rw-r--r-- 1 benutzername benutzername 1234 Sep 10 09:00 datei.txt
```

## 4.5. MKDIR – VERZEICHNIS ERSTELLEN

- **Befehl:** `mkdir [Verzeichnisname]`
- **Verwendung:** Erstellt ein neues Verzeichnis.
- **Beispiele:**

`mkdir neue_dateien` – Erstellt ein Verzeichnis namens „neue\_dateien“.

`mkdir -p ordner1/ordner2` – Erstellt mehrere Verzeichnisse, einschließlich verschachtelter Verzeichnisse, wenn sie noch nicht existieren.



## 4.6. RMDIR – VERZEICHNIS LÖSCHEN

- **Befehl:** `rmdir [Verzeichnisname]`
- **Verwendung:** Löscht ein leeres Verzeichnis.
- **Beispiel:**

`rmdir alte_dateien` – Löscht das Verzeichnis „alte\_dateien“, wenn es leer ist.

**Hinweis:** Um nicht leere Verzeichnisse zu löschen, verwenden Sie `rm -r [Verzeichnisname]`.

## 4.7. UMGANG MIT PFADEN

- **Relative vs. Absolute Pfade:**

- **Absolute Pfade:** Beginnen vom Wurzelverzeichnis /.

**Beispiel:** /home/benutzername/dokumente

- **Relative Pfade:** Relativ zum aktuellen Verzeichnis.

**Beispiel:** ../dokumente (geht ein Verzeichnis nach oben und dann in „dokumente“).

- **Spezielle Verzeichnisse:**

- . – Das aktuelle Verzeichnis.
- .. – Das übergeordnete Verzeichnis.

- **Beispiele:**

- cd /var/log – Wechselt in das Verzeichnis /var/log
- cd ../bilder – Geht ein Verzeichnis nach oben und dann in „bilder“.

## 4.8. TIPPS UND TRICKS

---

- **Tab-Vervollständigung:** Drücken Sie Tab, um den Dateinamen oder Verzeichnisnamen automatisch zu vervollständigen.
- **Verzeichnisse durchsuchen:** Verwenden Sie `ls -R`, um Inhalte von Verzeichnissen rekursiv anzuzeigen.

## 4.9. ZUSAMMENFASSUNG

---

- **Befehle:** `pwd`, `cd`, `ls`, `mkdir`, `rmdir`.
- **Pfadarten:** Absolute und relative Pfade.
- **Verzeichnisverwaltung:** Erstellen, Auflisten und Löschen von Verzeichnissen.

# ZUSAMMENFASSUNG

## **Installation von Linux:**

Einfach in einer virtuellen Maschine oder auf einem physischen Rechner

## **Grundlegende Konfiguration:**

Wichtige Schritte für einen funktionalen und sicheren Betrieb.



# DANKE

Costis Aivalis

[costis.Aivalis@gmail.com](mailto:costis.Aivalis@gmail.com)

[aivalis.eu](http://aivalis.eu)