Commands:

ls - Liste Dateien und Verzeichnisse.

cd - Wechseln Sie das Verzeichnis.

pwd - Zeigt das aktuelle Verzeichnis an.

touch - Erstellen Sie eine leere Datei.

mkdir - Erstellen Sie ein neues Verzeichnis.

cp - Kopieren Sie Dateien und Verzeichnisse.

mv - Verschieben oder Umbenennen von Dateien und Verzeichnissen.

rm - Löschen von Dateien und Verzeichnissen.

cat - Zeige den Inhalt einer Datei an.

less - Anzeigen des Inhalts einer Datei, Seite für Seite.

head - Zeige die ersten Zeilen einer Datei an.

tail - Zeige die letzten Zeilen einer Datei an.

grep - Suche nach einem Muster in Dateien.

find - Durchsuchen Sie das Dateisystem nach Dateien und Verzeichnissen.

chmod - Ändern Sie die Berechtigungen einer Datei oder eines Verzeichnisses.

chown - Ändern Sie den Besitzer einer Datei oder eines Verzeichnisses.

chgrp - Ändern Sie die Gruppe einer Datei oder eines Verzeichnisses.

ps - Zeige laufende Prozesse an.

kill - Beenden Sie einen laufenden Prozess.

top - Zeigt systemweite Ressourcen und laufende Prozesse an.

wget - Laden Sie Dateien aus dem Internet herunter.

curl - Führen Sie HTTP-Anfragen aus der Befehlszeile aus.

tar - Erstellen oder extrahieren Sie tar-Archive.

unzip - Entpacken Sie ZIP-Archive.

ssh - Sichere Verbindung zum Remote-Server.

scp - Kopieren Sie Dateien über eine sichere Verbindung auf einen Remote-Server.

ping - Prüfen Sie die Erreichbarkeit eines Netzwerkgeräts.

ifconfig - Zeigt Netzwerkschnittstelleninformationen an.

netstat - Anzeigen von Netzwerkstatistiken und -verbindungen.

du - Zeigen Sie die Speichernutzung von Dateien und Verzeichnissen an.

df - Zeige den verfügbaren Festplattenspeicher an.

uname - Zeige Informationen über das Betriebssystem an.

man - Zeigt Handbuchseiten an.

history - Zeigt die Befehlshistorie an.

alias - Erstellt eine benutzerdefinierte Befehlsverknüpfung.

export - Legt Umgebungsvariablen fest.

source - Führt Skripts in der aktuellen Shell aus.

sed - Stream Editor zum Bearbeiten von Text.

awk - Textverarbeitungswerkzeug für Datenstromsätze.

sort - Sortieren Sie Zeilen in einer Datei.

cut - Extrahieren Sie Abschnitte aus Dateien.

grep - Durchsuchen Sie Dateien nach einem Muster.

diff - Vergleichen Sie den Inhalt von Dateien.

wc - Zählen Sie Wörter, Zeilen oder Zeichen in einer Datei.

tee - Schreibt den Eingang in eine Datei und druckt ihn auf dem Bildschirm.

xargs - Führen Sie Befehle mit Argumenten aus einer Eingabe aus.

bg - Führen Sie einen Prozess im Hintergrund aus.

fg - Bringt einen Prozess in den Vordergrund.

su - Wechseln Sie zum Superuser oder zu einem anderen Benutzer.

exit - Beenden Sie die aktuelle Shell.

df -h: Zeigt Informationen zur Festplattennutzung an.

du -sh: Zeigt die Größe eines Verzeichnisses an.

find . -name "\*.txt": Sucht nach allen Dateien mit der Erweiterung ".txt" im aktuellen Verzeichnis und seinen Unterverzeichnissen.

grep -r "Suchbegriff" .: Sucht nach dem angegebenen Suchbegriff in allen Dateien im aktuellen Verzeichnis und seinen Unterverzeichnissen.

chmod +x dateiname.sh: Ändert die Berechtigungen einer Datei, um sie ausführbar zu machen.

tar -czvf dateiname.tar.gz verzeichnis: Erstellt ein komprimiertes tar.gz-Archiv eines Verzeichnisses.

history n: Zeigt die letzten n ausgeführten Befehle an.

ssh benutzername@hostname: Stellt eine SSH-Verbindung zu einem Remote-Server her.

sudo befehl: Führt einen Befehl mit Root-Berechtigungen aus.

alias name='befehl': Erstellt eine benutzerdefinierte Befehlsverknüpfung mit dem angegebenen Namen.

ln -s quelle ziel: Erstellt eine symbolische Verknüpfung (Symlink) von der Quelldatei zum Ziel.

wc -l dateiname: Zeigt die Anzahl der Zeilen in einer Datei an.

echo "Text": Gibt den angegebenen Text auf dem Bildschirm aus.

shutdown -h now: Fährt den Computer herunter.

uname -a: Zeigt umfassende Informationen über das Betriebssystem an.

whoami: Zeigt den aktuellen Benutzernamen an.

cut -d" " -f1 dateiname: Schneidet den Inhalt einer Datei anhand eines Trennzeichens ab und gibt nur das erste Feld aus.

sed 's/suchbegriff/ersetzung/g' dateiname: Ersetzt alle Vorkommen des Suchbegriffs durch die angegebene Ersetzung in einer Datei.

tail -f dateiname: Zeigt den Inhalt einer Datei in Echtzeit an.

ps aux: Zeigt eine Liste aller laufenden Prozesse an.

uname -r: Zeigt die aktuelle Linux-Kernel-Version an.

export variable=wert: Legt eine Umgebungsvariable mit dem angegebenen Wert fest.

diff datei1 datei2: Zeigt die Unterschiede zwischen zwei Dateien an.

history -c: Löscht die gesamte Befehlshistorie.

sleep n: Hält die Ausführung des Skripts oder der Befehlszeile für n Sekunden an.

Bash

Bourne again shell

>

If the specified file already exists, it will be overwritten with the new output.

>>

If the specified file already exists, the new output will be added to the end of the file without overwriting the existing content.

**Cronjob**:

\* \* \* \* \* befehl

- - - - -

| | | | |

| | | | +----- Tag der Woche (0 - 7, wobei 0 und 7 Sonntag sind)

| | | +------- Monat (1 - 12)

| | +--------- Tag im Monat (1 - 31)

| +----------- Stunde (0 - 23)

+------------- Minute (0 - 59)

Beispiel:

\*/5 \* \* \* \* /pfad/zum/deinem/skript.sh

/5 macht = in 5 minuten

5 = 5 um 5 uhr

shbang erste zeille immer mit script sprache

z.b. bash

!/bin/bash

# = kommentar

Mkdir make directory

Cd alleine geht zurück ins home

Cd mit einem directory wie /home zum start dann mit

ls die subfolder anschauen und so weiter fahren aber ohne /

Ip a zeigt alle infos der IP an

Logout

mv datei bewegen oder datei unbennenen

chgrp change group

chmod change user

Cal = calendar

Echo $(date) = zeigt datum mit tag jahr etc.

$(date "+%A,%d,%m,%Y") =datum

df -h zeigt Ordner grössen

Lsblk zeigt system speicher ssd

mkdir -p /home/student1/direcotryname create antoehr directory in another directory

rwx, rwx, rwx

Root, Gruppe, Benutzer

Zur überprüfung ob der letzte befehl erfolgreich war

echo $?

SSH:

Mann muss in den Netzwerk einstellung von der Virtuellen Maschine in die Port

Forwarding einstellungen als Port 22 nehmen weil das der SSH ist. Host IP sollte

man eine Virtualbox host only ip nehmen als client IP die Ip die, die virtuelle

maschine schon hatt und dan einfach mit cmd sich per ssh verbidnen.

**ssh -l user Ip**

SCP:

Secure Copy Protection

Von ssh zu anderen CMD Datei kopieren mit:

**scp student@171.20.10.25:/home/student/benicio.txt beni.txt**

**scp username@remote\_host:/path/to/remote/file.txt /path/to/local/destination**

Mit

Numeric comparisons

if [ "$a" -eq "$b" ]; then

# Code to execute if $a is equal to $b

fi

if [ "$a" -ne "$b" ]; then

# Code to execute if $a is not equal to $b

fi

if [ "$a" -lt "$b" ]; then

# Code to execute if $a is less than $b

fi

if [ "$a" -le "$b" ]; then

# Code to execute if $a is less than or equal to $b

fi

if [ "$a" -gt "$b" ]; then

# Code to execute if $a is greater than $b

fi

if [ "$a" -ge "$b" ]; then

# Code to execute if $a is greater than or equal to $b

Fi

String comparisons

if [ "$str1" = "$str2" ]; then

# Code to execute if $str1 is equal to $str2

fi

if [ "$str1" != "$str2" ]; then

# Code to execute if $str1 is not equal to $str2

Fi

File conditions

if [ -e "$file" ]; then

# Code to execute if $file exists

fi

if [ -d "$dir" ]; then

# Code to execute if $dir is a directory

fi

if [ -f "$file" ]; then

# Code to execute if $file is a regular file

fi

if [ -r "$file" ]; then

# Code to execute if $file is readable

fi

if [ -w "$file" ]; then

# Code to execute if $file is writable

fi

if [ -x "$file" ]; then

# Code to execute if $file is executable

fi

**dir beni\*.\***

Suchen

Port nummern:

|  |  |
| --- | --- |
| HTTP | 80 |
| HTTPS | 433 |
| MySQL  | 3306 |
| POP  | 110 |
| SSH | 22 |

Meinevariable="Text"

Echo $Meinevariable

Umgebungsvariablen:

Pfade, Hostname, Username div. Informationen,

~=Tilde

Bash=Bourne shell

\* Mal

+ Plus

- Minus

/ Division

TotalBetrag=$((Anzahl\*Artikelpreis))

Echo=$TotalBetrag

read -p "Bitte eine Zahl eingeben: " lesen

Ende=$((lesen \* lesen))

echo "Das Quadrat von $lesen ist $Ende"

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

*#!/bin/bash*

*#Programm: idk.sh*

*#Beschreibung:*

*#Aufruf: ./idk.sh*

*#Optionen:*

*#Parameter:*

*#Autor: Benicio von Felten*

Logical operators

if [ "$condition1" -a "$condition2" ]; then

# Code to execute if both condition1 and

condition2 are true

fi

if [ "$condition1" -o "$condition2" ]; then

# Code to execute if either condition1 or

condition2 is true

fi

if [ ! "$condition" ]; then

# Code to execute if condition is false

Fi

Combining conditions

if [ "$a" -eq "$b" ] && [ -f "$file" ]; then

# Code to execute if $a is equal to $b and $file

exists

fi

if [ "$str1" = "$str2" ] || [ -d "$dir" ]; then

# Code to execute if $str1 is equal to $str2 or

$dir is a directory

fi

*#Version: 1.02*

*#Datum: 09.01.2024*

*#Aenderung:*

*#Aenderungsdatum:*

*if [ $# -eq 0 ]; then*

 *echo "Bitte Zahl angeben."*

 *exit 1*

*fi*

*lesen=$1*

*lesenn=$2*

*Ende=$((lesen \* lesenn))*

*echo $Ende*

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

Script editoren:

**Vi**

CTRL+C

:q

If not possible

:q!

touch {foo,bar}.txt # Create multiple files

touch test{1..3} # Create test1, test2 and test3 files

touch test{a..c} # Create testa, testb and testc files

cat foo.txt # Print all contents

less foo.txt # Print some contents at a time (g - go to top of file, SHIFT+g, go to bottom of file, /foo to search for 'foo')

head foo.txt # Print top 10 lines of file

tail foo.txt # Print bottom 10 lines of file

open foo.txt # Open file in the default editor

wc foo.txt # List number of lines words and characters in the file

  free: zeigt an wie viel bytes man verbraucht

**Nano**

CTRL+X

**chmod:**

Symbolic Mode: Uses letters and symbols to represent permissions. It's composed of three parts:

Who: Represents the user(s) affected by the change:

u for user/owner

g for group

o for others

a for all (equivalent to ugo)

Operator: Determines how the permissions are modified:

+ to add permissions

- to remove permissions

= to set exact permissions

Permissions: Indicate the specific permissions being modified:

r for read

w for write

x for execute

For example:

To give the user read and write permissions: chmod u+rw file.txt

To remove execute permissions for the group: chmod g-x file.txt

To set read and execute permissions for all: chmod a=rx file.txt

Numeric Mode: Uses numeric values to represent permissions:

4 for read (r)

2 for write (w)

1 for execute (x)

You add these values together to specify the permissions. For example:



chmod 755 file.txt: This gives read, write, and execute permissions to the owner and read/execute permissions to the group and others.

Literal Mode: You can use a mixture of both symbolic and numeric representations. For instance:

chmod u=rw,g=r,o=r file.txt: This sets read and write permissions for the user, read for the group, and read for others.

**chmod o+x ok**

\* = wildcard

2>/dev/null = Papierkorb

echo "Von welcher Zahl möchtest du das Quadrat errechnen?"

read zahl

//Read heisst es nehment den input und tut ihn in die variable zahl

# Quadrat berechnen

quadrat=$((zahl \* zahl))

# Ausgabe

echo "$zahl hoch 2 = $quadrat"

Mail:

