Cheat-Sheet

Core

Command	Description
`cd [path]`	change directory
`ls [path]`	list files and directories
`sudo [command]`	execute command with root permissions
`nano [file]`	open file using nano editor
`vim [file]`	open file using vim editor
`echo [text]`	display line of text
`[Tab]`	autocomplete
`[↑]/[↓]`	go upwards/downwards in history
`[command][↑]/[↓]`	autocomplete based on history
`STRG + C`	cancel process

Help

Command	Description
`man [command]`	shows the help manual for [command]
`[command] -h/--help`	offers help
`whatis [command]`	gives a one line description of [command]
`whereis [command]`	find where a command is located
`whoami`	check for currently logged in user
`history`	show history of commands
`clear`	clear screen

File Management

Command	Description
`pwd`	print working directory
`.`	current directory
`..`	parent directory
`touch [file]`	create new file
`file [file]`	view type of any file
`mkdir [directory]`	make directory
`rm [file]`	remove file
`rmdir [directory]`	remove empty directory
`rm -r [directory]`	remove directory
`cat [file]`	concatenate to screen
`less [file]`	display text file
`cp [file] [new path]`	copy a file
`mv [file] [new path]`	move / rename a file
`./file`	execute file
`chmod [num] [file]`	edit file permissions

Pipes

Command	Description
`[command1] | [command2]`	let output of [command1] serve as input of [command2]
`[command] > [file]`	push output to file
`[command] >> [file]`	append to existing file
`[command] < [file]`	read content from file as input

Other Tools

Command	Description
`ping [URL]/[ip]`	check connection
`wget [URL]`	network downloader
`curl [URL]`	data transfer over various network protocols
`date`	display or set date and time
`diff [file1] [file2]`	compare two files line by line
`df`	display free space of mounted file system
`head -n [count][file]`	display first lines of a file (default: n=10)
`tail -n [count][file]`	display last lines of a file (default: n=10)
`grep [string] [file]`	searches file for string
`wc [file]`	word count

Bash-script-basics

Command	Description
`#!/bin/bash`	first line of a bash script
`x=foo`	define a variable
`$x`	read from a variable
`$((x+1))`	evaluate arithmetic expression
`python -c “code”`	use python script
`read x`	read input and store it in variable
`[[“string1”==”string2”]]`	compare strings
`[[“string1”==*”string2”*]]`	Check if string2 is a substring of string1

`if [condition1]; then`
`  expression1`
`elif [condition2]; then`
`  expression2`
`else`
	if-statement

`   expression3` `fi`	

`for ((i=0;i<n;i++)); do`
`   expression`
`done`
	for-loop

`for i in $(ls); do`
`   expression`
`done`
	loop over every file/directory

`while IFS= read -r line; do`
`   expression`
`done`
	read file input line by line

Vim Basics

Command	Description
`i`	insert mode
`esc`	normal mode
`:!`	force
`:w`	save
`:wq`	save and exit
`:saveas [file]`	save under filename
`:q`	quit without saving
`:[number]`	jump to line [number]
`:help`	open help