Setup

Install Z-shell and the dotfiles

Get the dotfiles from zsh-dotfiles on github and follow its install guide.

- set **zsh** as your default shell
- install the dotfiles
- enable the alt key as meta key

General Note:

Avoid spaces in names of files or directories

It is good practice to avoid spaces ` ` in names of files or directories. Spaces can be used in the command line by *escaping* them with a `\` : e.g. `cd Classes/Media\ Design/Homework`. But this is easy to forget and then cause errors. Because real spaces have different behavior: e.g. `touch Media Design` creates **two** files: one called *Media*, the other called *Design*.

Also spaces in url or file names on websites are likely to break. (Again, spaces can be *escaped* by using `%20`, but this is far from convenient)

Instead of spaces, use underscores (`_`), dashes (`-`), or **C**amel**C**asing.

---

zsh cheat sheet

Navigation

Command	Shortcut	Result
`cd`		**C**hange **D**irectory: go to the directory (needs a *path* as argument)
→|		Use the **tab** key to *autocomplete*
`ls`		**L**i**s**t files in current directory
`ls -l`	`l`	List files with **l**ong format
`ls -al`	`ll`	List **a**ll files in **l**ong format
`pwd`		**P**rint **w**orking **d**irectory: see the current path
`cd \`pwd\``	`.`	Go to the current directory (useful if you're here via a symlink)
`mkdir`		**M**a**k**e **Dir**ectory
`mkdir -p`	`md`	Create Directories up to path and create steps inbetween
`rmdir`	`rd`	**R**e**m**ove **Dir**ectory
`which`		Tells you which command will be executed or where the *executable* is located.

Keyword	Meaning	Example
`~`	Home directory	`cd ~` or `cd ~/Desktop`

`.`	Current directory	
`..`	Parent directory	`cd ..` or `cat ../../../file.txt`
`/`	Root directory	`cd /` or `ls /Volumes`
`?`	Single wildcard character	use `T?m.md` to find *Tim.md*, *Tom.md* or *Tam.md*, but not *TamTam.md*
`*`	Multiple wildcard characters	`ls *.png` to list all .png files in the current directory

Command	is Shortcut for	Result
`cd..`	`cd ..`	Go to the (1 level up) parent directory
`cd...`	`cd ../..`	Go to the (2 levels up) grand-parent directory
`cd....`	`cd ../../..`	Go to the (3 levels up) grand-grand-parent directory
`cd.....`	`cd ../../../..`	Go to the (4 levels up) grand-grand-grand-parent directory
`1`	`cd -`	Go to the previous directory (in your history)
`2`	`cd +2`	Go to 2 directories ago
`3`	`cd +3`	Go to 3 directories ago
`4`	`cd +4`	Go to 4 directories ago
`5`	`cd +5`	Go to 5 directories ago
`6`	`cd +6`	Go to 6 directories ago
`7`	`cd +7`	Go to 7 directories ago
`8`	`cd +8`	Go to 8 directories ago
`9`	`cd +9`	Go to 9 directories ago
`d`	`dirs -v`	Display all the previous directories

File Operations

Command	Meaning	Example
`cp`	**Cop**y or duplicate	`cp file.txt ~/Desktop` or `cp file.txt file2.txt`
`mv`	**Mov**e or rename	`mv file.txt ~/Desktop` or `mv file.txt file2.txt`
`touch`	Create new file or make *dirty*	`touch empty.txt`
`ln`	**Lin**k: make alias	`ln -s ~/Documents/MediaDesign/text-IO/Dirk/ ~/Desktop/textio`
`rm`	**Rem**ove: delete a file	**NOTE**: `rm` hard deletes a file directly! (no way to undo)

Logging in

Command	Meaning	Example
`ssh`	**S**ecure **Sh**ell: log into somewhere remote	`ssh local@domain_of_other_computer.somewhere`
`whoami`	Display as who you are currently logged in	
`logout`	To log out of the current shell	
`sudo`	**Su**peruser **Do**: do as admin	`sudo mkdir /Library/Logs/Test`

Internet

Command	Meaning	Example
`curl`	Print contents from a **url**	`curl www.artez.nl`
	To download a file:	`curl http://www.gutenberg.org/cache/epub/2701/pg2701.txt > mobydick.txt`
`ping`	Ping to a url to check the connection	`ping google.com` (type **^ c** to cancel)

Processes & Inspection

Command	Meaning	Example
`ps`	Show running processes	`ps -ax` Show *all* processes as root
`top`	Show the most consuming processes	This is live. Type `Q` to quit.
`df`	**D**isplay **F**ree disk space	
`du`	Display **d**isk **u**sage	Display all file and their sizes of the current directory recursively
`duh`	is Shortcut for `du -hs`	Display the total **s**ize of the current directory, **h**uman readable
`duh *`	(`du -hs *`)	Display totals sizes of all file and directories
`ifconfig`	See your network interfaces	The type of network your connected to (wifi, bluetooth, etc)
`netstat`	See the live network connections	`netstat -a` List all current connections.

History

Command	is Shortcut for	Meaning
`history`		Display the history of commands
⇡ (arrow up)		Select previous command(s)
⇣ (arrow down)		Select next command(s)
`gh`	`history 1 | grep $@`	**G**rep **H**istory: search history of commands for anything with this word
^ r		(Ctrl + r) Search used commands interactively
⌥ .		(Alt + dot) Complete the command with previously used last argument
`!!`		Repeat previous command
`!` abc		Repeat command with string `abc`
`!` 123		Repeat history command, number `123`

Command Line Navigation

Command	Result
⌥ click	(Alt + mouse click) Set the caret (text cursor) to the position of the mouse
⇠	(arrow left) Move caret left

⇢	(arrow right) Move caret right
⌥ b	(Alt + b) Move **b**ack one word (to the left)
⌥ f	(Alt + f) Move **f**orward one word (to the right)
^ a	(Ctrl + a) Jump to beginning of the line
^ e	(Ctrl + e) Jump to end of the line
^ k	(Ctrl + k) Clear from the current caret position to the end of the line

Printing to the screen

Command	Result	Example
`echo`	Prints the value of the argument	`echo "Hello"` or `echo $HOME`
`cat`	Con**cat**enate. Prints the content of the file(s)	`cat myfile.txt` or `cat file1.txt file2.txt`
`more`	Opens a reader for the file (`Q` to quit)	`more mobydick.txt`
`less`	Like more, but with **more** options	`less mobydick.txt`
`tail`	Prints the end of the file	`tail /var/log/system.log`
`head`	Prints the top of the file	`head mobydick.txt`
`man`	**Man**ual page for ...	try `man cp` or `man man`

Navigation within `more` / `less` / `man`

Command	Result
⇡	Navigate one line up
⇣	Navigate one line down
^ b	Navigate one window **b**ack
^ f	Navigate one window **f**orward
/	Search for a word or a pattern
n	Jump to next occurance
N	(⇧ n) Jump to previous occurance
h	Show the help about `less`
q	Quit

Finding stuff

Command	Example	Result
`find`		Search files from a specified directory
	`find . -name "*.txt"`	*Find all text files from the current directory*
	`find . -type d "m*"`	*Find all directories whose name starts with an "m"*
`locate`	`locate HelveticaNeue`	Search your whole computer to files with the given string in their path
		(`locate` needs first to index your system before you can use it.)

`grep`		Search for a pattern within the contents of files
	`grep 'Ishmael' md.txt`	*Print all occurances (in context) of 'Ishmael' in the file mb.txt*
	`grep -n 'Ishmael' md.txt`	`-n` : *Print the line number in front of the result.*
	`grep -c 'Ishmael' md.txt`	`-c` : *Count the number of occurances*
	`grep -c -i 'whale' md.txt`	`-i` : *Search* case **i**nsensitive*: finds 'whale', 'Whale', or 'WHALE'*

Input / Output (Combine commands and programs)

Keyword	Name	Purpose	Example
`|`	Pipe	Send the output of the 1st command to the 2nd command	`netstat | grep 'adobe'`
`>`	Redirect Output	Send the output of a command to a file	`curl 'www.../1234.png > image.png`
`>>`	Append	Append the output of a command to a file	`who >> users.txt`
`<`	Redirect Input	Use a file as keyboard input into a command	`mail me@myself.com < todo.txt`
`` ` ` ``	Backticks	Use the output of one command as argument of a 2nd	`cd `pwd``

Command	Purpose	Example
`pbcopy`	Send the stdin to the clipboard	`echo "Hello World" | pbcopy`
	(**stdin**: *standard input*)	`cat modydick.txt | grep 'Ishmael' | pbcopy`
`pbpaste`	Send the contents of the clipboard to the stdout	`echo `pbpaste``
	(**stdout**: *standard output*)	`echo `pbpaste``
`open`	Open file with its default application	`open image.png`
`open .`	Opens the current directory in the Finder	

Fun examples

Show all hidden files

`defaults write com.apple.finder AppleShowAllFiles YES`

(Use `NO` instead of `YES` to hide them again.)

The *defaults* command allows you to see and change application and system preferences. It also allows you to change settings for which there is no interface in the Preferences menu.

Star Wars in ASCII-art

`telnet`
`o`

`towel.blinkenlights.nl`
...
`^]`
`quit`

Doktor Eliza

`emacs`
`(ESC)`
`X`
`doctor`
...
`^x^c`

Dunnet Adventure

`emacs`
`(ESC)`
`X`
`doctor`
`look` / `take shovel` / `east` / `dig` / `inventory` ... ...
`^x^c`

For the following you will need **Homebrew**. Install `brew` from the Homebrew website:

Fortune

Install with

`brew install fortune`

Use:

`fortune`

Cowsay

Install with

`brew install cowsay`

Use:

`cowsay "Moo"`

List all possible characters:

`cowsay -l`

Combine with fortune:

`fortune | cowsay`

Make the computer talk

`say "Good morning"`

Use a different *voice* with the `-v` argument:

`say -v Victoria "Good morning"`

Use the following to see all the voices available

`say -v '?'`