Zsh Shell with power globbing

START HERE

`ls **/*.md`
Recurse into subdirectories

`print -l $path`
One PATH entry per line

`ls *(.)`
Only regular files

`setopt auto_cd`
Type a directory to enter it

`ls *(om[1,5])`
Five newest entries

`!!`
Repeat the last command

`${var:u}`
Uppercase a value

`autoload -Uz compinit`
Load tab completion

History !! and !$ → **alias** → **Aliases** global and suffix too → **expand** → **Expansion** ${...}, $(...), globs → **run** → **Command** status in $?

CONCEPTS

Zsh does not word-split unquoted $var, and a glob pattern that matches nothing is an error unless you add (N).

Startup files: .zshenv always, .zprofile for login, .zshrc for interactive. Put aliases and prompt in .zshrc.

Array indices start at 1, not 0, unless KSH_ARRAYS is set.

GLOBBING

`ls **/*.<ext>`
Match files in all subdirectories

`ls *(/)`
Directories only

`ls *(.)`
Regular files only
*Combine with a recursive glob: **/*(.)*

`ls *(@)`
Symlinks only

`ls *(*)`
Executable regular files only

`ls *(om[1,5])`
Five most recently modified

`ls *(Lm+10)`
Files larger than 10 MB

`ls *(mh-1)`
Modified in the last hour

`ls *(.N)`
Match or give nothing, no error

`ls ^*.log`
Everything except .log files
Needs EXTENDED_GLOB for ^ and ~ patterns

`ls *.(jpg|png)`
Alternation in a pattern

PARAMETER EXPANSION

`${var:-<default>}`
Use default if unset or empty

`${var:u}`
Uppercase

`${var:l}`
Lowercase

`${var:t}`
Tail: file name of a path

`${var:h}`
Head: directory of a path

`${var:r}`
Strip the extension

`${var:e}`
Print only the extension

`${#var}`
Length of a string

PARAMETER EXPANSION (cont.)

`${var/<old>/<new>}`
Replace first match

`${var//<old>/<new>}`
Replace all matches

`${var#<pat>}`
Trim shortest prefix

`${var%<pat>}`
Trim shortest suffix

`${(s:,:)var}`
Split on commas into an array

ARRAYS & MAPS

`arr=(a b c)`
Create an array

`$arr[1]`
First element

`$arr[-1]`
Last element

`$arr[2,3]`
Slice of elements

`arr+=(<item>)`
Append an element

`${#arr}`
Number of elements

`${arr[@]}`
All elements, each separate

`typeset -A <map>`
Declare an associative array

`map=(k1 v1 k2 v2)`
Fill a map with pairs

`print -l $arr`
One element per line

`typeset -U path`
Keep PATH entries unique
path is the array twin of PATH; edits to one change the other

`path=(~/bin $path)`
Prepend to PATH

HISTORY

`!!`
Repeat the previous command

`sudo !!`
Rerun the last command as root

`!$`
Last word of previous command

`!^`
First argument of previous command

`!*`
All arguments of previous command

`!<n>`
Run history entry number n

`!<str>`
Last command starting with str

HISTORY (cont.)

`^<old>^<new>`
Rerun last command with a fix

`history`
List commands with numbers

`Ctrl+R`
Search history backwards

`Up / Down`
Browse history

`Alt+.`
Insert last argument of previous

`setopt share_history`
Share history between sessions

`setopt hist_ignore_all_dups`
Drop older duplicate entries

OPTIONS

`setopt <option>`
Turn an option on

`unsetopt <option>`
Turn an option off

`setopt`
List options that are on

`setopt auto_cd`
Enter a directory by typing it

`setopt auto_pushd`
Make cd keep a directory stack

`setopt extended_glob`
Enable ^, ~ and # patterns

`setopt interactive_comments`
Allow # comments at the prompt

`HISTSIZE=<n> SAVEHIST=<n>`
Memory and file history sizes
Set HISTFILE too, or nothing is saved

`HISTFILE=~/.zsh_history`
Where history is stored

PROMPT, ALIASES & COMPLETION

`PROMPT='%n@%m %~ %# '`
Set the prompt

`RPROMPT='%*'`
Right-hand prompt with clock

`alias <name>='<cmd>'`
Define an alias

`autoload -Uz compinit && compinit`
Enable the completion system

`zstyle ':completion:*' menu select`
Arrow-key completion menu

`bindkey -e`
Emacs key bindings

`Tab Tab`
Show or cycle completions

PROMPT, ALIASES & COMPLETION (cont.)

`source ~/.zshrc`
Reload the config file

DIRECTORIES & FUNCTIONS

`cd -<n>`
Jump to nth stack directory

`dirs -v`
Show the directory stack

`<name>() { ... }`
Define a function

`whence -v <name>`
Say what a name resolves to

`noglob <cmd>`
Run a command without globbing

`zsh -n <file>`
Syntax check without running

DIFFERENCES FROM BASH

Behaviour changes

Topic	Bash	Zsh
Unquoted $var	word-split	not split
Array start	index 0	index 1
No glob match	passes pattern	error
Prompt read	read -p	read '?txt'
Config file	~/.bashrc	~/.zshrc
Ignore case	nocasematch	(#i)

`read -r '?Name: ' <var>`
Prompt text with read

`[[ $a == <pat> ]]`
Pattern test in conditions

`(( a < b ))`
Arithmetic condition

`for i in {1..5}; do ...`
Brace range loop

RECIPES

Starter ~/.zshrc

```
autoload -Uz compinit &&
compinit
setopt auto_cd share_history
HISTFILE=~/.zsh_history
SAVEHIST=50000
typeset -U path; path=(~/bin
$path)
```

Delete old logs, dry run first

```
ls -l **/*.log(.Lm+10mw+4)
# review the list, then
rm -- **/*.log(.Lm+10mw+4)
```

Rename extension in bulk

```
autoload -Uz zmv
zmv '(*).jpeg' '$1.jpg'
```

Split a path into parts

```
f=/srv/app/report.tar.gz
print ${f:h}   # /srv/app
print ${f:t}   # report.tar.gz
print ${f:r}   # /srv/app/
report.tar
```