

Caster Quick Reference

1. Numbers

numb <0...1m> **print digits**
word number <0...9> **print as words**

2. Alphabet ^a

arch	hotel	oscar	victor
brov	india	prime	whiskey
char	julia	quebec	x-ray
delta	kilo	romeo	yankee
echo	lima	sierra	zulu
foxy	mike	tango	
goof	novakeen	uniform	

3. Punctuation

ace ^d space	carrot ^
tabby ^d tab	questo ?
prekris ^b ()	underscore _
brax ^b []	modulo %
curly ^b { }	semper ;
angle ^b < >	deckle :
quotes ^b " "	pipe (sim symbol) ..
thin quotes ^b ' '	apostrophe ’
boom ^c ,	slash /
period dot	back slash \
clamor !	starling *
atty @	^e equals =
hashtag #	^e plus +
dolly \$	^e minus -
tilde ~	

4. Comparisons

[is] greater than >
[is] less than <
[is] greater [than] [or] equal [to] >=
[is] less [than] [or] equal [to] <=
[is] equal to ==

^aPrefix with "big" for capitals

^bLeft key is pressed after

^cSpace bar is pressed after

^dRepeatable by saying number after

^ePrefix with "long" to get spaces either side

5. Directions

sauce ^d **up**
dunce ^d **down**
lease ^d **left**
ross ^d **right**

6. Editing

clear ^d **bkspc**
deli ^d **del**
shock ^d **enter**
cancel **escape**
save **ctrl - s**

7. Selection, movement

shackle **select line**
shin ^{a b} **shift - <dir>**
queue ^{a b} **ctrl - shift - <dir>**
fly ^{a b} **ctrl - <dir>**
splat ^{a b} **ctrl - del/bkspc**
^a wally **home/end**
tell sink **end + enter**
tell dock **end + semicolon + enter**
kraken **ctrl + space**
hug <enclose> ^e **encloses target**

8. Copy and paste

stoosh ^c **copy**
cut ^c **cut**
spark ^{c f} **paste**
garb ^c **double-click + copy**
drop ^c **double-click + paste**
duple ^b **duplicate line**

^dRepeatable by saying number after

^aTakes optional direction parameter, one of the direction keys from table (5)

^bTakes optional number parameter for repetition

^cTakes prekris, brax, curly, angle, [thin] quotes

^eTakes optional number parameter, which is used to determine a persistent clipboard slot

^fTakes optional formatting parameters, e.g. "spark tie"

9. Programming core ^a

enable <language> **activate module**

iffae	lodge and
shells	lodge or
shell iffae	lodge not
switch	print to console
breaker	function
default	class
for each	add comment
for loop	long comment
do loop	value not
while loop	return
convert to integer	value true
convert to floating-point	value false
convert to string	

10. Text formatting

Capitalisation

yell **SOME WORDS**
tie **Some Words**
gerrish ^b **someWords**
sing **Some words**
laws **some words**

Spacing

gum | gun **somewords**
spine **some-words**
snake ^b **some_words**
pebble **some.words**
incline **some/words**
dissent | descent **some\words**

Capitalisation and spacing can be combined into a single command. Whether combining or not, they must be suffixed by "bow" and then dictation to format (ex: "tie snake some words")

^aSee the CCR languages quick reference in the docs for details of the language specific commands.

^bgerrish defaults to gum, snake to laws

11. Format setting

set [big] format <capitalisation> <spacing> [bow]
Set a standard/[secondary] format

[big] format <dictation>
Apply the standard/[secondary] format
to dictation

12. Mouse replacement

kick left click
kick mid middle click
psychic right click
shift right click shift right click
shift click shift click
squat left button down
bench left button up
lean right button down
hoist right button up
colic control left click
scree ^a scroll wheel
curse move cursor by pixels
douglas move cursor on grid
rainbow alternate grid
legion smart text selection
left point mouse to screen left
center point mouse to screen center
right point mouse to screen right

13. Template-based navigation

jump in ^b move cursor inside next ({{<
jump out ^b move cursor past next }}>
jump back ^b ... move cursor inside prev }}>
fill <target> ^b highlight <target> in line

^aTakes optional direction parameter, one of the direction keys from table (5)

^bAll of these are asynchronous, and can be cancelled with the word "cancel" if the search is taking too long. See *navigation.py* for a full list of targets.

14. Window management

minimise minimise window
maximise maximise window
remax force maximise
dredge [<n>] alt + tab
switch window display all windows
move window move with arrow keys
window left snap window to left
window right snap window to right
show work show active workspaces
new work create a new workspace
close work close the current workspace
close all work close all workspaces
previous work ^a go to the previous workspace
next work ^a go to the next workspace
go work <n> go to workspace number n
send work <n> . send the current window to
workspace n
move work <n> move the current window to
workspace n

15. Repetition

again do repeat last dictation
again <n> times repeat last dictation*n
record from history open macro recorder
complete save macro
delete recorded macros delete recorded macros

16. Aliasing

alias <dict> alias highlighted text
chain alias <dict> alias as CCR

17. Bring me

program/website/folder/file to bring me as
<key> create a new bring me binding
bring me <key> open bring me binding
remove <key> from bring me
restore bring me defaults

^aTakes an optional number parameter for repetition

18. Dragon

number/spell/dictation/normal/command
mode switch dragon modes

19. Update and Caster Management

clear caster log Clears Log Window
update caster Updates Caster
update dragonfly Updates Dragonfly
reboot dragon Restarts Caster
reboot windows speech recognition Restarts
Caster

20. Example command chains

shackle clear
 Select and delete the current line

queue lease stoosh three
 **Store the previous word/variable name
 in position three**

queue lease three hug brax
 **Enclose the previous three elements in
 square brackets**

shackle cut dunce four spark
 **Cut the current line, move down four,
 paste**

shin ross wally clear
 Select to the end of the line, delete

ross wally clear four
 **Go to the end of the line, backspace
 four**

snake variable name
 variable_name

curly quotes arch ross deckle quotes brov
 {"a":"b"}

backslash format command curly
 \command{}

(python) tell sink print to console spark three
 **new line, print(<third clipboard
 position>)**