

Talon Cheatsheet

1 Talon Lists

Table 1: **user.letter**

air	a
bat	b
cap	c
drum	d
each	e
fine	f
gust	g
harp	h
sit	i
jury	j
crunch	k
look	l
made	m
near	n
odd	o
pit	p
quench	q
red	r
sun	s
trap	t
urge	u
vest	v
whale	w
plex	x
yank	y
zip	z

Table 2: **user.number_key**

zero	0
one	1
two	2
three	3
four	4
five	5
six	6
seven	7
eight	8
nine	9

Table 3: **user.modifier_key**

alter	alt
control	ctrl
shift	shift
super	super
command	cmd
option	alt

Table 4: **user.special_key**

delete	backspace
forward	delete
delete	
tapper	tab
enter	enter
escape	escape
space	space

Table 5: **user.symbol_key**

dot	.
dash	-
prime	'
ampersand	&
back tick	`
comma	,
period	.
full stop	.
semicolon	;
colon	:
forward slash/	/
question	?
mark	
exclamation	!
mark	
exclamation	!
point	
asterisk	*
hash sign	#
percent sign	%
brick	‘
stroke	/
slash	/

backstroke	\
backslash	\
minus	-
equal	=
plus	+
tilde	~
bang	!
under score	_
question	?
single	'
double	"
dubquote	"
double quote	"
leper	(
open paren	(
repper)
close paren)
lack	[
open square	[
rack]
close square]
lace	{
open brace	{
race	}
close brace	}
langle	<
wrangle	>
pound	£
percy	%
tangle	^
amper	&
pipe	
semi	;
stack	:
drip	,
at sign	@
hash	#
nash	-
mash	-
euro	€
dollar	\$

Table 6: **user.arrow_key**

down	down
------	------

left	left
right	right
up	up

Table 7: **user.punctuation**

dot	.
dash	-
prime	'
ampersand	&
back tick	`
comma	,
period	.
full stop	.
semicolon	;
colon	:
forward slash/	/
question	?
mark	
exclamation	!
mark	
exclamation	!
point	
asterisk	*
hash sign	#
percent sign	%
euro sign	€
dollar sign	\$
pound sign	£

Table 8: **user.function_key**

F one	f1
F two	f2
F three	f3
F four	f4
F five	f5
F six	f6
F seven	f7
F eight	f8
F nine	f9
F ten	f10
F eleven	f11
F twelve	f12

2 Talon Formatters

Table 9: `user.formatters`

<code>allcaps</code>	<code>EXAMPLE OF FORMATTING WITH ALLCAPS</code>
<code>alldown</code>	<code>example of formatting with alldown</code>
<code>camel</code>	<code>exampleOfFormattingWithCamel</code>
<code>dotted</code>	<code>example.of.formatting.with.dotted</code>
<code>dubstring</code>	<code>"example of formatting with dubstring"</code>
<code>dunder</code>	<code>__example__offormattingwithdunder</code>
<code>hammer</code>	<code>ExampleOfFormattingWithHammer</code>
<code>kebab</code>	<code>example-of-formatting-with-kebab</code>
<code>packed</code>	<code>example::of::formatting::with::packed</code>
<code>padded</code>	<code>example of formatting with padded</code>
<code>slasher</code>	<code>/example/of/formatting/with/slasher</code>
<code>smash</code>	<code>exampleofformattingwithsmash</code>
<code>snake</code>	<code>example_of_formatting_with_snake</code>
<code>string</code>	<code>'example of formatting with string'</code>
<code>title</code>	<code>Example of Formatting With Title</code>

3 Talon Contexts

Table 10: **1password**

password new	New password
password dup	Duplicate password
password edit	Edit password
password delete	Delete password

Table 11: **1password Global**

password fill	Fill the password
password show	Show the password

Table 12: **Amethyst**

window next	Press alt-shift-j
window previous	Press alt-shift-k
window full	Press alt-shift-d
window tall	Press alt-shift-a
window middle	Press alt-shift-‘
window move main	Press alt-shift-enter
window grow	Press alt-shift-l
window shrink	Press alt-shift-h
window reevaluate	Press alt-shift-z

Table 13: **Anaconda**

anaconda	"conda "
anaconda help	"conda -help\n"
anaconda version	"conda -version\n"
anaconda environment list	"conda env list\n"
anaconda environment create	"conda env create -f "
anaconda environment re-	"conda env remove -n "
move	
anaconda activate	"conda activate "
anaconda clean	"conda clean "
anaconda compare	"conda compare "
anaconda config	"conda config "
anaconda create	"conda create "
anaconda info	"conda info "
anaconda init	"conda init "
anaconda install	"conda install "
anaconda list	"conda list "
anaconda package	"conda package "
anaconda remove	"conda remove "
anaconda uninstall	"conda uninstall "
anaconda run	"conda run "
anaconda search	"conda search "
anaconda update	"conda update "
anaconda upgrade	"conda upgrade "
anaconda build	"conda build "
anaconda convert	"conda convert "
anaconda debug	"conda debug "
anaconda develop	"conda develop "

anaconda environment	"conda env "
anaconda index	"conda index "
anaconda inspect	"conda inspect "
anaconda metapackage	"conda metapackage "
anaconda render	"conda render "
anaconda server	"conda server "
anaconda skeleton	"conda skeleton "
anaconda verify	"conda verify "

Table 14: **Discord**

[channel] mentions last	Go up to channel with unread mentions
[channel] mentions next	Go down to channel with unread mentions
oldest unread	Go to oldest unread message
toggle pins	Toggle pins popout
toggle inbox	Toggle inbox popout
toggle (members member	Toggle channel member list list)
pick emoji	Toggle emoji picker
pick (gif gif gift)	Toggle gif picker
mark inbox channel read	Mark top inbox channel read
[toggle] (mute unmute)	Toggle mute
[toggle] (deafen undeafen)	Toggle deafen
answer call	Answer incoming call
decline call	Decline incoming call

Table 15: **File Manager**

title force	Refreshes the title to match current directory. this is for e.g. windows command prompt that will need to do some magic.
manager show	Shows the pickers
manager refresh	Forces an update of the lists (e.g., when file or folder created)
go desk	Expands and opens the user directory
go docks	Expands and opens the user directory
go downloads	Expands and opens the user directory
go pictures	Expands and opens the user directory
go profile	Expands and opens the user directory
go talon home	Opens the directory that's already visible in the view
go talon user	Opens the directory that's already visible in the view
go user	Opens the directory that's already visible in the view

go back	File_manager_go_forward
go forward	File_manager_go_forward
(go parent daddy)	File_manager_open_parent
follow numb <number_small>	Let <directory> be the requested directory for the imgui display by index Opens the directory that's already visible in the view
follow user.file_manager_directories	{Opens the directory that's already visible in the view}
(select cell) folder user.file_manager_directories	{Selects the directory}
open <number_small>	Let <file> be the requested directory for the imgui display by index Opens the file
folder numb <number_small>	Let <directory> be the requested directory for the imgui display by index Selects the directory
file numb <number_small>	Let <file> be the requested directory for the imgui display by index Selects the file
file { user.file_manager_files	Selects the file
(select cell) file user.file_manager_files	{Selects the file}
folder new <user.text>	Creates a new folder in a gui filemanager or inserts the command to do so for terminals
properties show	Shows the properties for the file
terminal here	Opens terminal at current location
folder next	Next_folder_page
folder last	Previous_folder_page
file next	Next_file_page
file last	Previous_file_page

Table 16: **Firefox**

tab search	Focus address bar "% " "
tab search <user.text>	Focus address bar " " "
	Press down

Table 17: **Gdb**

until <number>	" " "
----------------	-------

for	clear all break points	"d br\n"
directory		"y\n"
break [on] clipboard		"break "
		Press ctrl-shift-v
		Press enter
list [source]		"list\n"
info source		"info source\n"
print		"p "
print [variable] <user.text>		" "
print hex		"p/x "
print hex [variable]		" "
<user.text>		" "
print string		"p/s "
hex dump <number> bytes		" "
hex dump <number> (half short) words		" "
hex dump <number> (d long) words		" "
hex dump <number> quad		" "
words		" "
hex dump		"x/100gx "
hex dump highlighted		"x/100gx "
		Copy selection to clipboard
		Paste clipboard at cursor
		Press enter
hex dump clipboard		"x/100gx "
		Paste clipboard at cursor
		Press enter
source		"source "
(list show info) display		"info display\n"
display assembly line		"display /i \$pc\n"
display source		"display "
enable display <number_small>		" "
disable display <number_small>		" "
undisplay		"undisplay\n"
(list show info) local		"info local "
(list show info) local typed		"info local -t "
(list show info) variable		"info variable "
(list show info) variable typed		"info variable -t "
(list show info) locals		"info local\n"
(list show info) variables		"info variables\n"
info threads		"info threads\n"
restart [program]		"r\n"
continue		"c\n"
back trace		"bt\n"
debug quit		"quit\n"
debug force quit		"quit\ny\n"
(show info) (inf inferiors)		"info inferiors\n"
inferior <number_small>		" "
inferior		"inferior "

resume main (inf inferior)	"inferior 1\n"
	"c\n"
resume [from] (inf inferior)	""
<number_small>	"c\n"
set args	"set args "
show follow (fork forks)	"show follow-fork-mode\n"
[mode]	
[set] follow (fork forks)	"set follow-fork-mode"
[mode] child	child\n"
[set] follow (fork forks)	"set follow-fork-mode par-
[mode] parent	ent\n"
show detach on fork	"show detach-on-fork\n"
set detach on fork	"set detach-on-fork on\n"
unset detach on fork	"set detach-on-fork off\n"
show list size	"show listsize\n"
set list size <num-	ber_small>
clear screen	"shell clear\n"

Table 18: **Generic Browser**

(address bar go address goFocus address bar url)	
(address copy url copy Focus address bar copy address copy url)	Copy selection to clipboard
go home	Go to home page
[go] forward	Go forward in the history
go (back backward)	Go back in the history
go to { user.website }	Go to a new url
go private	Open a private browsing window
bookmark it	Bookmark the current page
bookmark tabs	Bookmark all open tabs
(refresh reload) it	Reload current page
(refresh reload) it hard	Reload current page (harder)
bookmark show	Open the bookmarks editor
bookmark bar [show]	Toggle the bookmarks bar
downloads show	Show download list
extensions show	Show installed extensions
history show	Show recently visited pages
cache show	Show 'clear cache'dialog
dev tools [show]	Open or close the developer tools
show downloads	Show download list
show extensions	Show installed extensions
show history	Show recently visited pages
show cache	Show 'clear cache'dialog

Table 19: **Generic Debugger**

step into	Step into an instruction in the debugger
step over	Step over an instruction in the debugger

step line	Step into a source line in the debugger
step over line	Step over a source line in the debugger
step out	Step until function exit in the debugger
continue	Continue execution in the debugger
debug start	Start debugging
debug stop	Stop the debugger
debug exit	Exit the debugger
debug detach	Detach the debugger
debug restart	Restart execution in the debugger
show registers	Print the current registers in the debugger
get register	Print specific register in the debugger
set register	Set specific register in the debugger
break now	Break into the debugger
break here	Set a break on the current line
(list show) (breaks breakPrint the current breakpoints in the debugger	
points)	
(set add) (break breakAdd one software breakpoint in the debugger	
point)	
(set add) hardware (break Add one hardware break- break point) point in the debugger	
clear all (breaks breakClear all breakpoints in the debugger	
points)	
clear (break break point) Clear one breakpoint in the debugger	
clear (break break point) Clear one breakpoint id in <number_small> the debugger	
disable all (breaks breakDisable all breakpoints in the debugger	
points)	
disable (break break point) Disable one breakpoint in the debugger	
disable (break break point) Disable one breakpoint id in <number_small> the debugger	
enable all (breaks breakEnable all breakpoints in the debugger	
points)	
enable (break break point) Enable one breakpoint in the debugger	
enable (break break point) Enable one breakpoint id in <number_small> the debugger	
(stack back) trace	Print a back trace in the debugger
disassemble	Preps the disassemble command in the debugger
disassemble here	Disassembles instructions at the current instruction pointer

disassemble clipboard	Disassemble instructions at an address in the clipboard
jump to address	Jump to a specific address in the debugger
jump to clipboard	Jump to a specific address stored in the clipboard
jump to highlighted	Jump to a specific highlighted address in the debugger
dump string	Display as specific address as an ascii string in the debugger
dump unicode [string]	Display as specific address as an unicode string in the debugger
dump pointers	Display as specific address as a list of pointers in the debugger
list modules	List the loaded modules in the debuggee memory space
inspect type	Inspect a specific data type in the debugger
clear line	Clear unwanted data from the command line

Table 20: **I3wm**

port <number_small>	Execute a command on the system
port ten	Execute a command on the system
(port flip flipper)	Execute a command on the system
port right	Execute a command on the system
port left	Execute a command on the system
(win window) left	Execute a command on the system
(win window) right	Execute a command on the system
(win window) up	Execute a command on the system
(win window) down	Execute a command on the system
((win window) kill murder)	Execute a command on the system
(win window) stacking	Execute a command on the system
(win window) default	Execute a command on the system
(win window) tabbed	Execute a command on the system
reload i three config	Execute a command on the system

restart i three	Execute a command on the system
(full screen scuba)	Execute a command on the system
toggle floating	Execute a command on the system
focus floating	Execute a command on the system
center window	Execute a command on the system
resize mode	Execute a command on the system
focus parent	Execute a command on the system
focus child	Execute a command on the system
grow window	Execute a command on the system Press right:10 Press down:10 Press escape Execute a command on the system
shrink window	Execute a command on the system Press left:10 Press up:10 Press escape Execute a command on the system
horizontal (shell terminal)	Execute a command on the system Launch a shell
vertical (shell terminal)	Execute a command on the system Launch a shell
(shuffle move (win window) [to] port) <number_small>	Execute a command on the system
(shuffle move (win window) [to] port ten)	Execute a command on the system
(shuffle move (win window) [to] last port)	Execute a command on the system
(shuffle move (win window) left)	Execute a command on the system
(shuffle move (win window) right)	Execute a command on the system
(shuffle move (win window) up)	Execute a command on the system
(shuffle move (win window) down)	Execute a command on the system
(win window) horizontal	Execute a command on the system
(win window) vertical	Execute a command on the system

make scratch	Execute a command on the system
[(show hide)] scratch	Execute a command on the system
next scratch	Execute a command on the system
	Execute a command on the system
launch	Trigger the i3 launcher
launch <user.text>	Trigger the i3 launcher ""
lock screen	Trigger the i3 launcher
(launch shell koopa)	Launch a shell
new scratch (shell window)	Launch a shell
	Execute a command on the system
	Execute a command on the system

Table 21: **Jetbrains**

complete	Action codecompletion
perfect	Action codecompletion,action codecompletion
smart	Action smarttypecompletion
(done finish)	Action editorcompletestatement
grab <number>	Copies specified number of words to the left
(action please)	Action gotoaction
(action please) <user.text>	Action gotoaction "<text> "
refactor	Action refactorings.quicklistpopupaction
refactor <user.text>	Action refactorings.quicklistpopupaction "<text> "
extract variable	Action introducevariable
extract field	Action introducefield
extract constant	Action introduceconstant
extract parameter	Action introduceparameter
extract interface	Action extractinterface
extract method	Action extractmethod
refactor in line	Action inline
refactor move	Action move
refactor rename	Action renameelement
rename file	Action renamefile
fix (format formatting)	Action reformatcode
fix imports	Action optimizeimports
(go declaration follow)	Action gotodeclaration
go implementation	Action gotoimplementation
go usage	Action findusages
go type	Action gototypeddeclaration
go test	Action gototest
go back	Action back

go forward	Action forward
find (everywhere all)	Action searcheverywhere
find (everywhere all)<user.text>[over]	Action searcheverywhere "<text> "
(search find) class	Action gotoClass
(search find) file	Action gotoFile
(search find) path	Action findInPath
(search find) symbol	Action gotoSymbol
(search find) symbol <user.text>	Action gotoSymbol "<text> " Press enter
recent	Action recentFiles
surround [this] <user.text>[over]	withidea("action SurroundWith")sle Action generate
insert <user.text>[over]	generatedAction generate "<text> "
insert <user.text>[over]	templateidea("action InsertLiveTemplate")
create (template snippet)	Action saveasTemplate
toggle recording	Action startstopmacrorecording
change (recording recordings)	Action editmacros
play recording	Action playbacklastmacro
play <user.text>[over]	recordingidea("action PlaySavedMacrosAc")
go mark	Action showbookmarks
toggle mark	Action togglebookmark
go next mark	Action gotoNextbookmark
go last mark	Action gotoPreviousbookmark
toggle mark <number>	
go mark <number>	
expand deep	Action expandRegionrecursively
expand all	Action expandAllRegions
collapse deep	Action collapseRegionrecursively
collapse all	Action collapseAllRegions
go next (method function)	Action methoddown
go last (method function)	Action methodup
clippings	Action pasteMultiple
copy path	Action copyPaths
copy reference	Action copyReference
copy pretty	Action copyAsRichText
create sibling	Action newElementsSamePlace
create <user.text>[over]	siblingAction newElementsSamePlace "<text> "
create file	Action newElement
create file <user.text>[over]	Action newElement "<text> "
go task	Action tasks.goto
go browser task	Action tasks.open.in.browser
switch task	Action tasks.switch

clear task	Action tasks.close
configure servers	Action tasks.configure.servers
git pull	Action vcs.updateproject
git commit	Action checkinproject
git push	Action checkinproject
git log	Action vcs.showtabbedfilehistory
git browse	Action github.open.in.browser
git (gets gist)	Action github.create.gist
git (pull request request)	Action github.create.pull.request
git (view show list) (re-quests request)	Action github.view.pull.request
git (annotate blame)	Action annotate
git menu	Action vcs.quicklistpopupaction
toggle project	Action activateprojecttoolwindow
toggle find	Action activatefindtoolwindow
toggle run	Action activateruntoolwindow
toggle debug	Action activateddebugtoolwindow
toggle events	Action activateeventlogtoolwindow
toggle terminal	Action activateterminaltoolwindow
toggle git	Action activateversioncontroltoolwindow
toggle structure	Action activatestructuretoolwindow
toggle database	Action activatedatabasetoolwindow
toggle database changes	Action activate-databasechangestoolwindow
toggle make	Action activatemaketoolwindow
toggle to do	Action activatetodotoolwindow
toggle docker	Action activatedockertoolwindow
toggle favorites	Action activatefavoritestoolwindow
toggle last	Action jumptolastwindow
toggle pinned	Action togglepinnedmode
toggle docked	Action toggledockmode
toggle floating	Action togglefloatingmode
toggle windowed	Action togglewindowedmode
toggle split	Action togglesidemode
toggle tool buttons	Action viewtoolbuttons
toggle toolbar	Action viewtoolbar

toggle status [bar]	Action viewstatusbar
toggle navigation [bar]	Action viewnavigationbar
toggle power save	Action togglepowersave
toggle whitespace	Action editortoggleshowwhitespaces
toggle indents	Action editortoggleshowindentlines
toggle line numbers	Action editortoggleshowlinenumbers
toggle (bread crumbs bread-crumbs)	Action editortoggleshowbreadcrumbs
toggle gutter icons	Action editortoggleshowguttericons
toggle wrap	Action editortoggleusesoftwarewraps
toggle parameters	Action toggleinlinehintsaction
toggle fullscreen	Action togglefullscreen
toggle distraction [free mode]	Action toggledistraction-freemode
toggle presentation [mode]	Action togglepresentation-mode
toggle comment	Toggle comments on the current line(s)
change scheme	Action quickchangescheme
(toggle pop) (doc docu-mentation)	Action quickjavadoc
(pop deaf toggle definition)	Action quickimplementations
pop type	Action expressiontypeinfo
pop parameters	Action parameterinfo
go breakpoints	Action viewbreakpoints
toggle [line] breakpoint	Action togglelinebreakpoint
toggle method breakpoint	Action togglemethodbreakpoint
run menu	Action chooserunconfiguration
run test	Action runclass
run test again	Action rerun
debug test	Action debugclass
step over	Action stepover
step into	Action stepinto
step smart	Action smartstepinto
step to line	Action runtocursor
continue	Action resume
(grow shrink) window right	Action resizetoolwindowright
(grow shrink) window left	Action resizetoolwindowleft
(grow shrink) window up	Action resizetoolwindowup
(grow shrink) window down	Action resizetoolwindow-down
go next (error air)	Action gotonexterror
go last (error air)	Action gotopreviouserror
fix next (error air)	Action gotonexterror
	Action showintentionactions

fix last (error air)	Action gotopreviouserror
	Action showintentionactions
select less	Action editorunselectword
select (more this)	Action editorselectword
expand <number>	untilSelects lines from line_start
<number>	to line line_end
	Action expandregion
collapse <number>	untilSelects lines from line_start
<number>	to line line_end
	Action collapseregion
paste <number>	untilSelects lines from line_start
<number>	to line line_end
	Action editorpaste
refactor <number>	untilSelects lines from line_start
<number>	to line line_end
	Action refactor-ings.quicklistpopupaction
clone <number>	Clones specified line at current position
clear last <user.text>[over]	
clear next <user.text>[over]	
comment	last
<user.text>[over]	
comment	next
<user.text>[over]	
go last <user.text>[over]	
go next <user.text>[over]	
paste last <user.text>[over]	
paste next <user.text>[over]	
refactor	last
<user.text>[over]	
refactor	next
<user.text>[over]	
replace	last
<user.text>[over]	
replace	next
<user.text>[over]	
select last <user.text>[over]	
select next <user.text>[over]	
select camel left	Extends the selection by camel/subword to the left
select camel right	Extends the selection by camel/subword to the right
go camel left	Moves cursor to the left by camel case/subword
go camel right	Move cursor to the right by camel case/subword
blacken	Action blackreformatcode

Table 22: **Kubectl**

cube [control]	"kubectl "
cube create	"kubectl create "
cube expose	"kubectl expose "
cube run	"kubectl run "

cube set	"kubectl set "
cube run container	"kubectl run-container "
cube explain	"kubectl explain "
cube get	"kubectl get "
cube edit	"kubectl edit "
cube delete	"kubectl delete "
cube rollout	"kubectl rollout "
cube rolling update	"kubectl rolling-update "
cube scale	"kubectl scale "
cube auto scale	"kubectl autoscale "
cube certificate	"kubectl certificate "
cube top	"kubectl top "
cube drain	"kubectl drain "
cube taint	"kubectl taint "
cube (cord cordon)	"kubectl cordon "
cube (uncord uncordon)	"kubectl uncordon "
cube cluster (info informa- tion)	"kubectl cluster-info "
cube describe	"kubectl describe "
cube logs	"kubectl logs "
cube attach	"kubectl attach "
cube exec	"kubectl exec "
cube port forward	"kubectl port-forward "
cube proxy	"kubectl proxy "
cube copy	"kubectl cp "
cube auth	"kubectl auth "
cube diff	"kubectl diff "
cube apply	"kubectl apply "
cube patch	"kubectl patch "
cube replace	"kubectl replace "
cube wait	"kubectl wait "
cube convert	"kubectl convert "
cube customize	"kubectl kustomize "
cube label	"kubectl label "
cube annotate	"kubectl annotate "
cube completion	"kubectl completion "
cube (interface API)	"kubectl api "
cube interface resources	"kubectl api-resources "
cube interface versions	"kubectl api-versions "
cube config	"kubectl config "
cube help	"kubectl help "
cube plugin	"kubectl plugin "
cube version	"kubectl version "
cube { user.kubectl_action }	"<kubectl_object or "> "
cube detach	Press ctrl-p
	Press ctrl-q
cube shell	"kubectl exec -it - /bin/bash" Press left:13

Table 23: **Dunst (Linux)**

show notifications	Press ctrl-‘
--------------------	--------------

dismiss [notifications]	Execute a command on the system
dismiss all [notifications]	Execute a command on the system

Table 24: **Keepassx (Linux)**

open database	Press ctrl-o
save database	Press ctrl-s
close database	Press ctrl-w
lock database	Press ctrl-l
quit	Press ctrl-q
[add] new entry	Press ctrl-n
clone entry	Press ctrl-k
(view edit) entry	Press ctrl-e
delete entry	Press ctrl-d
copy user [name]	Press ctrl-b
copy password	Press ctrl-c
open (earl url link)	Press ctrl-u
copy (earl url link)	Press ctrl-alt-u
find	Press ctrl-f
find <user.text>	Press ctrl-f
	" "

Table 25: **Signal (Linux)**

show shortcuts	Press ctrl-/
(next nav navigate) [by]	Press ctrl-t
(sec section)	
(prev previous) (chat conversation)	Press alt-down
next (chat conversation)	Press alt-up
(prev previous) unread	Press alt-shift-down
next unread	Press alt-shift-up
[open] (pref preferences)	Press ctrl-,
open conversation menu	Press ctrl-shift-l
search	Press ctrl-f
search chat	Press ctrl-shift-f
focus (chat composer)	Press ctrl-shift-t
open media	Press ctrl-shift-m
open emoji	Press ctrl-shift-j
open sticker	Press ctrl-shift-s
record [voice] message	Press ctrl-shift-v
archive chat	Press ctrl-shift-a
unarchive chat	Press ctrl-shift-u
(first top) message	Press ctrl-up
(last bottom) message	Press ctrl-down
close chat	Press ctrl-shift-c
send it	Press enter
message details	Press ctrl-d
reply [message]	Press ctrl-shift-r
react [message]	Press ctrl-shift-e
save attachment	Press ctrl-s
delete [message]	Press ctrl-shift-d

send message	Press ctrl-enter
expand chat	Press ctrl-shift-x
attach [file]	Press ctrl-u
remove [link] preview	Press ctrl-p
remove [link] attachment	Press ctrl-shift-p

Table 26: **Taskwarrior (Linux)**

task version	"task -version\n"
task commands	"task commands\n"
task help	"task help\n"
task list	"task list\n"
task list orphans	"task project: list\n"
task list untagged	"task tags.none: list\n"
task list <user.text>	" "
task list project	"task list project: "
task list project <user.text>	" "
task add	"task add "
task add <user.text>	" "
task undo	"task undo\n"
(tasks task next)	"task next\n"
task <number> edit	" "
task <number> done	" "
task <number> delete	" "

Table 27: **Termite (Linux)**

shell yank	Press y
shell select	Press ctrl-shift-space
shell insert	Press escape
visual line	Press v
visual line mode	Press v

Table 28: **Tmux (Linux)**

mux	"tmux "
mux new session	"tmux new "
mux sessions	Press ctrl-b
	Press s
mux name session	Press ctrl-b
	Press \$
mux kill session	"tmux kill-session -t "
mux new window	Press ctrl-b
	Press c
mux window <number>	Press ctrl-b
	Press
mux previous window	Press ctrl-b
	Press p
mux next window	Press ctrl-b
	Press n
mux rename window	Press ctrl-b
	Press ,
mux close window	Press ctrl-b
	Press &

mux split horizontal	Press ctrl-b Press %
mux split vertical	Press ctrl-b Press "
mux next pane	Press ctrl-b Press o
mux	movePress ctrl-b
<user.arrow_key>	Press arrow_key
mux close pane	Press ctrl-b Press x
mux pane numbers	Press ctrl-b Press q

Table 29: **Finder (MacOS)**

preferences	Press cmd-,
options	Press cmd-j
search	Press cmd-alt-f
sort by none	Press ctrl-alt-cmd-0
sort by name	Press ctrl-alt-cmd-1
sort by kind	Press ctrl-alt-cmd-2
sort by date opened	Press ctrl-alt-cmd-3
sort by date added	Press ctrl-alt-cmd-4
sort by date modified	Press ctrl-alt-cmd-5
sort by size	Press ctrl-alt-cmd-6
icon view	Press cmd-1
column view	Press cmd-3
list view	Press cmd-2
gallery view	Press cmd-4
copy path	Press alt-cmd-c
trash it	Press cmd-backspace
hide [finder]	Press cmd-h
hide others	Hide all other windows

Table 30: **Rstudio (MacOS)**

run that	Press cmd-enter
run document	Press cmd-alt-r
run from top	Press cmd-alt-b
run to end	Press cmd-alt-e
run (function funk)	Press cmd-alt-f
run section	Press cmd-alt-t
run previous chunks	Press cmd-alt-p
run chunk	Press cmd-alt-c
run next chunk	Press cmd-alt-n
run all	Press cmd-shift-s
run knitter	Press cmd-shift-k
run profiler	Press cmd-shift-alt-p
jump back	Press cmd-f9
jump forward	Press cmd-f10
close all tabs	Press cmd-shift-w
indent lines	Press cmd-i
toggle comment	Press cmd-shift-c
reformat comment	Press cmd-shift-/
reformat R code	Press cmd-shift-a

line up	Press alt-up
line down	Press alt-down
duplicate line up	Press cmd-alt-up
duplicate line [down]	Press cmd-alt-down
select to paren	Press ctrl-shift-e
select to matching paren	Press ctrl-shift-alt-e
jump to matching	Press ctrl-p
expand selection	Press shift-alt-cmd-up
reduce selection	Press shift-alt-cmd-down
add cursor up	Press ctrl-alt-up
add cursor down	Press ctrl-alt-down
move active cursor up	Press ctrl-alt-shift-up
move active cursor down	Press ctrl-alt-shift-down
delete line	Press cmd-d
delete word left	Press alt-backspace
delete word right	Press alt-delete
assign that	Press alt-
pipe that	Press cmd-shift-m
insert knitter chunk	Press cmd-alt-i
fold that	Press cmd-alt-l
unfold that	Press cmd-shift-alt-l
fold all	Press cmd-alt-o
unfold all	Press cmd-shift-alt-o
find and replace	Press cmd-f
find next	Press cmd-g
find previous	Press cmd-shift-g
find with selection	Press cmd-e
find in files	Press cmd-shift-f
run replace	Press cmd-shift-j
run spell check	Press f7
go to source	Press ctrl-1
go to console	Press ctrl-2
go to help	Press ctrl-3
go to history	Press ctrl-4
go to files	Press ctrl-5
go to (plots plot)	Press ctrl-6
go to packages	Press ctrl-7
go to environment	Press ctrl-8
go to git	Press ctrl-9
go to build	Press ctrl-0
go to terminal	Press alt-shift-t
go to omni	Press ctrl-.
go to line	Press cmd-shift-alt-g
go to section	Press cmd-shift-alt-j
go to tab	Press ctrl-shift-.
go to previous tab	Press ctrl-f11
go to next tab	Press ctrl-f12
go to first tab	Press ctrl-shift-f11
go to last tab	Press ctrl-shift-f12
zoom source	Press ctrl-shift-1
(zoom show) all	Press ctrl-shift-0
help that	Press f1
define that	Press f2
previous plot	Press cmd-alt-f11

next plot	Press cmd-alt-f12
restart R session	Press cmd-shift-f10
dev tools build	Press cmd-shift-b
dev tools load all	Press cmd-shift-l
dev tools test	Press cmd-shift-t
dev tools check	Press cmd-shift-e
dev tools document	Press cmd-shift-d
toggle breakpoint	Press shift-f9
debug next	Press f10
debug step into (function funk)	Press shift-f4
debug finish (function funk)	Press shift-f6
debug continue	Press shift-f5
debug stop	Press shift-f8
run git diff	Press ctrl-alt-d
run git commit	Press ctrl-alt-m

Table 31: **Apple Terminal (MacOS)**

suspend	Press ctrl-z
resume	"fg" Press enter

Table 32: **Explorer (Windows)**

go <user.letter>	File _manager_ open _volume
go app data	Opens the directory that's already visible in the view
go program files	Opens the directory that's already visible in the view

Table 33: **Outlook (Windows)**

archive	Press alt-h-o-1
new e-mail	Press ctrl-n
calendar	Press ctrl-2
inbox	Press ctrl-1
Reply	Press ctrl-r
Reply all	Press ctrl-shift-r
Forward	Press ctrl-f
accept	Press shift-f10-c-c-enter

Table 34: **Windows Terminal (Windows)**

settings open	Press ctrl-,
focus left	Press ctrl-alt-shift-left
focus right	Press ctrl-alt-shift-right
focus up	Press ctrl-alt-shift-up
focus down	Press ctrl-alt-shift-down
split right	Press ctrl-shift-h
split down	Press ctrl-h
term menu	Press ctrl-shift-f1
run last	Press up-enter
kill all	Press ctrl-c "y" Press enter

find it	Open find dialog, optionally searching for text
find it <phrase>	Press escape Open find dialog, optionally searching for text

Table 35: **Ubuntu (Windows)**

go <user.letter>	File _manager_ open _volume
------------------	-----------------------------

Table 36: **Wsl (Windows)**

(wsl weasel) reset path de-Reset wsl path detection tection
--

Table 37: **Slack Mac (MacOS)**

workspace <number>	Press
(slack lack) [channel] info	Press cmd-shift-i
(move next) focus	Press ctrl-'
[next] (section zone)	Press f6
(previous last) (section zone)	Press shift-f6
(slack lack) [direct] mes-sages	Press cmd-shift-k
(slack lack) threads	Press cmd-shift-t
(slack lack) (history [next] back backward)	Press cmd-[
(slack lack) forward	Press cmd-]
[next] (element bit)	Press tab
(previous last) (element bit)	Press shift-tab
(slack lack) (my stuff ac-tivity)	Press cmd-shift-m
(slack lack) directory	Press cmd-shift-e
(slack lack) (starred [items] stars)	Press cmd-shift-s
(slack lack) unread [mes-sages]	Press cmd-shift-a
(go undo toggle) full	Press ctrl-cmd-f
grab left	Press shift-up
grab right	Press shift-down
add line	Press shift-enter
(slack lack) (slap slaw slapper)	Press cmd-right-shift-enter
(slack lack) (react reac-tion)	Press cmd-shift-\\
(insert command comman-dify)	Press cmd-shift-c
insert code	"~~~~~" Press left-left-left Press shift-enter Press shift-enter Press up
(slack lack) (bull bullet bulleted) [list]	Press cmd-shift-8

(slack lack) (number num-bered) [list]	Press cmd-shift-7
(slack lack) (quotes quota-tion)	Press cmd-shift->
bold	Press cmd-b
(italic italicize)	Press cmd-i
(strike strikethrough)	Press cmd-shift-x
(clear scrap scratch)	Press cmd-a-backspace
(slack lack) snippet	Press cmd-shift-enter
([toggle] mute unmute)	Press m
(slack lack) ([toggle] video)	Press v
(slack lack) invite	Press a
(slack lack) shortcuts	Press cmd-/
emote <user.text>	""
toggle left sidebar	Press cmd-shift-d
toggle right sidebar	Press cmd-.

Table 38: **Slack Win (Windows)**

workspace <number>	Press
(slack lack) [channel] info	Press ctrl-shift-i
(move next) focus	Press ctrl-'
[next] (section zone)	Press f6
(previous last) (section zone)	Press shift-f6
(slack lack) [direct] mes-sages	Press ctrl-shift-k
(slack lack) threads	Press ctrl-shift-t
(slack lack) (history [next] back backward)	Press alt-left
(slack lack) forward	Press alt-right
[next] (element bit)	Press tab
(previous last) (element bit)	Press shift-tab
(slack lack) (my stuff ac-tivity)	Press ctrl-shift-m
(slack lack) directory	Press ctrl-shift-e
(slack lack) (starred [items] stars)	Press ctrl-shift-s
(slack lack) unread [mes-sages]	Press ctrl-shift-a
grab left	Press shift-up
grab right	Press shift-down
add line	Press shift-enter
(slack lack) (react reac-tion)	Press ctrl-shift-\
(insert command comman-dify)	Press ctrl-shift-c
insert code	"""""" Press left-left-left Press shift-enter Press shift-enter Press up

(slack lack) (bull bullet-bulleted) [list]	Press ctrl-shift-8
(slack lack) (number num-bered) [list]	Press ctrl-shift-7
(slack lack) (quotes quota-tion)	Press ctrl-shift-9
bold	Press ctrl-b
(italic italicize)	Press ctrl-i
(strike strikethrough)	Press ctrl-shift-x
(clear scrap scratch)	Press ctrl-a-backspace
(slack lack) snippet	Press ctrl-shift-enter
([toggle] mute unmute)	Press m
(slack lack) ([toggle] video)	Press v
(slack lack) invite	Press a
(slack lack) shortcuts	Press ctrl-/
emote <user.text>	""
toggle left sidebar	Press ctrl-shift-d
toggle right sidebar	Press ctrl-.

Table 39: **Talon Repl**

test last	Let <phrase> be the history entry at the specified index Let <command> be "<command> " Press enter
test <phrase>	"" Press enter
test numb <number_small>	Let <phrase> be the history entry at the specified index Let <command> be "<command> " Press enter
debug action { user.talon_actions }	{} Press enter
debug list { user.talon_lists }	{} Press enter
debug tags	"actions.user.talon_pretty_print(reg...) Press enter
debug settings	"actions.user.talon_pretty_print(reg...) Press enter
debug modes	"actions.user.talon_pretty_print(sc...) Press enter
debug scope { user.talon_scopes }	{} Press enter
debug running apps	"actions.user.talon_pretty_print(ui...) Press enter
debug all windows	"actions.user.talon_pretty_print(ui...) Press enter
debug { user.running } win-dows	{} Press enter

Table 40: **Teams**

show shortcuts	Press ctrl-.
----------------	--------------

[go][to] search	Press ctrl-e
show commands	Press ctrl-/
open filter	Press ctrl-shift-f
go to	Press ctrl-g
open (apps applications)	Press ctrl-‘
[start] new chat	Press ctrl-n
open settings	Press ctrl-,
open help	Press fl
close	Press escape
open activity	Press ctrl-1
open chat	Press ctrl-2
open teams	Press ctrl-3
open calendar	Press ctrl-4
open planner	Press ctrl-5
open calls	Press ctrl-6
open files	Press ctrl-7
[go][to] (prev previous) [list] item	Press alt-up
[go][to] next [list] item	Press alt-down
move [selected] team up	Press ctrl-shift-up
move [selected] team down	Press ctrl-shift-down
[go][to] (prev previous) section	Press ctrl-shift-f6
[go][to] next section	Press ctrl-f6
[go][to] compose [box]	Press c
[expand] compose [box]	Press ctrl-shift-x
send	Press ctrl-enter
attach file	Press ctrl-o
[start] new line	Press shift-enter
reply [to][thread]	Press r
accept video call	Press ctrl-shift-a
accept audio call	Press ctrl-shift-s
decline call	Press ctrl-shift-d
start audio call	Press ctrl-shift-c
start video call	Press ctrl-shift-u
toggle mute	Press ctrl-shift-m
start screen share session	Press ctrl-shift-e
toggle video	Press ctrl-shift-o
[go][to] sharing toolbar	Press ctrl-shift-space
decline screen share	Press ctrl-shift-d
accept screen share	Press ctrl-shift-a
schedule [a] meeting	Press alt-shift-n
go to current time	Press alt-.
go to (prev previous) (day week)	Press ctrl-alt-left
go to next (day week)	Press ctrl-alt-right
view day	Press ctrl-alt-1
view work week	Press ctrl-alt-2
view week	Press ctrl-alt-3
(safe send) meeting request	Press ctrl-s
join [from] meeting [details]	Press alt-shift-j
go to suggested time	Press alt-shift-s

Table 41: **Teams Mac (MacOS)**

open history	Press super-shift-h
view shortcuts	Press super-.
show shortcuts	Press super-.
[go][to] search	Press super-e
show commands	Press super-/
open filter	Press super-shift-f
go to	Press super-g
open (apps applications)	Press ctrl-‘
[start] new chat	Press super-n
open settings	Press super-,
open help	Press fl
close	Press escape
reset	Press escape
	Press escape
	Press escape
zoom reset	Press super-0
open (act activity)	Press super-1
open chat	Press super-2
open teams	Press super-3
open calendar	Press super-4
open calls	Press super-5
open files	Press super-6
[go][to] (prev previous) [list] item	Press alt-up
[go][to] next [list] item	Press alt-down
move [selected] team up	Press super-shift-up
move [selected] team down	Press super-shift-down
[go][to] (prev previous) section	Press super-shift-f6
[go][to] next section	Press super-f6
[go][to] compose [box]	Press shift-alt-c
[expand] compose [box]	Press super-shift-x
send message	Press super-enter
attach file	Press shift-alt-o
attach local file	Press shift-alt-o
	Press down
	Press space
[start] new line	Press shift-enter
reply [to][thread]	Press shift-alt-r
accept video call	Press super-shift-a
accept audio call	Press super-shift-s
decline call	Press super-shift-d
start audio call	Press super-shift-c
start video call	Press super-shift-u
toggle mute	Press super-shift-m
start screen share session	Press super-shift-e
toggle video	Press super-shift-o
[go][to] sharing toolbar	Press super-shift-space
decline screen share	Press super-shift-d
accept screen share	Press super-shift-a
schedule [a] meeting	Press alt-shift-n
go to current time	Press alt-.

go to (prev previous) (day week)	Press super-alt-left
go to next (day week)	Press super-alt-right
view day	Press super-alt-1
view work week	Press super-alt-2
view week	Press super-alt-3
(save send) meeting request	Press super-s
join [from] meeting [details]	Press alt-shift-j
go to suggested time	Press alt-shift-s
(raise lower) hand	Press super-shift-k
leave team meeting	Press super-shift-h

Table 42: **Github**

focus search	Press s
go to notifications	"gn"
go to dashboard	"gd"
(keyboard shortcuts show show keyboard shortcuts)	Press ?
(selection move down move selection down)	Press j
(selection move up move selection up)	Press k
(selection toggle toggle selection)	Press x
(selection open open selection)	Press o
go to code	"gc"
go to issues	"gi"
go to pull requests	"gp"
go to wiki	"gw"
go to actions	"ga"
go to projects	"gb"
go to discussions	"gg"
[web] editor open	Press .
(file find find file)	Press t
jump to line	Press l
((branch tag) switch switch (branch tag))	Press w
(url expand expand url)	Press y
(show hide) comments	Press i
blame view open	Press b
(show hide) annotations	Press a
(issue create create [an] issue)	Press c
search (issues pull requests)	Press /
(filter by edit) labels	Press l
(filter by edit) milestones	Press m
(filter by edit) assignee	Press a
reply	Press r
(comment submit submit comment)	Press ctrl-enter
(comment preview preview comment)	Press ctrl-shift-p

git hub full screen	Press ctrl-shift-l
(form close close form)	Press escape
parent commit	Press p
other parent commit	Press o
mark as read	Press y
(thread mute mute thread)	Press shift-m
(issue open open issue)	Press o
(issue create create issue)	Press c
reviewer request	Press q
milestone set	Press m
assignee set	Press a
label set	Press l
go to workflow	"gf"
timestamps toggle	Press shift-t
fullscreen toggle	Press shift-f

Table 43: **Gitlab**

show shortcuts	Press ?
go to projects [page]	Press shift-p
go to groups [page]	Press shift-g
go to activity [page]	Press shift-a
go to milestones [page]	Press shift-l
go to snippets [page]	Press shift-s
search page	Press s
go to issues [page]	Press shift-i
go to merge requests [page]	Press shift-m
go to to do [list][page]	Press shift-t
(show hide) performance bar	Press p
edit last comment	Press l
toggle mark down [preview]	Press ctrl-shift-p
go [to] project home [page]	"gp"
go [to] project activity [feed]	"gv"
go [to] project releases [list]	"gr"
go [to] project files [list]	"gf"
go [to] project file search [page]	Press t
go [to] project (commit commits) [list]	"gc"
go [to] (repository repository) [graph [page]]	"gn"
go [to] (repository repository) charts	"gd"
go [to] project issues [list]	"gi"
go [to] new issues [list]	"i"
go [to] project issues boards [list]	"gb"
go [to] project merge requests [list]	"gm"
go [to] jobs [list]	"gj"
go [to] project metrics	"gl"
go [to] project environments	"ge"
go [to] project cubes	"gk"

go [to] project snippets [list]"gs"	
go [to] project wiki "gw"	
edit description	Press e
change assignee	Press a
change milestone	Press m
change label	Press l
right comment	Press r
next [unresolved] discussion	Press n
previous [unresolved] discussion	Press p
next file	Press]
previous file	Press [
back to files	Press escape
open permalink	Press y
edit page	Press e

Table 44: **Outlook**

new message	Press n
send [this] message	Press alt-s
reply [to][this] message	Press r
reply all [to][this] message	Press ctrl-shift-r
forward [this] message	Press ctrl-shift-f
save [draft]	Press ctrl-s
discard [draft]	Press esc
insert [a][hyper] link	Press ctrl-k
(select unselect) [this] message	Press ctrl-space
select all [messages]	Press ctrl-a
clear all [messages]	Press esc
select first [message]	Press home
select last [message]	Press end
open [this] message	Press o
open [this] message [in][a]	Press shift-enter
new window	
close [this] message	Press esc
[open][the] next (item message)	Press ctrl-. sage)
[open][the] (prev previous) item	Press ctrl-,
next reading [pane] (item message)	Press .
(prev previous) [pane] (item message)	Press ,
(expand collapse) [conversation]	Press x
go [to] mail	Press ctrl-shift-1
go [to] calendar	Press ctrl-shift-2
go [to] people	Press ctrl-shift-3
go [to] to do	Press ctrl-shift-4
go [to] inbox	Press g Press i
go to drafts	Press g Press d

go to sent	Press g
	Press s
search [email]	Press alt-q
show help	Press ?
undo [last][action]	Press ctrl-z
delete [this][message]	Press delete
(perm permanently) delete [this][message]	Press shift+delete
new folder	Press shift-e
mark [this][(item message)] as read	Press q
mark [this][(item message)] as unread	Press u
flag [this][(item message)]	Press insert
archive	Press e
mark [this][message][as] junk	Press j
moved to [a] folder	Press v
categorize [this] message	Press c

Table 45: **Protonmail**

open help	Press ?
[focus] search	Press /
confirm active	Press enter
close active	Press escape
open command [palette]	Press shift-space
new message	Press c
send message	Press ctrl-enter
save message	Press ctrl-s
(go jump) [to] inbox	Press g Press i
(go jump) [to] draft	Press g Press d
(go jump) [to] sent	Press g Press s
(go jump) [to] starred	Press g Press .
(go jump) [to] archive	Press g Press a
(go jump) [to] spam	Press g Press x
(go jump) [to] trash	Press g Press t
(prev previous) message	Press up
next message	Press down
exit message	Press left
enter message	Press right
(show display) newer [message]	Press k
(show display) older [message]	Press j
open message	Press enter
go back	Press escape
select all	Press * Press a

(deselect unselect) all	Press *
	Press n
select [the] (message conversation)	Press x
mark [as] read	Press r
mark [as] unread	Press u
star (message conversation)	Press .
move to inbox	Press i
move to trash	Press t
move to archive	Press a
move to spam	Press s
reply to (message conversation)	Press shift-r
reply all [to] (message conversation)	Press shift-a
forward (message conversation)	Press shift-f
(prev previous) contact	Press up
next contact	Press down
enter contact	Press right
delete contact	Press t
exit contact	Press left
save contact	Press ctrl-s

Table 46: **Twitter**

(show shortcuts shortcuts help)	Press ?
next tweet	Press j
previous tweet	Press k
page down	Press space
load new tweet	Press .
go home	"gh"
go explore	"ge"
go notifications	"gn"
go mentions	"gr"
go profile	"gp"
go likes	"gl"
go lists	"gi"
go direct messages	"gm"
go settings	"gs"
go book marks	"gb"
go to user	"gu"
display settings	"gd"
new tweet	Press n
send tweet	Press ctrl-enter
new direct message	Press m
search	Press /
like message	Press l
reply message	Press r
re tweet [message]	Press t
share tweet	Press s
bookmark	Press b
mute account	Press urge

block account	Press x
open details	Press enter
expand photo	Press o

Table 47: **Windbg (Windows)**

register <user.registers>	Press @ ""
open help	".hh\n"
add microsoft symbols	"srv*c:\symbols*http://msdl.microsoft.com/symbols/srv\n"
force reload symbols	".reload /f\n"
reload symbols	".reload\n"
loaded modules	"lm l\n"
display pointers	"dps "
dereference pointer	"poi()" Move cursor left one column
show version	Press ctrl-alt-w
view command	Press alt-1
view watch	Press alt-2
view locals	Press alt-3
view registers	Press alt-4
view memory	Press alt-5
view call stack	Press alt-6
view disassembly	Press alt-7
view scratch pad	Press alt-8
view (processes threads)	Press alt-9
dump function params (lib library)	"r @rcx,@rdx,@r8,@r9\n"
<user.windows_dlls>	

Table 48: **Batch**

soft exit	"exit /b 1\n"
hard exit	"exit 1\n"
echo	"echo "
echo off	"@echo off\n"
call	"call "
call shell	"call cmd \c "
if error	"if errorlevel 1 "
go to	"goto "
delayed expansion	"setlocal enabledelayedexpansion\n"
arg <number_small>	""

Table 49: **Go**

variadic	"..."
logical and	"& & "
logical or	" "
state comment	"// "
[line] comment <user.text>	Press cmd-right "// " "<text> (formatted with sentence)"
state (funk func fun)	"func "

function (Annette init)"func init() { \n"	
[over]	
function <user.text>[over]	"func "
	"<text> (formatted with private_camel_case)"
	"("
method <user.text>[over]	"meth "
	"<text> (formatted with private_camel_case)"
state var	"var "
variable [<user.text>][over]	"var "
	"<text> (formatted with private_camel_case)"
of type [<user.text>][over]	" "
	"<text> (formatted with private_camel_case)"
state break	"break"
state (chan channel)	"chan "
state go	"go "
state if	"if "
if <user.text>[over]	"if "
	"<text> (formatted with private_camel_case)"
spawn <user.text>[over]	"go "
	"<text> (formatted with private_camel_case)"
state else if	"else if "
else if <user.text>[over]	"else if "
	"<text> (formatted with private_camel_case)"
state else	"else "
else <user.text>[over]	"else { "
	Press enter
	"<text> (formatted with private_camel_case)"
state while	"while "
while <user.text>[over]	"while "
	"<text> (formatted with private_camel_case)"
state for	"for "
for <user.text>[over]	"for "
	"<text> (formatted with private_camel_case)"
state for range	"forn "
range <user.text>[over]	"forn "
	"<text> (formatted with private_camel_case)"
state format	"fmt"
format <user.text>[over]	"fmt."
	"<text> (formatted with public_camel_case)"
state switch	"switch "

switch <user.text>[over]	"switch "
	"<text> (formatted with private_camel_case)"
state select	"select "
state (const constant)	"const "
constant <user.text>[over]	"const "
	"<text> (formatted with public_camel_case)"
state case	"case "
state default	"default:"
case <user.text>[over]	"case "
	"<text> (formatted with private_camel_case)"
state type	"type "
type <user.text>[over]	"type "
	"<text> (formatted with public_camel_case)"
state true	"true "
state false	"false "
state (start struct struck)"struct { "	
	Press enter
(struct struck)"struct { "	
<user.text>[over]	Press enter
	"<text> (formatted with public_camel_case)"
[state] empty interface	"interface { } "
state interface	"interface { "
	Press enter
interface <user.text>[over]	"interface { "
	Press enter
	"<text> (formatted with public_camel_case)"
state string	"string "
[state] (int integer ant)	"int "
state slice	"[] "
slice of	"[] "
[state] (no nil)	"nil"
state (int integer ant) 64	"int64 "
state tag	"<<"
	Press left
field tag <user.text>[over]	"<<"
	Press left
	"<text> (formatted with snake)"
	" "
state return	"return "
return <user.text>[over]	"return "
	"<text> (formatted with private_camel_case)"
map of string to string	"map[string] string "
map of <user.text>[over]	"map["
	"<text> (formatted with private_camel_case)"
	Press right

comment line <user.text> over	Move cursor to start of line Inserts line comment prefix at current cursor location "<user.text> " ""
comment <user.text> over	Inserts line comment prefix at current cursor location "<user.text> "
comment <user.text>	Inserts line comment prefix at current cursor location "<user.text> "
(line inline) comment <user.text> over	Move cursor to end of line Inserts line comment prefix at current cursor location "<user.text> "
(line inline) comment <user.text>	Move cursor to end of line Inserts line comment prefix at current cursor location "<user.text> "

Table 55: **Data Null**

state (no nil null)	Inserts null
is not (none null)	Inserts check for non-null
is (none null)	Inserts check for null

Table 56: **Functions**

funky <user.text>	Inserts function declaration
pro funky <user.text>	Inserts protected function declaration
pub funky <user.text>	Inserts public function
static funky <user.text>	Inserts private static function
pro static funky <user.text>	Inserts public function
pub static funky <user.text>	Inserts public function
is type { user.code_type }	Inserts a type annotation
returns [type]	{Inserts a return type
user.code_type }	
type { user.code_type }	""

Table 57: **Functions Gui**

toggle funk	Gui
funk <user.code_functions>	Inserts a function and po- sitions the cursor appropri- ately
funk cell <number>	Inserts the selected function when the imgui is open
funk wrap <user.code_functions>	Inserts a function and po- sitions the cursor appropri- ately
funk wrap <number>	Inserts the selected function when the imgui is open

Table 58: **Imperative**

block	Inserts equivalent of {
-------	-------------------------

state if	Inserts if statement
state else if	Inserts else if statement
state else	Inserts else statement
state while	Inserts while statement
state for	Inserts for statement
state for in	Inserts for each equivalent statement
state switch	Inserts switch statement
state case	Inserts case statement
state do	Inserts do statement
state goto	Inserts go-to statement
state return	Inserts return statement
state break	Inserts break statement
state next	Inserts next statement

Table 59: **Libraries**

state import	Import/using equivalent
--------------	-------------------------

Table 60: **Object Oriented**

self dot	Inserts a reference to the current object (e.g., c++ "this" or python's "self") Inserts the object accessor operator (e.g., java's "." or php's "->")
state self	Inserts a reference to the current object (e.g., c++ "this" or python's "self")
state class	Starts a class definition (e.g., java's "class" keyword)

Table 61: **Operators Array**

op subscript	Code_operator_subscript (e.g., c++ [])
--------------	---

Table 62: **Operators Assignment**

op (equals assign)	Code_operator_assignment
op (minus subtract) equals	Code_operator_subtraction_equals
op (plus add) equals	Code_operator_addition_assignment
op (times multiply) equals	Code_operator_multiplication_assignment
op divide equals	Code_operator_division_assignment
op mod equals	Code_operator_modulo_assignment
[op] increment	Code_operator_increment
(op logical bitwise) (ex exclusive) or equals	Code_operator_bitwise_exclusive_or_equals
[(op logical bitwise)] (left right) shift shift left) equals	Code_operator_bitwise_left_right_shift_equals
[(op logical bitwise)] (left right) shift shift right) equals	Code_operator_bitwise_left_right_shift_equals

Table 63: **Operators Bitwise**

[op] bitwise and	Code_operator_bitwise_and
------------------	---------------------------

[op] bitwise or	Code_operator_bitwise_or
(op logical bitwise) (ex exclusive) or	Code_operator_bitwise_exclusive_or
(op logical bitwise) (left shift shift left)	Code_operator_bitwise_left_shift
(op logical bitwise) (right shift shift right)	Code_operator_bitwise_right_shift

Table 64: **Operators Lambda**

op lambda	Code_operator_lambda
-----------	----------------------

Table 65: **Operators Pointer**

op dereference	Code_operator_indirection
op address of	Code_operator_address_of (e.g., c++ & op)
op arrow	Code_operator_structure_dereference (e.g., c++ -> op)

Table 66: **Abbreviate**

(abbreviate abbreviate brief) { user.abbreviation }

Table 67: **Cancel**

cancel cancel	Do nothing
ignore [<phrase>]	Show a desktop notification

Table 68: **Datetimeinsert**

date insert	"the current time, formatted."
date insert UTC	"the current utc time, formatted."
timestamp insert	"the current time, formatted."
timestamp insert high resolution	"the current time, formatted."
timestamp insert UTC	"the current utc time, formatted."
timestamp insert UTC high resolution	"the current utc time, formatted."

Table 69: **Desktops**

desk <number_small>	Change the current desktop
desk next	Move to next desktop
desk last	Move to previous desktop
desk show	Shows the current desktops
window move desk <number>	Move the current window to a different desktop
window move desk left	Move the current window to the desktop to the left
window move desk right	Move the current window to the desktop to the right

Table 70: **Extensions**

{ user.file_extension }	""
-------------------------	----

Table 71: **Formatters**

phrase <user.text>	Inserts a phrase formatted according to formatters. formatters is a comma separated list of formatters (e.g. 'capitalize_all_words,double_quoted_string')
phrase <user.text> over	Inserts a phrase formatted according to formatters. formatters is a comma separated list of formatters (e.g. 'capitalize_all_words,double_quoted_string')
{ user.prose_formatter } <user.prose>	Inserts a phrase formatted according to formatters. formatters is a comma separated list of formatters (e.g. 'capitalize_all_words,double_quoted_string')
{ user.prose_formatter } <user.prose> over	Inserts a phrase formatted according to formatters. formatters is a comma separated list of formatters (e.g. 'capitalize_all_words,double_quoted_string')
<user.format_text> +	Insert a list of strings, sequentially.
<user.format_text> + over	Insert a list of strings, sequentially.
<user.formatters> that	Reformats the current selection.
word <user.word>	Inserts a phrase formatted according to formatters. formatters is a comma separated list of formatters (e.g. 'capitalize_all_words,double_quoted_string')
recent list	Toggles list of recent phrases
recent repeat <number_small>	"the nth most recent phrase"
recent copy <number_small>	Set clipboard text
select that	Selects the last phrase
before that	Moves left before the last phrase
nope that scratch that	Clears the last phrase
nope that <user.formatters>	Clears and reformats last formatted phrase

Table 72: **Git Add Patch**

yank	Press y Press enter
near	Press n Press enter
quench	Press q Press enter
drum	Press d Press enter
air	Press a Press enter

Table 73: **Help**

help alphabet	Provides the symbol dictionary
help symbols	Provides the symbol dictionary
help numbers	Provides the symbol dictionary
help punctuation	Provides the symbol dictionary
help modifier	Provides the symbol dictionary
help special keys	Provides the symbol dictionary
help function keys	Provides the symbol dictionary
help arrows	Provides the symbol dictionary
(help formatters help for-mat format help)	Provides the list of formatter keywords
help context	Display contextual command info
help active	Display contextual command info
help search <user.text>	Display command info for search phrase
help context {user.help_contexts }	{Display command info for selected context}
help help	Display command info for search phrase

Table 74: **Help Open**

help next	Navigates to next page
help previous	Navigates to previous page
help <number>	Select the context by a number
help return	To the main help window
help close	Hides the help

Table 75: **History**

command history	Toggles viewing the history
command history clear	Clear the history
command history less	Show less history
command history more	Show more history

Table 76: **Keys**

go <user.arrow_keys>	Press arrow_keys
<user.letter>	Press letter
(ship uppercase)<user.letters>[(lowercase sunk)]	Inserts a phrase formatted according to formatters. formatters is a comma separated list of formatters (e.g. 'capitalize_all_words,double_quoted_string')
<user.symbol_key>	Press symbol_key
<user.function_key>	Press function_key
<user.special_key>	Press special_key
<user.modifiers>	Press
<user.unmodified_key>	
press <user.modifiers>	Press modifiers

Table 77: **Macro (MacOS)**

macro record	Record a new macro
macro stop	Stop recording
macro play	Player recorded macro

Table 78: **Media**

volume up	Press volup
volume down	Press voldown
set volume <number>	user.media_set_volume(number)
(volume media) mute	Press mute
[media] play next	Press next
[media] play previous	Press prev
media (play pause)	Plays or pauses media

Table 79: **Messaging**

previous (workspace server)	Move to previous workspace/server
next (workspace server)	Move to next workspace/server
channel	Open channel picker
channel <user.text>	Open channel picker " <u><user.text></u> " (formatted with all_lowercase)"
channel up	Move to previous channel
channel down	Move to next channel
([channel] unread last go-preev)	Move to previous unread channel

([channel] unread next go-Moved to next unread channel)	
go (find search)	Open message search
mark (all workspace server) read	Mark this workspace/server as read
mark channel read	Mark this channel as read.
upload file	Upload a file as a message

Table 80: **Microphone Selection**

microphone show	<code>user.microphone_selection_toggle()</code>
microphone pick <num>	Selects a microphone
ber_small>	

Table 81: **Mouse**

control mouse	Toggles control mouse
zoom mouse	Toggles zoom mouse
camera overlay	Toggles camera overlay
run calibration	Start calibration
touch	<code>mouse_click(0)# close the mouse grid if open</code>
righty	<code>mouse_click(1)# close the mouse grid if open</code>
midclick	<code>mouse_click(2)# close the mouse grid if open</code>
<user.modifiers> touch	<code>key("{modifiers}:down")mouse_click(0)key("{modifiers}:up")# close the mouse grid if open</code>
<user.modifiers> righty	<code>key("{modifiers}:down")mouse_click(1)key("{modifiers}:up")# close the mouse grid if open</code>
(dubclick duke)	<code>mouse_click()mouse_click()# close the mouse grid if open</code>
(tripclick triplclick)	<code>mouse_click()mouse_click()mouse_click()# close the mouse grid if open</code>
left drag drag	<code>user.mouse_drag(0)# close the mouse grid if open</code>
right drag righty drag	<code>user.mouse_drag(1)# close the mouse grid if open</code>
end drag drag end	Releases any held mouse buttons
wheel down	Scrolls down
wheel down here	Move the mouse cursor to the center of the currently active window Scrolls down
wheel tiny [down]	Scrolls down
wheel tiny [down] here	Move the mouse cursor to the center of the currently active window Scrolls down
wheel downer	Scrolls down continuously
wheel downer here	Move the mouse cursor to the center of the currently active window Scrolls down continuously
wheel up	Scrolls up
wheel up here	Move the mouse cursor to the center of the currently active window Scrolls up
wheel tiny up	Scrolls up

wheel tiny up here	Move the mouse cursor to the center of the currently active window Scrolls up
wheel upper	Scrolls up continuously
wheel upper here	Move the mouse cursor to the center of the currently active window Scrolls up continuously
wheel gaze	Starts gaze scroll
wheel gaze here	Move the mouse cursor to the center of the currently active window Starts gaze scroll
wheel stop	Stops scrolling
wheel stop here	Move the mouse cursor to the center of the currently active window Stops scrolling
wheel left	Scrolls left
wheel left here	Move the mouse cursor to the center of the currently active window Stops scrolling
wheel right	Scrolls right
wheel right here	Move the mouse cursor to the center of the currently active window Scrolls right
wheel tiny right	Scrolls right
wheel tiny right here	Move the mouse cursor to the center of the currently active window Scrolls right
curse yes	Shows the cursor
curse no	Hides the cursor
copy mouse position	Copy the current mouse position coordinates

Table 82: **Multiple Cursors**

cursor multiple	Enables multi-cursor mode
cursor stop	Disables multi-cursor mode
cursor up	Adds cursor to line above
cursor down	Adds cursor to line below
cursor less	Removes selection & cursor at last occurrence
cursor more	Adds cursor at next occurrence of selection

cursor skip	Skips adding a cursor at next occurrence of selection
cursor all	Adds cursor at every occurrence of selection
cursor lines	Adds cursor at end of every selected line

Table 83: **Repeater**

<user.ordinals>	Repeat the last command n times
<number_small> times	Repeat the last command n times
(repeat that twice)	Repeat the last command n times
repeat that <number_small> times	Repeat the last command n times

Table 84: **Screens**

screen numbers	Show screen number on each screen
----------------	-----------------------------------

Table 85: **Screenshot**

grab screen	Takes a screenshot of the entire screen and saves it to the pictures folder.
grab screen <number_small>	Takes a screenshot of the entire screen and saves it to the pictures folder.
grab window	Takes a screenshot of the active window and saves it to the pictures folder
grab selection	Triggers an application is capable of taking a screenshot of a portion of the screen
grab screen clip	Takes a screenshot of the entire screen and saves it to the clipboard.
grab screen <number_small> clip	Takes a screenshot of the entire screen and saves it to the clipboard.
grab window clip	Takes a screenshot of the active window and saves it to the clipboard

Table 86: **Splits**

split right	Move active tab to right split
split left	Move active tab to left split
split down	Move active tab to lower split
split up	Move active tab to upper split
split (vertically vertical)	Splits window vertically

split (horizontally horizontal)	Splits window horizontally
split flip	Flips the orietation of the active split
split window	Splits the window
split clear	Clears the current split
split clear all	Clears all splits
split next	Goes to next split
split last	Goes to last split
go split <number>	Navigates to a the specified split

Table 87: **Tabs**

tab (open new)	Open a new tab
tab (last previous)	Switch to previous tab for this window
tab next	Switch to next tab for this window
tab close	Closes the current tab
tab (reopen restore)	Re-open the last-closed tab
go tab <number>	Jumps to the specified tab
go tab final	Jumps to the final tab

Table 88: **Talon Helpers**

talon check updates	Check for updates
talon open log	Open talon log
talon open rebel	Open talon repl
talon home	Open talon config folder
talon copy context pie	Adds os-specific context info to the clipboard for the focused app for .py files. assumes you've a module named mod declared.
talon copy context	Adds os-specific context info to the clipboard for the focused app for .talon files
talon copy name	Let <name> be active app's name Set clipboard text
talon copy executable	Let <executable> be active app's executable name Set clipboard text
talon copy bundle	Let <bundle> be active app's bundle identifier Set clipboard text
talon copy title	Let <title> be window title Set clipboard text
talon dump version	Let <result> be talon & operation system verison information Display an object in the log

talon insert version	Let <result> be talon & operation system version information Pastes text and preserves clipboard
talon dump context	Let <result> be active context info Display an object in the log
talon test last	Let <phrase> be the history entry at the specified index Sims the phrase in the active app and dumps to the log
talon test numb <number_small>	Let <phrase> be the history entry at the specified index Sims the phrase in the active app and dumps to the log
talon test <phrase>	Sims the phrase in the active app and dumps to the log
talon debug action user.talon_actions }	{Runs action.find for the provided action and dumps to the log
talon debug list user.talon_lists }	{Dumps the contents of list to the console
talon copy list user.talon_lists }	{Dumps the contents of list to the console
talon debug tags	Dumps the active tags to the console
talon debug modes	Dumps active modes to the console
talon debug scope user.talon_scopes }	{Dumps the active scope information to the console
talon debug setting user.talon_settings }	{Dumps the current setting to the console
talon debug all settings	Dumps all settings to the console
talon debug active app	Let <result> be all active app info to the clipboard Display an object in the log Display an object in the log Display an object in the log
talon copy active app	Let <result> be all active app info to the clipboard Set clipboard text

Table 89: **Websites And Search Engines**

open { user.website }	Visit the given url.
{ user.search_engine } hunt <user.text>	Search a search engine for given text
{ user.search_engine } (that this)	Let <text> be the selected text Search a search engine for given text

Table 90: **Window Management (Windows)**

window (new open)	Open a new window
window next	Switch to next window for this app
window last	Switch to previous window for this app
window close	Close the current window
focus <user.running_applications>name	Focus a new application by name
focus	Open a menu of running apps to switch to
running list	Shows/hides all running applications
launch <user.launch_applications>path (all oses), or appuser_model_id path on windows	Launch a new application by path (all oses), or appuser_model_id path on windows
snap <user.window_snap_position>	Move the active window to a specific position on-screen.
snap next [screen]	Move the active window to a specific screen.
snap last [screen]	Move the active window to the previous screen.
snap screen <number>	Move the active window leftward by one.
snap <user.running_applications>another screen.	Snap a specific application to another screen.
snap <user.window_snap_position>	
snap <user.running_applications>[screen]	Move a specific application to another screen.
<number>	

Table 91: **Dictation Mode**

press <user.keys>	Press
<user.prose>	"<prose> "
new line	"\n"
new paragraph	"\n\n"
cap <user.word>	Let <result> be <word> (formatted with capitalize_first_word) "<result> "
go up <number_small> (line lines)	Move cursor up one row Repeat <number_small>- 1 times
go down <number_small> (line lines)	Move cursor down one row Repeat <number_small>- 1 times
go left <number_small> (word words)	Move cursor left one word Repeat <number_small>- 1 times
go right <number_small> (word words)	Move cursor right one word Repeat <number_small>- 1 times
go line start	Move cursor to start of line
go line end	Move cursor to end of line

select left <number_small> (word words)	Extend selection left one word Repeat <number_small>- 1 times
select right <number_small> (word words)	Extend selection right one word Repeat <number_small>- 1 times
select left <number_small> (character characters)	Extend selection left one column Repeat <number_small>- 1 times
select right <number_small> (character characters)	Extend selection right one column Repeat <number_small>- 1 times
clear left <number_small> (word words)	Extend selection left one word Repeat <number_small>- 1 times Delete selection
clear right <number_small> (word words)	Extend selection right one word Repeat <number_small>- 1 times Delete selection
clear left <number_small> (character characters)	Extend selection left one column Repeat <number_small>- 1 times Delete selection
clear right <number_small> (character characters)	Extend selection right one column Repeat <number_small>- 1 times Delete selection
formatted <user.format_text>	Inserts text as-is, without invoking the dictation formatter.
format selection <user.formatters>	Reformats the current selection.
scratch that	Clears the last phrase
scratch selection	Delete selection
select that	Selects the last phrase
spell that <user.letters>	"<letters> "
spell that <user.formatters> <user.letters>	Let <result> be <letters> (formatted with <formatters>) Inserts text as-is, without invoking the dictation formatter.
escape <user.text>	"<user.text> "

Table 92: **Dragon Modes**

dragon mode	For windows and mac with dragon, disables talon commands and exits dragon's command mode
talon mode	For windows and mac with dragon, enables talon commands and dragon's command mode.

Table 93: **Modes**

dictation mode	Disable a mode Disable a mode Enable a mode Clears the active language mode, and re-enables code.language Disable a mode
command mode	Disable a mode Disable a mode Enable a mode

Table 94: **Sleep Mode Wav2letter**

<phrase>	Do nothing
----------	------------

Table 95: **Find And Replace**

hunt this	Finds text in current editor
hunt this <user.text>	Finds text in current editor
hunt all	Finds text across project
hunt all <user.text>	Finds text across project
hunt case	Toggles find match by case sensitivity
hunt word	Toggles find match by whole words
hunt expression	Toggles find match by regex
hunt next	Navigates to the next occurrence
hunt previous	Navigates to the previous occurrence
replace this [<user.text>]	Search and replace for text in the active editor
replace all	Search and replaces for text in the entire project
replace <user.text> all	Search and replaces for text in the entire project
replace confirm that	Confirm replace at current position
replace confirm all	Confirm replace all
clear last <user.text> [over]	Selects the previous occurrence of the text, and suppresses any find/replace dialogs. Delete selection

clear next <user.text>[over]	Selects the next occurrence of the text, and suppresses any find/replace dialogs. Delete selection	go next clip	Selects the next occurrence of the text, and suppresses any find/replace dialogs. Move cursor right one column
clear last clip	Selects the previous occurrence of the text, and suppresses any find/replace dialogs. Delete selection	paste last <user.text>[over]	Selects the previous occurrence of the text, and suppresses any find/replace dialogs. Move cursor right one column Paste clipboard at cursor
clear next clip	Selects the next occurrence of the text, and suppresses any find/replace dialogs. Delete selection	paste next <user.text>[over]	Selects the next occurrence of the text, and suppresses any find/replace dialogs. Move cursor right one column Paste clipboard at cursor
comment <user.text>[over]	lastSelects the previous occurrence of the text, and suppresses any find/replace dialogs. Toggle comments on the current line(s)	replace <user.text>[over]	lastSelects the previous occurrence of the text, and suppresses any find/replace dialogs. Paste clipboard at cursor
comment last clip	Selects the previous occurrence of the text, and suppresses any find/replace dialogs. Toggle comments on the current line(s)	replace <user.text>[over]	nextSelects the next occurrence of the text, and suppresses any find/replace dialogs. Paste clipboard at cursor
comment <user.text>[over]	nextSelects the next occurrence of the text, and suppresses any find/replace dialogs. Toggle comments on the current line(s)	select last <user.text>[over]	Selects the previous occurrence of the text, and suppresses any find/replace dialogs.
comment next clip	Selects the next occurrence of the text, and suppresses any find/replace dialogs. Toggle comments on the current line(s)	select next <user.text>[over]	Selects the next occurrence of the text, and suppresses any find/replace dialogs.
go last <user.text>[over]	Selects the previous occurrence of the text, and suppresses any find/replace dialogs. Move cursor right one column	select last clip	Selects the previous occurrence of the text, and suppresses any find/replace dialogs.
go last clip	Selects the previous occurrence of the text, and suppresses any find/replace dialogs. Move cursor right one column	select next clip	Selects the next occurrence of the text, and suppresses any find/replace dialogs.
go next <user.text>[over]	Selects the next occurrence of the text, and suppresses any find/replace dialogs. Move cursor right one column		

Table 96: **Homophones**

phones <user.homophones_canonical>	Show the homophones display
phones that	Show the homophones display for the selected text
phones force <user.homophones_canonical>	Show the homophones display forcibly
phones force	Show the homophones display for the selected text forcibly
phones hide	Hides the homophones display

phones word	Select word under cursor Show the homophones display for the selected text
phones [<user.ordinals>] word left	Let <n> be <ordinals> or 1 Moves left by n words. Extend selection left one word Show the homophones display for the selected text
phones [<user.ordinals>] word right	Let <n> be <ordinals> or 1 Moves right by n words. Extend selection right one word Show the homophones display for the selected text

Table 97: **Homophones Open**

choose <number_small>	Let <result> be homophone number <number_small> "<result> " Hides the homophones display
choose <user.formatters> <number_small>	Let <result> be homophone number <number_small> "<result> (formatted with <formatters>)" Hides the homophones display

Table 98: **Line Commands**

lend	Move cursor to end of line
bend	Move cursor to start of line
go <number>	Move cursor to line <n>
go <number> end	Move cursor to line <n> Move cursor to end of line
comment [line] <number>	Selects lines from line_start to line_end Toggle comments on the current line(s)
comment <number> until <number>	Selects lines from line_start to line_end Toggle comments on the current line(s)
clear [line] <number>	Move cursor to line <n> Selects lines from line_start to line_end Delete selection
clear <number> until <number>	Selects lines from line_start to line_end Delete selection
copy [line] <number>	Selects lines from line_start to line_end Copy selection to clipboard

copy <number> until <number>	Selects lines from line_start to line_end Copy selection to clipboard
cut [line] <number>	Selects lines from line_start to line_end Cut selection to clipboard
cut [line] <number> until <number>	Selects lines from line_start to line_end Cut selection to clipboard
(paste replace) <number> until <number>	Selects lines from line_start to line_end Paste clipboard at cursor
(select cell sell) [line] <number>	Selects lines from line_start to line_end
(select cell sell) <number> until <number>	Selects lines from line_start to line_end
tab that	Add a tab stop of indentation
tab [line] <number>	Move cursor to line <n> Add a tab stop of indentation
tab <number> until <number>	Selects lines from line_start to line_end Add a tab stop of indentation
retab that	Remove a tab stop of indentation
retab [line] <number>	Selects lines from line_start to line_end Remove a tab stop of indentation
retab <number> until <number>	Selects lines from line_start to line_end Remove a tab stop of indentation
drag [line] down	Swap the current line with the line below
drag [line] up	Swap the current line with the line above
drag up [line] <number>	Selects lines from line_start to line_end Swap the current line with the line above
drag up <number> until <number>	Selects lines from line_start to line_end Swap the current line with the line above
drag down [line] <number>	Selects lines from line_start to line_end Swap the current line with the line below
drag down <number> until <number>	Selects lines from line_start to line_end Swap the current line with the line below
clone (line that)	Create a new line identical to the current line

Table 99: **Numbers**

<user.number_string>	" "
----------------------	-----

Table 100: **Symbols**

question [mark]	"?"
(downscore underscore)	"_"
double dash	"--"
(bracket brack left bracket)	"{"
(rbrack are bracket right bracket)	"}"
triple quote	"'"
(triple grave triple back tick gravy)	"`"
(dot dot dotdot)	".."
ellipses	"..."
(comma and spamma)	", "
plus	"+"
arrow	"-> "
dub arrow	"=> "
new line	"\n"
carriage return	"\r"
line feed	"\r\n"
empty dubstring	" "" "
	Press left
empty escaped (dubstring dub quotes)	" \" \" \" "
	Press left
	Press left
empty string	" "" "
	Press left
empty escaped string	" \" \" \" \" "
	Press left
	Press left
(inside parens args)	"()"
	Press left
inside (squares square brackets list)	"[]"
	Press left
inside (bracket braces)	"{}"
	Press left
inside percent	"%%"
	Press left
inside (quotes string)	" "" "
	Press left
inside (double quotes dubquotes)	" "" "" "
	Press left
inside (graves back ticks)	" `` "
	Press left
angle that	Let <text> be the selected text Pastes text and preserves clipboard

(square square bracket) that	Let <text> be the selected text Pastes text and preserves clipboard
(bracket brace) that	Let <text> be the selected text Pastes text and preserves clipboard
(parens args) that	Let <text> be the selected text Pastes text and preserves clipboard
percent that	Let <text> be the selected text Pastes text and preserves clipboard
quote that	Let <text> be the selected text Pastes text and preserves clipboard
(double quote dubquote) that	Let <text> be the selected text Pastes text and preserves clipboard
(grave back tick) that	Let <text> be the selected text Pastes text and preserves clipboard

Table 101: **Cheat Sheet**

cursorless help	Toggle cursorless cheat sheet
-----------------	-------------------------------

Table 102: **Cursorless**

{	Perform cursorless simple action
<user.cursorless_simple_action>	
{	Execute multi-target cursorless command
<user.cursorless_swap_targets>	
{	Execute multi-target cursorless command
<user.cursorless_move_bring_targets>	
{	Reformat targets with formatter
<user.formatters> at	
<user.cursorless_target>	
<user.cursorless_wrapper>	Perform cursorless wrap action
<user.cursorless_target>	
cursorless help	Toggle cursorless cheat sheet
cursorless instructions	Open web page with cursorless instructions

Table 103: Talon Docs

replace dock	Insert the docstring for an action
--------------	---------------------------------------

Table 104: Choice Commands

[option]	{Activate a choice available on
user.talon_hud_choices	the screen
}	
option	{Activate multiple choices
user.talon_hud_numerical_choices	available on the screen
}	
+	
option	{Activate multiple choices
user.talon_hud_numerical_choices	available on the screen
}	
+	{Activate a choice available on
user.talon_hud_choices	the screen
}	

Table 105: **Commands**

head up (show open)	Enables the hud
head up theme	{Switches the ui theme
user.talon_hud_themes	
}	
head up (drop stop con-firm)	Starts a setup mode which can change position
head up cancel	Starts a setup mode which can change position
head up (hide close)	Disables the hud
head up (show open)	{Enables a specific hud ele-
user.talon_hud_widget_name	ment
}	
head up (hide close)	{Disables a specific hud ele-
user.talon_hud_widget_name	ment
}	
head up resize	{Starts a setup mode which can change position
user.talon_hud_widget_name	}
}	
head up expand	{Starts a setup mode which can change position
user.talon_hud_widget_name	}
}	
head up text scale	{Starts a setup mode which can change position
user.talon_hud_widget_name	}
}	
head up drag	{Starts a setup mode which can change position for multi-
user.talon_hud_widget_name	ple widgets at the same time
}	
head up basic	{Set a specific widget prefer-
user.talon_hud_widget_name	ence
}	
head up fancy	{Set a specific widget prefer-
user.talon_hud_widget_name	ence
}	

head	up	hide	{Set a specific widget preference
user.talon_hud_widget_name			
}	on sleep		
head	up	show	{Set a specific widget preference
user.talon_hud_widget_name			
}	on sleep		
head	up	align	{Set a specific widget preference
user.talon_hud_widget_name			
}	right		
head	up	align	{Set a specific widget preference
user.talon_hud_widget_name			
}	left		
head	up	align	{Set a specific widget preference
user.talon_hud_widget_name			
}	center		
head	up	align	{Set a specific widget preference
user.talon_hud_widget_name			
}	top		
head	up	align	{Set a specific widget preference
user.talon_hud_widget_name			
}	bottom		
{			Enables a specific hud element
user.talon_hud_widget_name			
}	(show open)		
{			Disables a specific hud element
user.talon_hud_widget_name			
}	(hide close)		
{			Set a specific widget preference
user.talon_hud_widget_name			
}	minimize		
{			Set a specific widget preference
user.talon_hud_widget_name			
}	maximize		
{			Increase the content page of the widget if it has pages
user.talon_hud_widget_name			
}	next		available
{			Decrease the content page of the widget if it has pages
user.talon_hud_widget_name			
}	(back previous)		available
{			Connect the widget to the context menu to show the options
user.talon_hud_widget_name			
}	options		options

Table 106: **Documentation**

```
{
  Show the general documentation
  user.talon_hud_documentation_title
}
```

Table 107: **Toolkit**

toolkit options	Shows the content available in the hud toolkit
toolkit scope	Start debugging the talon scope in the talon hud

toolkit microphones	Show the microphone options in a choice panel
toolkit documentation	Show the general documentation
toolkit walk throughs	Show all the currently available walk through options

Table 108: **Walkthrough**

read more	Increase the content page of the widget if it has pages available
skip step	Skip the current walk through step
skip everything	Skip the current walk through step
previous step	Skip the current walk through step
head up theme dark	Switches the ui theme
head up theme light	Switches the ui theme
music and video playlist	Visit the given url.

Table 109: **Widget Quick Choices**

{	Activate a choice available on the screen
user.talon_hud_quick_choices	
}	

Table 110: **Personal**

[insert] my [first] name	"wen"
[insert] my last name	"kokke"
[insert] my full name	"wen kokke"
[insert] my protonmail	"wen.kokke@pm.me"
[insert] my private proton-mail	"keri.kokke@pm.me"
[insert] my google mail	"wen.kokke@gmail.com"
[insert] my street address	"3-1 12 high street"
[insert] my post code	"g1 1nl"
[insert] my city	"glasgow"
kind regards	"kind regards,"
	Press enter

Table 111: **Haskell**

op (integral integer) divide	"div "
op (fractional real) divide	"/ "
op (integral integer) mod	"mod "
op (integral integer) (power exponent)	"^ "
op (fractional real) (power exponent)	"^^ "
op (floating float) (power exponent)	"** "
op implies	"=> "
op (to arrow)	"-> "

op apply	"\$ "
op map	"<\$> "
has type	":: "
make <user.text> dot	Let <constructor_name> be <text> (formatted with public_camel_case) ""
make <user.text>	Let <constructor_name> be <text> (formatted with public_camel_case) ""
var <user.text> dot	Let <variable_name> be <text> (formatted with private_camel_case) ""
var <user.text>	Let <variable_name> be <text> (formatted with private_camel_case) ""
(deaf define) data [type] <user.text>	Let <type_name> be <text> (formatted with public_camel_case) "" Insert line below cursor "= "
(deaf define) type [alias] <user.text>	Let <type_name> be <text> (formatted with public_camel_case) ""
(deaf define) new type <user.text>	Let <type_name> be <text> (formatted with public_camel_case) ""
(deaf define) (funk function) <user.text>	Let <function_name> be <text> (formatted with private_camel_case) "" Insert line below cursor "" Move cursor up one row Move cursor to end of line
(deaf define) class <user.text>	Let <class_name> be <text> (formatted with public_camel_case) "" Move cursor left one column Repeat 5 times
(deaf define) instance <user.text>	Let <instance_name> be <text> (formatted with public_camel_case) "" Move cursor left one column Repeat 5 times

(deaf define) type familyLet <type_name> be <user.text>	Let <type_name> be <text> (formatted with public_camel_case) "" Move cursor left one column Repeat 2 times
(deaf define) closed type family <user.text>	Let <type_name> be <text> (formatted with public_camel_case) "" Move cursor left one column Repeat 5 times
deriving instanceLet <instance_name> be <user.text> via <user.text>	Let <instance_name> be <text> (formatted with pub- lic_camel_case) ""
deriving instanceLet <instance_name> be <user.text>	Let <instance_name> be <text> (formatted with pub- lic_camel_case) ""
import [<user.text>]	Let <module_name> be <text or "> (formatted with dot_separated,capitalize_all_words) ""
using	"()" Move cursor left one column
hiding	"hiding ()" Move cursor left one column
qualified as [<user.text>]	Let <module_name> be <text or "> (formatted with dot_separated,capitalize_all_words) ""
pragma	"" Move cursor left one column Repeat 3 times
language pragma	"" Move cursor left one column Repeat 3 times

Table 112: **Spotlight (MacOS)**

spot light	Press cmd-space
------------	-----------------

Table 113: **Java**

boxed [type]	{"<user.java_boxed_types> user.java_boxed_types "} Press space
generic [type]	{"<java_generic_data_structures> user.java_generic_data_structures "} Press left
type { user.code_type } ar- ray	"<user.code_type> " Code_operator_subscript (e.g., c++ [])

[state]	{"<user.java_access_modifiers> user.java_access_modifiers" } Press space
[state] { user.java_modifiers	"<user.java_modifiers> " } Press space
op array	Code_operator_subscript (e.g., c++ [])
op new	"new "

Table 114: **Operators Math**

op (minus subtract)	Code_operator_subtraction
op (plus add)	Code_operator_addition
op (times multiply)	Code_operator_multiplication
op divide	Code_operator_division
op mod	Code_operator_modulo
(op (power exponent) to the power [of])	Code_operator_exponent
(op is) equal	Code_operator_equal
(op is) not equal	Code_operator_not_equal
(op is) (greater more)	Code_operator_greater_than
(op is) (less below) [than]	Code_operator_less_than
(op is) greater [than] or equal	Code_operator_greater_than_or_equal
(op is) less [than] or equal	Code_operator_less_than_or_equal
(op logical) and	Code_operator_and
(op logical) or	Code_operator_or

Table 115: **Misc**

(op pad) colon	": "
------------------	------

Table 116: **Talon**

dot talon	".talon"
action block	"action():" Move cursor left one column Move cursor left one column
setting block	"settings():\n "
setting { user.talon_settings	Pastes text and preserves clipboard
win require	"os: windows\n"
mac require	"os: mac\n"
linux require	"os: linux\n"
title require	"win.title: "
application [require][user.talon_apps]	{Let <app> be <talon_apps or "> Pastes text and preserves clipboard
mode require [user.talon_modes]	{Let <mode> be <talon_modes or "> Pastes text and preserves clipboard

tag require [user.talon_tags]	{Let <tag> be <talon_tags or "> Pastes text and preserves clipboard
tag set [{ user.talon_tags }]	Let <tag> be <talon_tags or "> Pastes text and preserves clipboard
list { user.talon_lists }	"
capture user.talon_captures }	{"
key <user.keys> over	"
key <user.modifiers> over	"
funk <user.code_functions>	Inserts a function and positions the cursor appropriately
funk { user.talon_actions }	Inserts a function and positions the cursor appropriately
funk cell <number>	Inserts the selected function when the imgui is open
funk wrap <user.code_functions>	Inserts a function and positions the cursor appropriately
funk wrap <number>	Inserts the selected function when the imgui is open

Table 117: **Vimscript**

assign <user.vimscript_scope> (variable var)	["let " "<vimscript_scope or "> " [Inserts properly formatted private function name
[<user.vimscript_scope> (variable var)	"<vimscript_scope or "> " [Inserts properly formatted private function name
<user.vimscript_functions>	"
state command	"command! "
state end if	"endif"
state end for	"endfor"
state end while	"endwhile"
state end function	"endfunction"
state continue	"continue"

Table 118: **Language Modes**

force see sharp	Sets the active language mode, and disables extension matching
force see plus plus	Sets the active language mode, and disables extension matching
force go (lang language)	Sets the active language mode, and disables extension matching

force java	Sets the active language mode, and disables extension matching
force java script	Sets the active language mode, and disables extension matching
force type script	Sets the active language mode, and disables extension matching
force markdown	Sets the active language mode, and disables extension matching
force python	Sets the active language mode, and disables extension matching
force are language	Sets the active language mode, and disables extension matching
force talon [language]	Sets the active language mode, and disables extension matching
force high school	Sets the active language mode, and disables extension matching
force hen	Sets the active language mode, and disables extension matching
clear language modes	Clears the active language mode, and re-enables code.language
[enable] debug mode	Enable a mode
disable debug mode	Disable a mode

Table 119: **Cheatsheet**

print [html] help	Print out a sheet of talon commands.
print latex help	Print out a sheet of talon commands.

Table 120: **Data Bool**

data true	Insert true value
data false	Insert false value

Table 121: **Eclipse**

please [<user.text>]	Press ctrl-3 "<user.text or "> "
bar explore	Press alt-shift-w-p
bar outline	Press alt-shift-q-o
panel output	Press alt-shift-q Press c
panel problems	Press alt-shift-q Press x

panel errors	Press alt-shift-q Press l
panel breakpoints	Press alt-shift-q Press b
panel search	Press alt-shift-q Press s
panel variables	Press alt-shift-q Press v
show settings	Press alt-w-p
show shortcuts	Press ctrl-shift-l
file hunt [<user.text>]	Press ctrl-shift-r "<text or "> "
file create	Press ctrl-n
file open folder	Press alt-shift-w-x
file rename	Press alt-shift-w-p-enter-f2
file reveal	Press alt-shift-w-p-enter
imports fix	Press ctrl-shift-o
refactor rename	Press alt-shift-r
refactor this	Press alt-shift-i
(go declaration follow)	Press f3
go back	Press alt-left
go forward	Press alt-right
go marks	Press alt-end
toggle mark	Press ctrl-alt-b-down-enter
go next mark	Press alt-pagedown
go last mark	Press alt-pageup
break point	Press ctrl-shift-b
step over	Press f6
debug step into	Press f5
debug step out [of]	Press f7
debug continue	Press f8
copy line down	Press ctrl-alt-down
copy line up	Press ctrl-alt-up

Table 122: **Generic Snippets**

snip { user.snippets }	Inserts a snippet
snip hunt <user.text>	Triggers the program's snippet search
snip hunt	Triggers the program's snippet search
snip create	Triggers snippet creation
snip show	Toggles ui for available snippets

Table 123: **Generic Terminal**

lisa	Lists directories
lisa all	Lists all directories including hidden
katie [<user.text>]	Goes to the provided directory
katie root	Goes to the root of the current drive

katie home	Goes to the provided directory
katie talon user	Goes to the talon user directory
clear screen	Clear screen
run last	Repeats the last command
rerun [<user.text>]	Searches through the previously executed commands
rerun search	Searches through the previously executed commands
kill current	Kills the running command
copy paste	Copy selection to clipboard Paste clipboard at cursor
talon repl	Open the talon repl
talon log	Open the talon log

Table 124: **Visual Studio**

panel solution	Press ctrl-alt-l
panel properties	Press f4
panel output	Press ctrl-alt-o
panel class	Press ctrl-shift-c
panel errors	Press ctrl-\-ctrl-e
panel design	Press shift-f7
panel marks	Press ctrl-k-ctrl-w
panel breakpoints	Press ctrl-alt-b
show settings	Press alt-t-o
fullscreen switch	Press shift-alt-enter
wrap switch	Press ctrl-e-ctrl-w
file hunt [<user.text>]	Press ctrl-shift-t "<text or "> "
file create	Press ctrl-n
file rename	Press ctrl-[s-f2
file reveal	Press ctrl-[s
hint show	Press ctrl-shift-space
definition show	Press f12
definition peek	Press alt-f12
references find	Press shift-f12
format that	Press ctrl-k-ctrl-d
format selection	Press ctrl-k-ctrl-f
imports fix	Press ctrl-r-ctrl-g
refactor field	Press ctrl-r-ctrl-e
refactor interface	Press ctrl-r-ctrl-i
refactor method	Press ctrl-r-ctrl-m
refactor reorder parameters	Press ctrl-r-ctrl-o
refactor remove parameters	Press ctrl-r-ctrl-v
refactor that	Press ctrl-r-ctrl-r
(go declaration follow)	Press ctrl-f12
go back	Press ctrl-
go forward	Press ctrl-shift-
go implementation	Press f12
go recent [<user.text>]	Press ctrl-l-ctrl-r "<text or "> "

go type [<user.text>]	Press ctrl-l-ctrl-t "<text or "> "
go member [<user.text>]	Press alt-\ "<text or "> "
go usage	Press shift-f12
go marks	Press ctrl-k-ctrl-w
toggle mark	Press ctrl-k-ctrl-k
go next mark	Press ctrl-k-ctrl-n
go last mark	Press ctrl-k-ctrl-p
fold toggle	Press ctrl-m-ctrl-m
fold toggle all	Press ctrl-m-ctrl-l
fold definitions	Press ctrl-m-ctrl-o
break point	Press f9
step over	Press f10
debug step into	Press f11
debug step out [of]	Press f10
debug start	Press f5
debug stopper	Press shift-f5
debug continue	Press f5

Table 125: **Vscode**

window reload	Workbench.action.reloadwindow
window close	Workbench.action.closewindow
please [<user.text>]	Workbench.action.showcommands "<user.text or "> "
bar explore	Workbench.view.explorer
bar extensions	Workbench.view.extensions
bar outline	Outline.focus
bar run	Workbench.view.debug
bar search	Workbench.view.search
bar source	Workbench.view.scm
bar switch	Workbench.action.toggleidebarvisibility
symbol hunt [<user.text>]	Workbench.action.gotosymbol "<text or "> "
panel control	Workbench.panel.repl.view.focus
panel output	Workbench.panel.output.focus
panel problems	Workbench.panel.markers.view.focus
panel switch	Workbench.action.togglepanel
panel terminal	Workbench.action.terminal.focus
focus editor	Workbench.action.focusactiveeditorgroup
show settings	Workbench.action.openglobalsettings
show shortcuts	Workbench.action.openglobalkeybindings
show snippets	Workbench.action.opensnippets
centered switch	Workbench.action.togglecenteredlayout
fullscreen switch	Workbench.action.togglefullscreen
theme switch	Workbench.action.selecttheme
wrap switch	Editor.action.togglewordwrap
zen switch	Workbench.action.togglezenmode
file hunt [<user.text>]	Workbench.action.quickopen "<text or "> "
file copy path	Copyfilepath

file create sibling	Execute command via vscode command server, if available, and wait
file create	Workbench.action.files.newuntitledfile
file rename	Fileutils.renamefile
file move	Fileutils.movefile
file open folder	Revealfileinos
file reveal	Workbench.files.action.showactivefile
save ugly	Workbench.action.files.savewithoutfsync
suggest show	Editor.action.triggersuggest
hint show	Editor.action.triggerparameterhints
definition show	Editor.action.revealdefinition
definition peek	Editor.action.peekdefinition
definition side	Editor.action.revealdefinitionaside
references show	Editor.action.gotoreferences
references find	References-view.find
format that	Editor.action.formatdocument
format selection	Editor.action.formatselection
imports fix	Editor.action.organizeimports
problem next	Editor.action.marker.nextinfiles
problem last	Editor.action.marker.prevfiles
problem fix	Problems.action.showquickfixes
rename that	Editor.action.rename
refactor that	Editor.action.refactor
whitespace trim	Editor.action.trimtrailingwhitespace
language switch	Workbench.action.editor.changelang
refactor rename	Editor.action.rename
refactor this	Editor.action.refactor
(go declaration follow)	Editor.action.revealdefinition
go back	Workbench.action.navigateback
go forward	Workbench.action.navigateforward
go implementation	Editor.action.gotoimplementation
go type	Editor.action.gototypedefinition
go usage	References-view.find
go recent [<user.text>]	Workbench.action.openrecent "<text or "> "
go marks	Workbench.view.extension.bookmarks
toggle mark	Bookmarks.toggle
go next mark	Bookmarks.jumptonext
go last mark	Bookmarks.jumptonprevious
fold that	Editor.fold
unfold that	Editor.unfold
fold those	Editor.foldallmarkerregions
unfold those	Editor.unfoldrecursively
fold all	Editor.foldall
unfold all	Editor.unfoldall
fold comments	Editor.foldallblockcomments
git branch	Git.branchfrom
git branch this	Git.branch
git checkout [<user.text>]	Git.checkout "<text or "> "

git commit [<user.text>]	Git.commitstaged Inserts a phrase formatted according to formatters. formatters is a comma separated list of formatters (e.g. 'capitalize_all_words,double_quoted_string')
git commit undo	Git.undocommit
git commit ammend	Git.commitstagedamend
git diff	Git.openchange
git ignore	Git.ignore
git merge	Git.merge
git output	Git.showoutput
git pull	Git.pullrebase
git push	Git.push
git push focus	Git.pushforce
git rebase abort	Git.rebaseabort
git reveal	Git.revealinexplorer
git revert	Git.revertchange
git stash	Git.stash
git stash pop	Git.stashpop
git status	Workbench.scm.focus
git stage	Git.stage
git stage all	Git.stageall
git unstage	Git.unstage
git unstage all	Git.unstageall
pull request	Pr.create
change next	Press alt-f5
change last	Press shift-alt-f5
break point	Editor.debug.action.togglebreakpoint
step over	Workbench.action.debug.stepover
debug step into	Workbench.action.debug.stepinto
debug step out [of]	Workbench.action.debug.stepout
debug start	Workbench.action.debug.start
debug pause	Workbench.action.debug.pause
debug stopper	Workbench.action.debug.stop
debug continue	Workbench.action.debug.continue
debug restart	Workbench.action.debug.restart
debug console	Workbench.action.debug.action.toggleconsole
terminal external	Workbench.action.terminal.openexternalconsole
terminal new	Workbench.action.terminal.new
terminal next	Workbench.action.terminal.focusnext
terminal last	Workbench.action.terminal.focusprevious
terminal split	Workbench.action.terminal.split
terminal zoom	Workbench.action.togglemaximizedpanel
terminal trash	Workbench.action.terminal.kill
terminal toggle	Execute command via vscode command server, if available, and wait
terminal scroll up	Workbench.action.terminal.scrollup
terminal scroll down	Workbench.action.terminal.scrolldown
terminal <number_small>	Activate a terminal by number
copy line down	Editor.action.copylinesdownaction

copy line up	Editor.action.copylinesupaction
select less	Editor.action.smartselect.shrink
select (more this)	Editor.action.smartselect.expand
minimap	Editor.action.toggleminimap
maximize	Workbench.action.minimizeothereditor
restore	Workbench.action.eveneditorwidths
replace here	Search and replace for text in the active editor Press cmd-alt-l
hover show	Editor.action.showhover
join lines	Editor.action.joinlines
full screen	Workbench.action.togglefullscreen
curse undo	Cursorundo
select word	Editor.action.addselectiontonextfind
skip word	Editor.action.moveselectiontonextfind
cell next	Jupyter.gotonextcellinfile
cell last	Jupyter.gotoprevcellinfile
cell run above	Jupyter.runallcellsabove.palette
cell run	Jupyter.runcurrentcell
install local	Workbench.extensions.action.install

Table 126: C

funky <user.text>	Inserts function declaration
static funky <user.text>	Inserts private static function
state include	"#include "
state include system	"#include <> "
	Move cursor left one column
state include local	"#include ""
	Move cursor left one column
state type deaf	"typedef "
state type deaf struct	"typedef struct"
	"{ \n\n } "
	Move cursor up one row
	Press tab
state define	"#define "
state undefine	"#undef "
state if define	"#ifdef "
state pre if	"#if "
state error	"#error "
state pre else if	"#elif "
state pre end	"#endif "
state pragma	"#pragma "
state default	"default:\nbreak;"
push brackets	Move cursor to end of line "{ } "
	Move cursor left one column
	Press enter
	Press enter
	Move cursor up one row
<user.c_variable>	""
<phrase>	"<phrase> (formatted with private_camel_case,no_spaces)"

<user.c_variable>	""
<user.letter>	
cast to <user.c_cast>	""
standard cast to	""
<user.stdint_cast>	
<user.c_types>	""
<user.c_pointers>	""
<user.c_signed>	""
standard	""
<user.stdint_types>	
int main	"int main()"
	Move cursor left one column
toggle includes	Gui
include	Inserts a library and positions the cursor appropriately
<user.code_libraries>	Press end-enter

Table 127: **Javascript**

(op is) strict equal	"=== "
(op is) strict not equal	"!== "
state const	"const "
state let	"let "
state var	"var "
state async	"async "
state await	"await "
state map	".map()"
	Press left
state filter	".filter()"
	Press left
state reduce	".reduce()"
	Press left
state spread	"..."
from import	"from """
	Press left

Table 128: **Python**

dunder in it	"__init__"
state (def deaf deft)	"def "
state try	"try:\n"
state except	"except "
state raise	"raise "
self taught	"self."
pie test	"pytest"
state past	"pass"
raise	{Insert a string. leave the cursor wherever [] is in the text}
user.python_exception	
}	
except	{""
user.python_exception	
}	

dock string	Inserts a document comment and positions the cursor appropriately
dock	{""
user.python_docstring_field	Move cursor left one column
}	
dock type { user.code_type	Insert a string. leave the cursor wherever [] is in the text
}	
dock returns type	{Insert a string. leave the cursor wherever [] is in the text
user.code_type }	
toggle imports	Gui
import	Inserts a library and positions the cursor appropriately
<user.code_libraries>	Press end-enter
from import	"from import "
	Press left
	Move cursor left one word
	Press space
	Move cursor left one column

Table 129: **R**

(op is) in	"%in% "
toggle library	Gui
library	Inserts a library and positions the cursor appropriately
<user.code_libraries>	Press end-enter
(chain pipe that)	Press end
	"%> %"
	Press enter
state na	"na"
function define <user.text>	Inserts private function declaration
named arg	{Inserts a named argument
user.code_parameter_name	
}	

Table 130: **Ruby**

args pipe	" "
	Press left
dock string	Inserts a document comment and positions the cursor appropriately
state end	"end"
state begin	"begin"
state rescue	"rescue "
state module	"module "
instance <user.text>	"@"
	Inserts properly formatted private function name

Table 131: **Typescript**

(op is) strict equal	"=== "
(op is) strict not equal	"!== "
state const	"const "
state let	"let "
state var	"var "
state async	"async "
state await	"await "
state map	".map()" Press left
state filter	".filter()" Press left
state reduce	".reduce()" Press left
state spread	"..."
from import	"from """ Press left

Table 132: **Git**

git add patch	"git add . -p\n"
git add	"git add "
git add everything	"git add -u\n"
git bisect	"git bisect "
git blame	"git blame "
git branch	"git branch "
git branch list	"git branch -all\n"
git remote branches	"git branch -remote\n"
git branch <user.text>	""
git checkout	"git checkout "
git checkout master	"git checkout master\n"
git checkout main	"git checkout main\n"
git checkout <user.text>	""
git cherry pick	"git cherry-pick "
git cherry pick continue	"git cherry-pick -continue "
git cherry pick abort	"git cherry-pick -abort "
git cherry pick skip	"git cherry-pick -skip "
git clone	"git clone "
git clean everything	"git clean -dfx"
git commit message< <user.text>	"git commit\n"
git commit	"git commit\n"
git diff (colour color) words	"git diff -color-words "
git diff	"git diff "
git diff cached	"git diff -cached\n"
git fetch	"git fetch\n"
git fetch all	"git fetch -all\n"
git fetch <user.text>	""
git fetch prune	"git fetch -prune\n"
git in it	"git init\n"
git log all	"git log\n"
git log all changes	"git log -c\n"
git log	"git log "

git log changes	"git log -c "
git merge	"git merge "
git merge <user.text>	""
git move	"git mv "
git pull	"git pull\n"
git pull origin	"git pull origin "
git pull rebase	"git pull -rebase\n"
git pull fast forward	"git pull -ff-only\n"
git pull <user.text>	""
git push	"git push\n"
git push origin	"git push origin "
git push up stream origin	"git push -u origin "
git push <user.text>	""
git push tags	"git push -tags\n"
git rebase	"git rebase\n"
git rebase continue	"git rebase -continue"
git rebase skip	"git rebase -skip"
git remove	"git rm "
git (remove delete) branch	"git branch -d "
git (remove delete) remote	"git push -delete origin "
branch	
git reset	"git reset "
git reset soft	"git reset -soft "
git reset hard	"git reset -hard "
git restore	"git restore "
git restore staged	"git restore -staged "
git restore source	"git restore -source="
git remote list	"git remote -verbose\n"
git remote	"git remote "
git remote add	"git remote add "
git remote add origin	"git remote add origin "
git remote add upstream	"git remote add upstream "
git remote remove	"git remote remove "
git remote show origin	"git remote show origin\n"
git remote rename	"git remote rename "
git show	"git show "
git stash pop	"git stash pop\n"
git stash	"git stash\n"
git stash apply	"git stash apply\n"
git stash list	"git stash list\n"
git stash show	"git stash show"
git status	"git status\n"
git switch [<user.text>]	""
git switch master	"git switch master "
git switch main	"git switch main "
git switch detached	"git switch -detach "
git (switch create new branch) [<user.text>]	""
git switch orphan	"git switch -orphan "
git submodule add	"git submodule add "
git tag	"git tag "
git tag list	"git tag -l\n"
git edit config	"git config -local -e\n"

git clone clipboard	"git clone " Paste clipboard at cursor Press enter
git diff highlighted	Copy selection to clipboard "git diff " Paste clipboard at cursor
git diff clipboard	"git diff " Paste clipboard at cursor
git add highlighted	Copy selection to clipboard "git add " Paste clipboard at cursor
git add clipboard	"git add " Paste clipboard at cursor
git commit highlighted	Copy selection to clipboard "git add " Paste clipboard at cursor "\ngit commit\n"

Table 133: **Wake Up**

welcome back	Enable control mouse, zoom mouse, and disables cursor Enables the history For windows and mac with dragon, enables talon commands and dragon's command mode.
sleep all [<phrase>]	Hides list of running applications Disables the history Hides the homophones display Hides the help Disables control mouse, zoom mouse, and re-enables cursor Disable speech recognition Sleep the engine
drowse [<phrase>]	Disable speech recognition
wake up	Enable speech recognition

Table 134: **Generic Editor**

go word left	Move cursor left one word
go word right	Move cursor right one word
go left	Move cursor left one column
go right	Move cursor right one column
go up	Move cursor up one row
go down	Move cursor down one row
go line start	Move cursor to start of line
go line end	Move cursor to end of line
go way left	Move cursor to start of line Move cursor to start of line
go way right	Move cursor to end of line

go way down	Move cursor to end of file (start of line)
go way up	Move cursor to start of file
go bottom	Move cursor to end of file (start of line)
go top	Move cursor to start of file
go page down	Move cursor down one page
go page up	Move cursor up one page
select line	Select entire line <n>, or current line
select all	Select all text in the current document
select left	Extend selection left one column
select right	Extend selection right one column
select up	Extend selection up one full line
select down	Extend selection down one full line
select word	Select word under cursor
select word left	Extend selection left one word
select word right	Extend selection right one word
select way left	Extend selection to start of line
select way right	Extend selection to end of line
select way up	Extend selection to start of file
select way down	Extend selection to end of file
indent [more]	Add a tab stop of indentation
(indent less out dent)	Remove a tab stop of indentation
clear line	Delete line under cursor
clear left	Press backspace
clear right	Press delete
clear up	Extend selection up one full line Delete selection
clear down	Extend selection down one full line Delete selection
clear word	Delete word under cursor
clear word left	Extend selection left one word Delete selection
clear word right	Extend selection right one word Delete selection
clear way left	Extend selection to start of line Delete selection

clear way right	Extend selection to end of line Delete selection
clear way up	Extend selection to start of file Delete selection
clear way down	Extend selection to end of file Delete selection
clear all	Select all text in the current document Delete selection
copy all	Select all text in the current document Copy selection to clipboard
copy word	Select word under cursor Copy selection to clipboard
copy word left	Extend selection left one word Copy selection to clipboard
copy word right	Extend selection right one word Copy selection to clipboard
copy line	Select entire line <n>, or current line Copy selection to clipboard
cut all	Select all text in the current document Cut selection to clipboard

cut word	Select word under cursor Cut selection to clipboard
cut word left	Extend selection left one word Cut selection to clipboard
cut word right	Extend selection right one word Cut selection to clipboard
cut line	Select entire line <n>, or current line Cut selection to clipboard

Table 135: **Standard**

zoom in	Zoom in
zoom out	Zoom out
scroll up	Move cursor up one page
scroll down	Move cursor down one page
copy (it that)	Copy selection to clipboard
cut (it that)	Cut selection to clipboard
paste (it that)	Paste clipboard at cursor
nope	Undo
redo that	Redo
paste match	Paste clipboard without style information
file save	Save current document
wipe	Press backspace
(pad padding)	"" Press left
slap	Insert line below cursor

List of Tables

1	user.letter	1
2	user.number_key	1
3	user.modifier_key	1
4	user.special_key	1
5	user.symbol_key	1
6	user.arrow_key	1
7	user.punctuation	1
8	user.function_key	1
9	user.formatters	2
10	1password	3
11	1password Global	3
12	Amethyst	3
13	Anaconda	3
14	Discord	3
15	File Manager	3
16	Firefox	4
17	Gdb	4
18	Generic Browser	5
19	Generic Debugger	5
20	I3wm	6
21	Jetbrains	7
22	Kubectl	9
23	Dunst (Linux)	9
24	Keepassx (Linux)	10
25	Signal (Linux)	10
26	Taskwarrior (Linux)	10
27	Termite (Linux)	10
28	Tmux (Linux)	10
29	Finder (MacOS)	11
30	Rstudio (MacOS)	11
31	Apple Terminal (MacOS)	12
32	Explorer (Windows)	12
33	Outlook (Windows)	12
34	Windows Terminal (Windows)	12
35	Ubuntu (Windows)	12
36	Wsl (Windows)	12
37	Slack Mac (MacOS)	12
38	Slack Win (Windows)	13
39	Talon Repl	13
40	Teams	13
41	Teams Mac (MacOS)	14
42	Github	15
43	Gitlab	15
44	Outlook	16
45	Protonmail	16
46	Twitter	17
47	Windbg (Windows)	17
48	Batch	17
49	Go	17
50	Markdown	19
51	Sql	19
52	Comment Block	19
53	Comment Documentation	19
54	Comment Line	19

55	Data Null	20
56	Functions	20
57	Functions Gui	20
58	Imperative	20
59	Libraries	20
60	Object Oriented	20
61	Operators Array	20
62	Operators Assignment	20
63	Operators Bitwise	20
64	Operators Lambda	21
65	Operators Pointer	21
66	Abbreviate	21
67	Cancel	21
68	Datetimeinsert	21
69	Desktops	21
70	Extensions	21
71	Formatters	21
72	Git Add Patch	22
73	Help	22
74	Help Open	22
75	History	22
76	Keys	22
77	Macro (MacOS)	22
78	Media	22
79	Messaging	22
80	Microphone Selection	23
81	Mouse	23
82	Multiple Cursors	23
83	Repeater	24
84	Screens	24
85	Screenshot	24
86	Splits	24
87	Tabs	24
88	Talon Helpers	24
89	Websites And Search Engines	25
90	Window Management (Windows)	25
91	Dictation Mode	25
92	Dragon Modes	26
93	Modes	26
94	Sleep Mode Wav2letter	26
95	Find And Replace	26
96	Homophones	27
97	Homophones Open	28
98	Line Commands	28
99	Numbers	29
100	Symbols	29
101	Cheat Sheet	29
102	Cursorless	29
103	Talon Docs	30
104	Choice Commands	30
105	Commands	30
106	Documentation	30
107	Toolkit	30
108	Walkthrough	31
109	Widget Quick Choices	31
110	Personal	31

111	Haskell	31
112	Spotlight (MacOS)	32
113	Java	32
114	Operators Math	32
115	Misc	32
116	Talon	32
117	Vimscript	33
118	Language Modes	33
119	Cheatsheet	33
120	Data Bool	33
121	Eclipse	33
122	Generic Snippets	34
123	Generic Terminal	34
124	Visual Studio	34
125	Vscode	35
126	C	36
127	Javascript	37
128	Python	37
129	R	37
130	Ruby	37
131	Typescript	38
132	Git	38
133	Wake Up	39
134	Generic Editor	39
135	Standard	40