

Org-Mode Reference Card (1/2)

(for version 9.1.2)

Getting Started

To read the on-line documentation try M-x org-info

Visibility Cycling

rotate current subtree between states TAB
rotate entire buffer between states S-TAB
restore property-dependent startup visibility C-u C-u TAB
show the whole file, including drawers C-u C-u C-u TAB
reveal context around point C-c C-r

Motion

next/previous heading C-c C-n/p
next/previous heading, same level C-c C-f/b
backward to higher level heading C-c C-u
jump to another place in document C-c C-j
previous/next plain list item S-UP/DOWN [2]

Structure Editing

insert new heading/item at current level M-RET
insert new heading after subtree C-RET
insert new TODO entry/checkbox item M-S-RET
insert TODO entry/ckbx after subtree C-S-RET
turn (head)line into item, cycle item type C-c -
turn item/line into headline C-c *
promote/demote heading M-LEFT/RIGHT
promote/demote current subtree M-S-LEFT/RIGHT
move subtree/list item up/down M-UP/DOWN
move the line at point up/down M-S-UP/DOWN
sort subtree/region/plain-list C-c ^
clone a subtree C-c C-x c
copy visible parts of the region C-c C-x v
kill/copy subtree C-c C-x C-w/M-w
yank subtree C-c C-x C-y or C-y
narrow buffer to subtree / widen C-x n s/w

Capture - Refile - Archiving

capture a new item (C-u C-u = goto last) C-c c [1]
refile subtree (C-u C-u = goto last) C-c C-w
archive subtree using the default command C-c C-x C-a
move subtree to archive file C-c C-x C-s
toggle ARCHIVE tag / to ARCHIVE sibling C-c C-x a/A
force cycling of an ARCHIVED tree C-TAB

Filtering and Sparse Trees

construct a sparse tree by various criteria C-c /
view TODO's in sparse tree C-c / t/T
global TODO list in agenda mode C-c a t [1]

Tables

Creating a table

just start typing, e.g. |Name|Phone|Age RET |- TAB
convert region to table C-c |
... separator at least 3 spaces C-3 C-c |

Commands available inside tables

The following commands work when the cursor is inside a table.
Outside of tables, the same keys may have other functionality.

Re-aligning and field motion

re-align the table without moving the cursor C-c C-c
re-align the table, move to next field TAB
move to previous field S-TAB
re-align the table, move to next row RET
move to beginning/end of field M-a/e

Row and column editing

move the current column left M-LEFT/RIGHT
kill the current column M-S-LEFT
insert new column to left of cursor position M-S-RIGHT
move the current row up/down M-UP/DOWN
kill the current row or horizontal line M-S-UP
insert new row above the current row M-S-DOWN
insert hline below (C-u : above) current row C-c -
insert hline and move to line below it C-c RET
sort lines in region C-c ^

Regions

cut/copy/paste rectangular region C-c C-x C-w/M-w/C-y

Miscellaneous

to limit column width to N characters, use ...| <N> |...
edit the current field in a separate window C-c ‘
make current field fully visible C-u TAB
export as tab-separated file M-x org-table-export
import tab-separated file M-x org-table-import
sum numbers in current column/rectangle C-c +

Tables created with the table.el package

insert a new table.el table C-c ~
recognize existing table.el table C-c C-c
convert table (Org-mode ↔ table.el) C-c ~

Spreadsheet

Formulas typed in field are executed by TAB, RET and C-c C-c.
= introduces a column formula, := a field formula.

Example: Add Col1 and Col2 |=\$1+\$2 |
... with printf format specification |=\$1+\$2;%2f|
... with constants from constants.el |=\$1/\$c/\$cm |
sum from 2nd to 3rd hline |:=vsum(@II..@III)|
apply current column formula | = |
set and eval column formula C-c =
set and eval field formula C-u C-c =
re-apply all stored equations to current line C-c *
re-apply all stored equations to entire table C-u C-c *
iterate table to stability C-u C-u C-c *
rotate calculation mark through # * ! ^ _ \$ C-#
show line, column, formula reference C-c ?
toggle grid / debugger C-c }/{

Formula Editor

edit formulas in separate buffer C-c ‘
exit and install new formulas C-c C-c
exit, install, and apply new formulas C-u C-c C-c
abort C-c C-q
toggle reference style C-c C-r
pretty-print Lisp formula TAB
complete Lisp symbol M-TAB
shift reference point S-cursor
shift test line for column references M-up/down
scroll the window showing the table M-S-up/down
toggle table coordinate grid C-c }

Links

globally store link to the current location C-c l [1]
insert a link (TAB completes stored links) C-c C-l
insert file link with file name completion C-u C-c C-1
edit (also hidden part of) link at point C-c C-1
open file links in emacs C-c C-o
...force open in emacs/other window C-u C-c C-o
open link at point mouse-1/2
...force open in emacs/other window mouse-3
record a position in mark ring C-c %
jump back to last followed link(s) C-c &
find next link C-c C-x C-n
find previous link C-c C-x C-p
edit code snippet of file at point C-c ‘
toggle inline display of linked images C-c C-x C-v

Working with Code (Babel)

execute code block at point C-c C-c
open results of code block at point C-c C-o
check code block at point for errors C-c C-v c
insert a header argument with completion C-c C-v j
view expanded body of code block at point C-c C-v v
view information about code block at point C-c C-v I
go to named code block C-c C-v g
go to named result C-c C-v r
go to the head of the current code block C-c C-v u
go to the next code block C-c C-v n
go to the previous code block C-c C-v p
demarcate a code block C-c C-v d
execute the next key sequence in the code C-c C-v x
edit buffer
execute all code blocks in current buffer C-c C-v b
execute all code blocks in current subtree C-c C-v s
tangle code blocks in current file C-c C-v t
tangle code blocks in supplied file C-c C-v f
ingest all code blocks in supplied file into the C-c C-v i
Library of Babel
switch to the session of the current code block C-c C-v z
load the current code block into a session C-c C-v l
view sha1 hash of the current code block C-c C-v a

Completion

In-buffer completion completes TODO keywords at headline
start, TeX macros after “\”, option keywords after “#-”, TAGS
after “:”, and dictionary words elsewhere.
complete word at point M-TAB

Org-Mode Reference Card (2/2)

(for version 9.1.2)

TODO Items and Checkboxes

rotate the state of the current item	C-c C-t
select next/previous state	S-LEFT/RIGHT
select next/previous set	C-S-LEFT/RIGHT
toggle ORDERED property	C-c C-x o
view TODO items in a sparse tree	C-c / t
view 3rd TODO keyword's sparse tree	C-3 C-c / t
set the priority of the current item	C-c , [ABC]
remove priority cookie from current item	C-c , SPC
raise/lower priority of current item	S-UP/DOWN [2]
insert new checkbox item in plain list	M-S-RET
toggle checkbox(es) in region/entry/at point	C-c C-x C-b
toggle checkbox at point	C-c C-c
update checkbox statistics (C-u : whole file)	C-c #

Tags

set tags for current heading	C-c C-q
realign tags in all headings	C-u C-c C-q
create sparse tree with matching tags	C-c \
globally (agenda) match tags at cursor	C-c C-o

Properties and Column View

set property/effort	C-c C-x p/e
special commands in property lines	C-c C-c
next/previous allowed value	S-left/right
turn on column view	C-c C-x C-c
capture columns view in dynamic block	C-c C-x i
quit column view	q
show full value	v
edit value	e
next/previous allowed value	n/p or S-left/right
edit allowed values list	a
make column wider/narrower	> / <
move column left/right	M-left/right
add new column	M-S-right
Delete current column	M-S-left

Timestamps

prompt for date and insert timestamp	C-c .
like C-c . but insert date and time format	C-u C-c .
like C-c . but make stamp inactive	C-c !
insert DEADLINE timestamp	C-c C-d
insert SCHEDULED timestamp	C-c C-s
create sparse tree with all deadlines due	C-c / d
the time between 2 dates in a time range	C-c C-y
change timestamp at cursor ±1 day	S-RIGHT/LEFT [2]
change year/month/day at cursor by ±1	S-UP/DOWN [2]
access the calendar for the current date	C-c >
insert timestamp matching date in calendar	C-c <
access agenda for current date	C-c C-o
select date while prompted	mouse-1/RET
toggle custom format display for dates/times	C-c C-x C-t

Clocking time

start clock on current item	C-c C-x C-i
stop/cancel clock on current item	C-c C-x C-o/x
display total subtree times	C-c C-x C-d
remove displayed times	C-c C-c
insert/update table with clock report	C-c C-x C-r

Agenda Views

add/move current file to front of agenda	C-c [
remove current file from your agenda	C-c]
cycle through agenda file list	C-'
set/remove restriction lock	C-c C-x </>
compile agenda for the current week	C-c a a [1]
compile global TODO list	C-c a t [1]
compile TODO list for specific keyword	C-c a T [1]
match tags, TODO kwds, properties	C-c a m [1]
match only in TODO entries	C-c a M [1]
find stuck projects	C-c a # [1]
configure custom commands	C-c a C [1]
agenda for date at cursor	C-c C-o

Commands available in an agenda buffer

View Org file	
show original location of item	SPC/mouse-3
show and recenter window	L
goto original location in other window	TAB/mouse-2
goto original location, delete other windows	RET
show subtree in indirect buffer, ded. frame	C-c C-x b
toggle follow-mode	F
Change display	
delete other windows	o
view mode dispatcher	v
switch to day/week/month/year/def view	d w vm vy vSP
toggle diary entries / time grid / habits	D / G / K
toggle entry text / clock report	E / R
toggle display of logbook entries	l / v 1/L/c
toggle inclusion of archived trees/files	v a/A
refresh agenda buffer with any changes	r / g
filter with respect to a tag	/
save all org-mode buffers	s
display next/previous day,week,...	f / b
goto today / some date (prompt)	. / j

Remote editing

digit argument	0-9
change state of current TODO item	t
kill item and source	C-k
archive default	\$ / a
refile the subtree	C-c C-w
set/show tags of current headline	: / T
set effort property (prefix=nth)	e
set / compute priority of current item	, / P
raise/lower priority of current item	S-UP/DOWN [2]
run an attachment command	C-c C-a
schedule/set deadline for this item	C-c C-s/d
change timestamp one day earlier/later	S-LEFT/RIGHT [2]
change timestamp to today	>
insert new entry into diary	i

start/stop/cancel the clock on current item	I / 0 / X
jump to running clock entry	J
mark / unmark / execute bulk action	m / u / B
Misc	
follow one or offer all links in current entry	C-c C-o

Calendar commands

find agenda cursor date in calendar	c
compute agenda for calendar cursor date	c
show phases of the moon	M
show sunrise/sunset times	S
show holidays	H
convert date to other calendars	C

Quit and Exit

quit agenda, remove agenda buffer	q
exit agenda, remove all agenda buffers	x

LaTeX and cdlatex-mode

preview LaTeX fragment	C-c C-x C-1
expand abbreviation (cdlatex-mode)	TAB
insert/modify math symbol (cdlatex-mode)	' / '
insert citation using RefTeX	C-c C-x [

Exporting and Publishing

Exporting creates files with extensions .txt and .html in the current directory. Publishing puts the resulting file into some other place.	
export/publish dispatcher	C-c C-e
toggle asynchronous export	C-c C-e C-a
toggle body/visible only export	C-c C-e C-b/v
toggle subtree export	C-c C-e C-s
insert template of export options	C-c C-e #
toggle fixed width for entry or region	C-c :
toggle pretty display of scripts, entities	C-c C-x \

Comments: Text not being exported

Lines starting with # and subtrees starting with COMMENT are never exported.

toggle COMMENT keyword on entry	C-c ;
---------------------------------	-------

Dynamic Blocks

update dynamic block at point	C-c C-x C-u
update all dynamic blocks	C-u C-c C-x C-u

Notes

[1] This is only a suggestion for a binding of this command. Choose your own key as shown under ACTIVATION.

[2] Keybinding affected by org-support-shift-select and also org-replace-disputed-keys.