Windows - Managing and Moving To Other Windows

Operation	Keystroke	<u>e</u>	Function						<u>Not</u>	<u>e</u>			
Window			gement commands are	e bound to C	-x o, C-	x 0, C-x 1	, C-x 2	2 and C-x		_	atives and su	upport for	multiple
<u>Operations</u>	frames. These basi	ic facilities	can be extended by seved by PEL, but not usin	veral built-in	and exter	nal packages:							·
See also:	moving across to	ext with cur	rsor.	J						, , <u>, , , , , , , , , , , , , , , , , </u>			
 ∑ Customize 	_	<u>inner,</u> also built-in, which provides the ability to restore previous window pane layouts. 🖸 PEL activates it when pel-use-winner user option is t. :e-window, 📦 an external package, which extends the C−x o command by displaying Ace target in the windows' upper left corner to quickly se											
 ∑ Key-Chords ∑ Frames 	target window to	o move to a	and possibly operate on	ı.								_ '	
∑ Speedbar		 <u>key-chord</u>, an external package, that PEL use (when <u>pel-use-key-chord</u> user option is t) to activate dual-key chords to move across windows. Windows can be <u>dedicated</u> to specific buffers, for example by <u>Speedbar</u> (see <u>Speedbar</u>). 											
			ame buffers can operat										
	PEL provides extra	a command	s and key bindings:										
			ngs under the <f11> ke cs mode only, the 器 ke</f11>					s and terr	minal mod	les.			
			nu key is mapped to the use operations are available.		Below the	s icon is use	ed to rep	resent the	e Menu ke	y under Wi	ndows.		
	They can a	also be enab	oled in terminal mode, v	with the xtern								n-mouse	mode.
			n be applied to window me is visible at a time th		mes, whe	ther Emacs is	s running	j in graphi	ics mode	or in termin	al mode.		
Open this PDF file.	<f11> w <f1></f1></f11>		(pel-help-pdf &optiona	al OPEN-	Open th	e local copy c	of the ∑	Windows	PDF file	unless a co	mmand pre	fix (like C	− u) was used
See also: <u>∑ Help/</u> Info			WEB-PAGE)		In that c	ase it opens t	he Githu	b-hosted	file instea	d.			
∑ Customize PEL	<f11> w <f2></f2></f11>		(pel-customize-pel &c	optional	Custom	ze PEL Windo	ow supp	ort.					
window control			OTHER-WINDOW)		• If OTH	IER-WINDOW	/ is non-	nil (use C -	-u), disp	lay in other	window.		
∑ Customize Emacs window	<f11> w <f3></f3></f11>		(pel-customize-library OTHER-WINDOW)	y &optional		ze Emacs Wir vindmove.	ndow su	pport gro	ups: wind	ows, ace-w	indow, ace-	window-	display,
control													
PEL Window			ernal package. 🔼 With								l package a	and also	creates a
Management Hydra	1		peed up navigation and <f7> key, then hit one</f7>	•			•	dentified	in the tab	e below.			
,	The keys that are without typing the series of the se		L window hydra are all i efix again.	identified belo	ow with a	<f7> prefix,</f7>	but afte	r typing <	f7> once	e, you can h	it several ot	her wind	ow hydra keys
	While active the	Hydra Hint	is normally shown in th		•							ggle it off	or back on. T
			en the Hydra activates s <f7> key again or use</f7>							oggle it on/	oπ with ?.		
			ow hydra commands are the Function column.								ends with the	e same n	ame as the
	A snapshot of the	window ma	nagement hydra hint m	enu that sho	ws up in	he minibuffer	area as	soon as c	one of its i	ceys is pres			
			buffers that can be dis in the current window.	missed like tl	ne *Help*	buffer. It also	changes	s the buffe	er visible i	n the norma	al windows.	You can	also use b and
	-UUU:F1 *sc:	ratch*	All (4,0) (Lis		on WK Fly	² Anzu ElDo	c)						
	Move	Resize	Split									Other	
	Cupy: up	=: balance	-	Split.	above	Layout	 avout	Close/Bu		Close.	un>: above		un/dedicate
	<up>: up <down>: down <left>: left</left></down></up>	=: balance V: taller v: shorter	3: vertically	C- <up>:</up>	above below left	n: next p: last	layout	0: this k: &kill	window l buffer	C-S-<1	up>: above wn>: below ft>: left		un/dedicate hint cancel
	<down>: down</down>		3: vertically : horizontally 2: horizontally	C- <up>: C-<down>: C-<left>:</left></down></up>	left	n: next	layout with vert.	0: this k: &kill	window l buffer others window	C-S- <doi:< td=""><td>up>: above wm>: below ft>: left ht>: right</td><td></td><td></td></doi:<>	up>: above wm>: below ft>: left ht>: right		
	<pre><down>: down <left>: left</left></down></pre>	V: taller v: shorte: H: wider h: narrowe	3: vertically : horizontally 2: horizontally	C- <up>: C-<down>: C-<left>:</left></down></up>	left	n: next : p: last : x: swap v M-v: flip v	layout with vert.	0: this k: &kill 1: all c q: quit	window l buffer others window buffer	C-S- <doi:< td=""><td>ft>: left</td><td></td><td></td></doi:<>	ft>: left		
	<pre><down>: down <left>: left <right>: right Showing Hydra Him</right></left></down></pre>	V: taller v: shorte: H: wider h: narrowe	3: vertically : horizontally 2: horizontally	C- <up>: C-<down: C-<left>: C-<right>:</right></left></down: </up>	left right	n: next p: last : x: swap v M-v: flip v M-h: flip l	layout with wert. noriz.	0: this k: &kill 1: all c q: quit b: next B: prev	window buffer thers window buffer buffer	C-S-< C-S- <don C-S-<le C-S-<rig < td=""><td>ft>: left ht>: right</td><td></td><td></td></rig <></le </don 	ft>: left ht>: right		
Move point to other	<pre></pre>	V: taller v: shorter H: wider h: narrowe nt w bound to	3: vertically : horizontally 2: horizontally C-x o key provides a	C- <up>: C-<down>: C-<left>: C-<right>:</right></left></down></up>	left right	n: next: p: last: x: swap v M-v: flip v M-h: flip v	layout with vert. noriz. they dif	0: this k: &kill 1: all c q: quit b: next B: prev	window buffer thers window buffer buffer	C-S-< C-S- <don C-S-<le C-S-<rig < td=""><td>ft>: left ht>: right</td><td></td><td></td></rig <></le </don 	ft>: left ht>: right		
Move point to other window	<pre><down>: down <left>: left <right>: right Showing Hydra Him</right></left></down></pre>	V: taller v: shorte: H: wider h: narrowe nt	3: vertically : horizontally 2: horizontally	C- <up>: C-<down>: C-<left>: C-<right>:</right></left></down></up>	right rlapping f Select (r Select	n: next p: last; x: swap v M-v: flip v M-h: flip) eature set but nove point) to	they diff	0: this k: &kill l: all q: quit b: next B: prev fer in their indow. yclic order	window l buffer others window buffer buffer r key assig	C-S-< C-S- <dor C-S-<rig< td=""><td>ft>: left ht>: right</td><td></td><td></td></rig<></dor 	ft>: left ht>: right		
	<pre></pre>	V: taller v: shorte: H: wider h: narrowe nt	3: vertically : horizontally 2: horizontally C-x o key provides a (other-window COUN	C- <up>: C-<down>: C-<left>: C-<right>:</right></left></down></up>	right right Select (r Select With p	n: next: p: last: x: swap v M-v: flip v M-h: flip v eature set but	they different consideration	o: this k: &kill l: all c q: quit b: next B: prev fer in their indow. yclic order der all fran	window l buffer others window buffer buffer key assi ring of wir	c-s- <t C-s-<dor c-s-<le c-s-<ri>gnments. S</ri></le </dor </t 	ft>: left ht>: right lee below.	 d: ?: <f7>:</f7>	hint cancel
window - C-u: swap - C-u C-u: delete	<pre></pre>	V: taller v: shorter H: wider h: narrow	c-x o key provides a (other-window COUN' ALL-FRAMES)	C- <up>: C-<down>: C-<left>: C-<right>:</right></left></down></up>	right clapping f Select (r Selece With r This is	n: next: p: last: x: swap v M-v: flip v M-h: flip v nove point) to	they diffusion to considure the diffusion that considure the diffusion the diffusion the diffusion that considure the diffusion the diffusion that considure the diffusion that consideration the diffusion that considerati	0: this k: &kil! l: all c q: quit b: next B: prev c c c c c c c c c c c c c c c c c c c	window l. buffer bthers window buffer buffer window buffer buffer buffer window buffer buffer window window buffer buffer window	C-S- C-S- <do< p=""> C-S-<do< p=""> C-S-<rig< p=""> gnments. S adows. And PEL'</rig<></do<></do<>	see below.	 d: ?: <f7>:</f7>	hint cancel
window - C-u: swap	<pre></pre>	V: taller v: shorter H: wider h: narrow	3: vertically : horizontally 2: horizontally C-x o key provides a (other-window COUN	C- <up>: C-<down>: C-<left>: C-<right>:</right></left></down></up>	lapping f Select (r Select (r Select and the select in the	n: next: p: last: x: swap v M-v: flip v M-h: flip v M-	they diffusion to the total to	o: this k: skil: 1: all o q: quit b: next B: prev fer in their indow. yclic order der all fran yiour for th e functior e on) wind	window buffer others window buffer buffer wing of wirnes. In the window buffer wing of wing of wing of wing of wing of wing of wing wing wing wing wing wing wing wing	C-S- C-S- <dorder- </dorder- C-S- <le>C-S-<rig< td=""> gnments. S ddows. And PEL' ed by an Aden b</rig<></le>	s default: pext row.	d: ?: <f7>:</f7>	hint cancel
window - C-u: swap - C-u C-u: delete • Move to other window • Move to specified	<pre></pre>	V: taller v: shorter H: wider h: narrow	c-x o key provides a (other-window COUN' ALL-FRAMES)	C- <up>: C-<down>: C-<left>: C-<right>:</right></left></down></up>	lapping f Select (r Select With p This is nil. C	n: next p: last: x: swap v M-v: flip v M-h: flip v move point) to another wind refix argumen t Emacs defau hange it to ac	they difference to the considerate the conside	0: this k: skill l: all o q! quit b: next b: next lindow. yclic order der all fran yiour for the q functior e on) wind v externa	window buffer others window buffer buffer wkey assigned with the second window buffer wkey assigned with the second wkey assigned with the second window select package	C-S- <c-s-<de color="" color<="" of="" td="" the=""><td>s default: pext row.</td><td>d: ?: <f7>:</f7></td><td>hint cancel</td></c-s-<de>	s default: pext row.	d: ?: <f7>:</f7>	hint cancel
window - C-u: swap - C-u C-u: delete • Move to other window • Move to specified window Ace target	<pre></pre>	v: taller v: shorter H: wider h: narrow nt w bound to	c-x o key provides a (other-window COUNALL-FRAMES) (ace-window ARG)	C- <up>: C-<down> C-<left>: C-<right>: Dartially over T & optional</right></left></down></up>	lapping f Select (r Select (r Select (r Select (r Require for the sele	n: next p: last: x: swap n M-v: flip n M-h: flip n move point) to a another winc arefix argument be ange it to acc (and possibly dires the acce- pel-use-acce indow.	they different with they different with they different with the window in consideration to the window operate window e-window e-w	o: this k: skil! l: all o q: quit b: next B: prev fer in their indow. y/clic order der all fran y/our for the e function e on) wind y externa w user op	window l buffer others window buffer buffer wing of winnes. In the window buffer buffe	c-s- <te>c-s-<documents. control="" of="" someone="" t<="" td="" the=""><td>s default: pext row. ce target coownloads, in</td><td>d: ?: <f7>:</f7></td><td>hint cancel</td></documents.></te>	s default: pext row. ce target coownloads, in	d: ?: <f7>:</f7>	hint cancel
window - C-u: swap - C-u C-u: delete • Move to other window • Move to specified window Ace	<pre></pre>	v: taller v: shorter H: wider h: narrow nt w bound to	C-x o key provides a (other-window COUN' ALL-FRAMES) (ace-window ARG) • With only 2 window • With 3 windows or n • Type this number	c- <up>: C-<up>: C-<down> C-<left>: C-<ri>chight>: C-<ri>chight>: C-<ri>chight>: C-<ri>chight>: C-<ri>chight>: C-</ri></ri></ri></ri></ri></left></down></up></up>	lapping f Select (r Select Select With p This is nil. C Move to Requ when th	n: next p: last x: swap M-v: flip M-h: flip M-	they diffusion they diffuse they diffuse they diffuse they diffuse they down to consider the diffuse they down they	o: this k: skill: l: all o q: quit b: next B: prev fer in their indow. yclic order der all fran yiour for th e function e on) wind v externa w user op ' upper le	window buffer others window buffer buffer wing of winnes. In skey. It is a buffer own select I package ption is seen to corner to the corner window buffer window select I package ption is seen to corner to the corner window select to corner to the corner to the corner window seen to corner to the corner to th	C-S- <c-s-<de control="" control<="" of="" td="" the=""><td>s default: pext row. ce target coopynloads, in</td><td>el-use-ac de. nstalls and</td><td>hint cancel</td></c-s-<de>	s default: pext row. ce target coopynloads, in	el-use-ac de. nstalls and	hint cancel
window - C-u: swap - C-u C-u: delete Move to other window Move to specified window Ace target Operate on	<pre></pre>	v: taller v: shorter H: wider h: narrow nt w bound to	c-x o key provides a (other-window COUN' ALL-FRAMES) (ace-window ARG) With only 2 window With 3 windows or n	c- <up>: C-<up>: C-<down> C-<left>: C-<ri>tight>: C-<ri>tight>: C-<ri>tight>: C-</ri></ri></ri></left></down></up></up>	lapping f Select (r Select Select With p This is nil. C Move to Requ when th	n: next p: last x: swap M-v: flip M-h: flip M-	they diffusion they diffuse they diffuse they diffuse they diffuse they down to consider the diffuse they down they	o: this k: skill: l: all o q: quit b: next B: prev fer in their indow. yclic order der all fran yiour for th e function e on) wind v externa w user op ' upper le	window buffer others window buffer buffer wing of winnes. In skey. It is a buffer own select I package ption is seen to corner to the corner window buffer window select I package ption is seen to corner to the corner window select to corner to the corner to the corner window seen to corner to the corner to th	C-S- <c-s-<de control="" control<="" of="" td="" the=""><td>s default: pext row. ce target coopynloads, in</td><td>el-use-ac de. nstalls and</td><td>hint cancel</td></c-s-<de>	s default: pext row. ce target coopynloads, in	el-use-ac de. nstalls and	hint cancel
window - C-u: swap - C-u C-u: delete • Move to other window • Move to specified window Ace target • Operate on specified window	<pre></pre>	v: taller v: shorter H: wider h: narrow nt w bound to	C-x o key provides a (other-window COUN' ALL-FRAMES) (ace-window ARG) • With only 2 window • With 3 windows or n • Type this number • Type one of the fo • x - delete w • m - swap windows windows windows windows or n • Type one of the fo • x - delete w	c- <up>: C-<up>: C-<down>: C-<left>: C-<right>: C-<right>: C- T & optional s, move to the control of the cont</right></right></left></down></up></up>	lapping f Select (r Select Select With p This is nil. C Move to Requ when th	n: next p: last x: swap M-v: flip M-h: flip M-	they diffusion they diffuse they diffuse they diffuse they diffuse they down to consider the diffuse they down they	o: this k: skill: l: all o q: quit b: next B: prev fer in their indow. yclic order der all fran yiour for th e function e on) wind v externa w user op ' upper le	window buffer others window buffer buffer wing of winnes. In skey. It is a buffer own select I package ption is seen to corner to the corner window buffer window select I package ption is seen to corner to the corner window select to corner to the corner to the corner window seen to corner to the corner to th	C-S- <c-s-<de control="" control<="" of="" td="" the=""><td>s default: pext row. ce target coopynloads, in</td><td>el-use-ac de. nstalls and</td><td>hint cancel</td></c-s-<de>	s default: pext row. ce target coopynloads, in	el-use-ac de. nstalls and	hint cancel
window - C-u: swap - C-u C-u: delete • Move to other window • Move to specified window Ace target • Operate on specified window See also: Customize Demo: C'est la Z,	<pre></pre>	v: taller v: shorter H: wider h: narrow nt w bound to	C-x o key provides a (other-window COUN' ALL-FRAMES) (ace-window ARG) • With only 2 window • With 3 windows or n • Type this number • Type one of the fo	c- <up>: C-<up>: C-<down>: C-<left>: C-<right>: C-<right>: C- T & optional T & optional</right></right></left></down></up></up>	lapping f Select (r Select Select With p This is nil. C Move to Requ when th	n: next p: last x: swap M-v: flip M-h: flip M-	they diffusion they diffuse they diffuse they diffuse they diffuse they down to consider the diffuse they down they	o: this k: skill: l: all o q: quit b: next B: prev fer in their indow. yclic order der all fran yiour for th e function e on) wind v externa w user op ' upper le	window buffer others window buffer buffer wing of winnes. In skey. It is a buffer own select I package ption is seen to corner to the corner window buffer window select I package ption is seen to corner to the corner window select to corner to the corner to the corner window seen to corner to the corner to th	C-S- <c-s-<de control="" control<="" of="" td="" the=""><td>s default: pext row. ce target coopynloads, in</td><td>el-use-ac de. nstalls and</td><td>hint cancel</td></c-s-<de>	s default: pext row. ce target coopynloads, in	el-use-ac de. nstalls and	hint cancel
window - C-u: swap - C-u C-u: delete • Move to other window • Move to specified window Ace target • Operate on specified window See also: Customize	<pre></pre>	v: taller v: shorter H: wider h: narrow nt w bound to	c-x o key provides a (other-window COUNALL-FRAMES) With only 2 window With 3 windows or n Type this number Type one of the fo x - delete w m - swap win M - move win c - copy wir j - select bu	C- <up>: C-<down> C-<left>: C-<right>: C-<color o<="" or="" td=""><td>lapping f Select (r Select (r Select (n) Select (n) Requirements (n) Move to Requirements (n) Requirements (</td><td>n: next p: last x: swap M-v: flip M-h: flip M-</td><td>they diffusion they diffuse they diffuse they diffuse they diffuse they down to consider the diffuse they down they</td><td>o: this k: skill: l: all o q: quit b: next B: prev fer in their indow. yclic order der all fran yiour for th e function e on) wind v externa w user op ' upper le</td><td>window buffer others window buffer buffer wing of winnes. In skey. It is a buffer own select I package ption is seen to corner to the corner window buffer window select I package ption is seen to corner to the corner window select to corner to the corner to the corner window seen to corner to the corner to th</td><td>C-S-<c-s-<de control="" control<="" of="" td="" the=""><td>s default: pext row. ce target coopynloads, in</td><td>el-use-ac de. nstalls and</td><td>hint cancel</td></c-s-<de></td></color></right></right></right></right></right></right></right></right></right></right></right></right></right></right></left></down></up>	lapping f Select (r Select (r Select (n) Select (n) Requirements (n) Move to Requirements (n) Requirements (n: next p: last x: swap M-v: flip M-h: flip M-	they diffusion they diffuse they diffuse they diffuse they diffuse they down to consider the diffuse they down they	o: this k: skill: l: all o q: quit b: next B: prev fer in their indow. yclic order der all fran yiour for th e function e on) wind v externa w user op ' upper le	window buffer others window buffer buffer wing of winnes. In skey. It is a buffer own select I package ption is seen to corner to the corner window buffer window select I package ption is seen to corner to the corner window select to corner to the corner to the corner window seen to corner to the corner to th	C-S- <c-s-<de control="" control<="" of="" td="" the=""><td>s default: pext row. ce target coopynloads, in</td><td>el-use-ac de. nstalls and</td><td>hint cancel</td></c-s-<de>	s default: pext row. ce target coopynloads, in	el-use-ac de. nstalls and	hint cancel
window - C-u: swap - C-u C-u: delete • Move to other window • Move to specified window Ace target • Operate on specified window See also: Customize Demo: C'est la Z,	<pre></pre>	v: taller v: shorter H: wider h: narrow nt w bound to	**C-x o key provides a (other-window COUN' ALL-FRAMES) **With only 2 window With 3 windows or n Type this number Type one of the form the	c- <up>: C-<up>: C-<down> C-<left>: C-<right>: C- C- <l>C- C- </l></right></right></right></right></right></right></right></right></right></right></right></right></left></down></up></up>	right Select (r Select (r Select (r Select (r Select (r Require (r Move to r Select (r Selec	n: next p: last x: swap M-v: flip M-h: flip M-	they diffusion they diffuse they diffuse they diffuse they diffuse they down to consider the diffuse they dif	o: this k: skill: l: all o q: quit b: next B: prev fer in their indow. yclic order der all fran yiour for th e function e on) wind v externa w user op ' upper le	window buffer others window buffer buffer wing of winnes. In skey. It is a buffer own select I package ption is seen to corner to the corner window buffer window select I package ption is seen to corner to the corner window select to corner to the corner to the corner window seen to corner to the corner to th	C-S- <c-s-<de control="" control<="" of="" td="" the=""><td>s default: pext row. ce target coopynloads, in</td><td>el-use-ac de. nstalls and</td><td>hint cancel</td></c-s-<de>	s default: pext row. ce target coopynloads, in	el-use-ac de. nstalls and	hint cancel
window - C-u: swap - C-u C-u: delete • Move to other window • Move to specified window Ace target • Operate on specified window See also: Customize Demo: C'est la Z,	<pre></pre>	v: taller v: shorter H: wider h: narrow nt w bound to	c-x o key provides a (other-window COUNALL-FRAMES) with only 2 window With 3 windows or n Type this number Type one of the fo x - delete w m - swap win M - move win c - copy wir j - select bu n - select th	c- <up>c-<up>c-<down>c-<left>c-<left>c-<ri>c-<ri>c-<ri>c-<ri>c-<ri>c-<ri>c-<ri>c-<ri>c-<ri>c-</ri>c-</ri>c-</ri>c-</ri>c-</ri>c-</ri>c-</ri>c-</ri>c-</ri>c-and own and own an</left></left></down></up></up>	I left right Replace to the second of the s	n: next p: last: x: swap M-v: flip M-h: flip M	they diffusion they diffuse they diffuse they diffuse they diffuse they down to consider the diffuse they dif	o: this k: skill: l: all o q: quit b: next B: prev fer in their indow. yclic order der all fran yiour for th e function e on) wind v externa w user op ' upper le	window buffer others window buffer buffer wing of winnes. In skey. It is a buffer own select I package ption is seen to corner to the corner window buffer window select I package ption is seen to corner to the corner window select to corner to the corner to the corner window seen to corner to the corner to th	C-S- <c-s-<de control="" control<="" of="" td="" the=""><td>s default: pext row. ce target coopynloads, in</td><td>el-use-ac de. nstalls and</td><td>hint cancel</td></c-s-<de>	s default: pext row. ce target coopynloads, in	el-use-ac de. nstalls and	hint cancel
window - C-u: swap - C-u C-u: delete • Move to other window • Move to specified window Ace target • Operate on specified window See also: Customize Demo: C'est la Z,	<pre></pre>	v: taller v: shorter H: wider h: narrow nt w bound to	c-x o key provides a (other-window COUN' ALL-FRAMES) With only 2 window With 3 windows or n Type this number Type one of the fo x - delete w m - swap wii M - move wii C - copy wir j - select bu n - select th u - select th v - split wind b - split wind b - split wind	c- <up>: C-<up>: C-<down>: C-<left>: C-<left>: C-<right>: C-<left>: C-<right>: C- T & optional <l< td=""><td>I left right Select (r Select With r Select Move to Requ when th e other w an Ace t ne window her windd her vertic / ally</td><td>n: next p: last: x: swap M-v: flip M-h: flip M</td><td>they diffusion they diffuse they diffuse they diffuse they diffuse they down to consider the diffuse they dif</td><td>o: this k: skill: l: all o q: quit b: next B: prev fer in their indow. yclic order der all fran yiour for th e function e on) wind v externa w user op ' upper le</td><td>window buffer others window buffer buffer wing of winnes. In skey. It is a buffer own select I package ption is seen to corner to the corner window buffer window select I package ption is seen to corner to the corner window select to corner to the corner to the corner window seen to corner to the corner to th</td><td>C-S-<c-s-<de control="" control<="" of="" td="" the=""><td>s default: pext row. ce target coopynloads, in</td><td>el-use-ac de. nstalls and</td><td>hint cancel</td></c-s-<de></td></l<></right></left></right></left></left></down></up></up>	I left right Select (r Select With r Select Move to Requ when th e other w an Ace t ne window her windd her vertic / ally	n: next p: last: x: swap M-v: flip M-h: flip M	they diffusion they diffuse they diffuse they diffuse they diffuse they down to consider the diffuse they dif	o: this k: skill: l: all o q: quit b: next B: prev fer in their indow. yclic order der all fran yiour for th e function e on) wind v externa w user op ' upper le	window buffer others window buffer buffer wing of winnes. In skey. It is a buffer own select I package ption is seen to corner to the corner window buffer window select I package ption is seen to corner to the corner window select to corner to the corner to the corner window seen to corner to the corner to th	C-S- <c-s-<de control="" control<="" of="" td="" the=""><td>s default: pext row. ce target coopynloads, in</td><td>el-use-ac de. nstalls and</td><td>hint cancel</td></c-s-<de>	s default: pext row. ce target coopynloads, in	el-use-ac de. nstalls and	hint cancel
window - C-u: swap - C-u C-u: delete • Move to other window • Move to specified window Ace target • Operate on specified window See also: Customize Demo: C'est la Z,	<pre></pre>	v: taller v: shorter H: wider h: narrow nt w bound to	c-x o key provides a (other-window COUN' ALL-FRAMES) With only 2 window With 3 windows or n Type this number Type one of the fo x - delete w m - swap wii M move wii n - select bu n - select th u - select th v - split wind v - spli	c- <up>: C-<up>: C-<down> C-<down> C-<left> C-<ri>child > C-<ri>child > C-<ri>child > C-<ri>child > C-T & optionalT &</ri></ri></ri></ri></left></down></down></up></up>	I left right Select (r Select Select With p This is nil. C Move to Requ when th e other w an Ace t ne window her window her window her vertic d ally dow	n: next p: last x: swap y M-v: flip y M-h:	they diffusion they diffuse they diffuse they diffuse they diffuse they down to consider the diffuse they dif	o: this k: skill: l: all o q: quit b: next B: prev fer in their indow. yclic order der all fran yiour for th e function e on) wind v externa w user op ' upper le	window buffer others window buffer buffer wing of winnes. In skey. It is a buffer own select I package ption is seen to corner to the corner window buffer window select I package ption is seen to corner to the corner window select to corner to the corner to the corner window seen to corner to the corner to th	C-S- <c-s-<de control="" control<="" of="" td="" the=""><td>s default: pext row. ce target coopynloads, in</td><td>el-use-ac de. nstalls and</td><td>hint cancel</td></c-s-<de>	s default: pext row. ce target coopynloads, in	el-use-ac de. nstalls and	hint cancel
window - C-u: swap - C-u C-u: delete • Move to other window • Move to specified window Ace target • Operate on specified window See also: Customize Demo: C'est la Z,	<pre></pre>	v: taller v: shorter H: wider h: narrow nt w bound to	** vertically : horizontally 2: horizontally 2: horizontally 2: horizontally 2: horizontally 2: horizontally 2: horizontally 3: horizontally 2: horizontally 3: horizontally 4: horizontally 3: horizontally 3: horizontally 3: horizontally 4: horizontally 4	c- <up>c-<down>c-<down>c-<down>c-<down>c-<down>c-<down>c-<down-c- </down-c- c-<ri>ght>c-<ri>ght>c-c-<ri>ght>c-c-c-c-c- c-<br< td=""><td>I left right Select (r Select Select Hermin Select Hermin</td><td>n: next p: last x: swap w M-v: flip w M-v: flip w M-h: flip w M-h:</td><td>they diffusion they diffuse they diffuse they down in considult behaviorizate they operate windows e-windows et numb</td><td>o: this k: skil! l: all o q: quit b: next l: en or fer in their indow. /clic order der all fran /iour for the function e on) wind vexterna w user o ' upper le er to move cs and tel</td><td>window buffer others window buffer bu</td><td>c-s-<te>c-s-<de>c-s-<de>c-s-<de>c-s-<de>c-s-<ide>c-s-<re>c-s-<re>c-s-<re>c-s-<re>c-s-</re></re></re></re></ide></de></de></de></de></te><te>c-s-<re>c-s-</re></te><te>c-s-</te><te>c-s-</te><te>c-s-</te><te>c-s-</te><te>c-s-</te><te>c-s-</te><te>c-s-</te><te>c-s-</te><te>c-s-</te><te>c-s-</te><te>c-s-</te><te>c-s-</te><te>c-s-</te><te>c-s-</te><te>c-s-</te><te>c-s-</te><te>c-s-</te><te>c-s-</te><te>c-s-</te><te>c-s-</te><te>c-s-</te><te>c-s-</te><te>c-s-</te><te>c-s-</te><te>c-s-</te><te>c-s-</te><te>c-s-</te><te>c-s-</te><te>c-s-</te><te>c-s-</te><te>c-s-</te><te>c-s-</te><te>c-s-</te><te>c-s-</te><te>c-s-</te><te>c-s-</te><te>c-s-</te><te>c-s-</te><te>c-s-</te><te>c-s-</te><te>c-s-</te><te>c-s-</te><te>c-s-</te><te>c-s-</te><te>c-s-</te><te>c-s-</te><te>c-s-</te><te>c-s-</te><te>c-s-</te><te>c-s-</te><te>c-s-</te><te>c-s-</te><te>c-s-</te><te>c-s-</te><te>c-s-</te><te>c-s-</te><te>c-s-</te><te>c-s-</te><te>c-s-</te><te>c-s-</te><te>c-s-</te><te>c-s-</te><te>c-s-</te><te>c-s-</te><te>c-s-</te><te>c-s-</te><te>c-s-</te><te>c-s-</te><te>c-s-</te><te>c-s-</te><te>c-s-</te><te>c-s-</te><te>c-s-</te><te>c-s-</te><te>c-s-</te><te>c-s-</te><te>c-s-</te><te>c-s-</te><te>c-s-</te><te>c-s-</te><te>c-s-</te><te>c-s-</te><te>c-s-</te><te>c-s-</te><te>c-s-</te><te>c-s-</te><te>c-s-</te><te>c-s-</te><te>c-s-</te><te>c-s-</te><te>c-s-</te><te>c-s-</te><te>c-s-</te><te>c-s-</te><te>c-s-</te><te>c-s-</te><te>c-s-</te><te>c-s-</te><te>c-s-</te><te>c-s-</te><te>c-s-</te><te>c-s-</te><te>c-s-</te><te>c-s-</te><te>c-s-</te><te>c-s-</te><te>c-s-</te><te>c-s-</te><te>c-s-</te><te>c-s-</te><te>c-s-</te><te>c-s-</te><te>c-s-</te><te>c-s-</te><te>c-s-</te><te>c-s-</te><te>c-s-</te><te>c-s-</te><te>c-s-</te><te>c-s-</te><te>c-s-</te><te>c-s-</te><te>c-s-</te><te>c-s-</te><te>c-s-</te><te>c-s-</te><te>c-s-</te><te>c-s-</te><te>c-s-</te><te>c-s-</te><te>c-s-</te><te>c-s-</te><te>c-s-</te><te>c-s-</te><te>c-s-</te><te>c-s-</te><te>c-s-</te><te>c-s-</te><te>c-s-</te><te>c-s-</te><te>c-s-</te><te>c-s-</te><te>c-s-</te><te>c-s-</te><te>c-s-</te><te>c-s-</te><te>c-s-</te><te>c-s-</te><te>c-s-<td>s default: pext row. ce target coopynloads, in</td><td>el-use-ac de. nstalls and</td><td>hint cancel</td></te></td></br<></br></br></br></br></br></br></br></br></br></br></br></ri></ri></ri></down></down></down></down></down></down></up>	I left right Select (r Select Select Hermin	n: next p: last x: swap w M-v: flip w M-v: flip w M-h:	they diffusion they diffuse they diffuse they down in considult behaviorizate they operate windows e-windows et numb	o: this k: skil! l: all o q: quit b: next l: en or fer in their indow. /clic order der all fran /iour for the function e on) wind vexterna w user o ' upper le er to move cs and tel	window buffer others window buffer bu	c-s- <te>c-s-<de>c-s-<de>c-s-<de>c-s-<de>c-s-<ide>c-s-<re>c-s-<re>c-s-<re>c-s-<re>c-s-</re></re></re></re></ide></de></de></de></de></te> <te>c-s-<re>c-s-</re></te> <te>c-s-</te> <te>c-s-<td>s default: pext row. ce target coopynloads, in</td><td>el-use-ac de. nstalls and</td><td>hint cancel</td></te>	s default: pext row. ce target coopynloads, in	el-use-ac de. nstalls and	hint cancel
window - C-u: swap - C-u C-u: delete • Move to other window • Move to specified window Ace target • Operate on specified window See also: Customize Demo: C'est la Z,	<pre></pre>	v: taller v: shorter H: wider h: narrow nt w bound to	** vertically : horizontally 2: horizontally 2: horizontally 2: horizontally 2: horizontally 2: horizontally 2: horizontally 3: horizontally 4: horizontally 4	partially over C- <ref>coleft> C-<ref>coleft> C-<ref>coleft> C-<ref>coleft> coleft> col</ref></ref></ref></ref>	right Select (r Select (r Select (r Select (r Select (r Select (r Required (r Move to r Move to r Required (r Move to r Move to r Move to r Required (r Move to r Move to r Move to r Required (r Move to r Move to r Move to r Required (r Move to r Move to r Move to r Required (r Move to r Move to r Move to r Required (r Move to r Move to r Move to r Required (r Move to r Move to r Move to r Required (r Move to r Move to r Move to r Required (r Move	n: next p: last x: swap w M-v: flip w M-h: flip l M-h:	they diffusion they down in considute the window in considute they window e-window windows et numb	o: this k: skil: 1: all o q: quit b: next b: next cerin their indow. yclic order der all fran yiour for th e functior e on) wind vexterna w user op 'upper le er to move cs and ten dow.	window buffer others window buffer buffer wing of wirnnes. In package of the total package o	c-s- <cccccccccccccccccccccccccccccccccc< td=""><td>iee below. s default: pext row. ce target coownloads, ir</td><td>el-use-acide. stalls and the on it:</td><td>hint cancel de-window =</td></cccccccccccccccccccccccccccccccccc<>	iee below. s default: pext row. ce target coownloads, ir	el-use-acide. stalls and the on it:	hint cancel de-window =
window - C-u: swap - C-u C-u: delete • Move to other window • Move to specified window Ace target • Operate on specified window See also: Customize Demo: C'est la Z,	<pre></pre>	v: taller v: shorter H: wider h: narrow nt w bound to	**Select but no split wine v - show the split wine v - show the split wine v - sp	partially over C- c-c	I left right Select (r Select With y This is nil. C Move to Requivement he window her window her window her vertic / ally dow d binding in other macs fram macs fram form more	n: next p: last: x: swap M-v: flip M-h: flip M-h: flip move point) to a another winc perfix argumet Emacs defat hange it to ac (and possibly tires the ace- pel-use-ace indow. arget in the w y, or d by the target w ally or horizor arget in othe the sare in othe the sare in othe the sare in othe the sare in othe the performance operations:	they diffusion they diffuse they diffuse they diffuse they down to consider the consideration that they diffuse they down the consideration that they down t	o: this k: skill l: all o g: quit b: next B: prev fer in their indow. yclic order der all fran yiour for the e function e on) wind w externa w user op ' upper le er to move cs and tendow. y occupy	window buffer others window wi	C-S-< C-S- <do c-s-<do="" c-s-<le="">C-S-<ri>C-S-</ri></do> dows. And PEL' cribed in ne ed by an Ac to to t. it identifies arget windows de) same OS v	dee below. Is default: pext row. Is the window and operation of the window and operation of the window and operation.	d: ?: <fr>: del-use-acc de. destalls and w target. ate on it:</fr>	hint cancel e-window = d activates it
window - C-u: swap - C-u C-u: delete • Move to other window • Move to specified window Ace target • Operate on specified window See also: Customize Demo: C'est la Z,	<pre></pre>	v: taller v: shorter H: wider h: narrow nt w bound to	**Silvertically : horizontally : comparison of the foliation of the fol	partially over C- c-c	I left right Select (r Select With p This is nil. C Move to Requ when th e other w an Ace t ne window her window her window her vertic / ally dow d binding in other nacs fram macs fram form more mpt, use	n: next p: last: x: swap y M-v: flip y M-v: flip y M-h: flip y M-h	they diffusion they diffuse they diffuse they diffuse they down to consider the consideration that they diffuse they down the consideration that they down t	o: this k: skill l: all o g: quit b: next B: prev fer in their indow. yclic order der all fran yiour for the e function e on) wind w externa w user op ' upper le er to move cs and tendow. y occupy	window buffer others window wi	C-S-< C-S- <do c-s-<do="" c-s-<le="">C-S-<ri>C-S-</ri></do> dows. And PEL' cribed in ne ed by an Ac to to t. it identifies arget windows de) same OS v	dee below. Is default: pext row. Is the window and operation of the window and operation of the window and operation.	d: ?: <fr>: del-use-acc de. destalls and w target. ate on it:</fr>	hint cancel e-window = d activates it
window - C-u: swap - C-u C-u: delete • Move to other window • Move to specified window Ace target • Operate on specified window See also: Customize Demo: C'est la Z,	<pre></pre>	v: taller v: shorter H: wider h: narrow nt w bound to	** Vertically : horizontally 2: horizontally 2: horizontally 2: horizontally 2: horizontally 2: horizontally 2: horizontally 3: horizontally 2: horizontally 3: horizontally 3	partially over C- c-c	I left right Select (r Select (r Select (r Select (r Required to	n: next p: last p: last x: swap w N-v: flip w N-h: fli	they diffusion in graphing of the rumb	o: this k: skil: 1: all o q: quit b: next b: next list prev fer in their indow. y/clic order der all fran y/our for the function e on) wind vexterna w user o y upper le er to move cs and ten dow. y occupy acluding ju indow and	window l buffer others window buffer buffer buffer wing of winnes. In a little window buffer	c-s- <c c-s-<="" c-s-<doc="" do=""></c>	dee below. Is default: peat row. It to the window and operation window): just window): window): just window): ju	el-use-acde. at one Emittin sever selected	e-window = d activates it acs frame is al frames buffer moves
window - C-u: swap - C-u C-u: delete • Move to other window • Move to specified window Ace target • Operate on specified window See also: Customize Demo: C'est la Z,	<pre></pre>	v: taller v: shorter H: wider h: narrow nt w bound to	**Service ally : horizontally : horizontally : horizontally 2: horizontally 3:	partially over C- <re> C-<up>C-<down> C-<left> C-<re> C- I down in the control of the contro</re></left></down></up></re>	I left right Select (r Select With r This is nil. C Move to Requ when th e other w an Ace t ane window for y, followed d binding in other nacs fram form more mpt, use windows swap be be buffer m windows.	n: next p: last p: last x: swap M-v: flip M-h:	they diffusion they diffuse they diffuse they diffuse they down to considult behavior to the considuation they diffuse the	o: this k: skill: 1: all o q! quit b: next b: next b: next corder in their indow. yclic order der all fran yiour for the q function e on) wind v externa w user o v u	window buffer others window buffer buffer wing of wirnes. It is to the tall package ption is seen to the tal	c-s- <c c-s-<="" c-s-<doc="" do=""></c>	dee below. Is default: peat row. It to the window and operation window): just window): window): just window): ju	el-use-acde. at one Emittin sever selected	e-window = d activates it acs frame is al frames buffer moves
window - C-u: swap - C-u C-u: delete • Move to other window • Move to specified window Ace target • Operate on specified window See also: Customize Demo: C'est la Z,	<pre></pre>	V: taller v: shorter H: wider h: narrow M bound to	**Supervised Supervised Supervise	partially over s, move to the control of the contr	I left right Select (r Select With p This is nil. C Move to Requivement he windows, follower an Ace the windows, follower and the windows, form more macs frammer, use windows is swap be the buffer mindows.	n: next p: last: x: swap y M-v: flip y M-v	they diffusion they distribute they diffuse they diffuse they down in control to the considual to the control t	o: this k: skill l: all o g: quit b: next B: prev fer in their indow. yclic order der all fran yiour for the e function e on) wind w externa w user op ' upper le er to move cs and ten dow. y occupy cultuding ju indow and ow). The	window buffer others window buffer bu	c-s- <ce>c-s-<de>c-s-<de>c-s-<de>c-s-<de>c-s-<de>c-s-<le>c-s-<ie>c-s-<re>c-s-<re>c-s-<re>c-s- and OPEL's aribed in ne ed by an Ac arget window de) same OS v c alone ent window 1> w x ke</re></re></re></ie></le></de></de></de></de></de></ce>	dee below. Is default: peat row. Dee target cooperate window and operate window and operate window. Window): just window is the window and operate window and operate window and operate window.	de. nstalls and tone Email tith sever selected same (but	e-window = d activates it acs frame is al frames buffer moves
window - C-u: swap - C-u C-u: delete • Move to other window • Move to specified window Ace target • Operate on specified window See also: Customize Demo: C'est la Z,	<pre></pre>	V: taller v: shorter H: wider h: narrow M bound to	**Service ally : horizontally : horizontally : horizontally 2: horizontally 3:	partially over C- <re> C-<up>C-<down> C-<left> C-<re> C- I down in the control of the contro</re></left></down></up></re>	right Select (r	n: next p: last p: last x: swap M-v: flip M-h:	they diffusion they diffuse they diffuse they diffuse they down to considult behavious they distribute they operate window windows the trumb they diffuse they di	o: this k: skill: 1: all o g! quit b: next b: next b: next corder of the inindow. yclic order der all fran yiour for the e function e on) wind vexterna w user o or vexterna v	window buffer others window buffer buffer wing of wirnes. In a buffer own select a package ption is set to the tall the exact with the exact set typing of the currence of the	c-s- <ce>c-s-<de>c-s-<de>c-s-<de>c-s-<de>c-s-<ee>c-s-<ee>c-s-<ee>c-s-<ee>c-s-<ee>c-s-<ee>c-s-<ee>c-s-<ee>c-s-<ee>c-s-<ee>c-s-<ee>c-s-<ee>c-s-<ee>c-s-<ee>c-s-<ee>c-s-<ee>c-s-<ee>c-s-<ee>c-s-<ee>c-s-<ee>c-s-<ee>c-s-<ee>c-s-<ee>c-s-<ee>c-s-<ee>c-s-<ee>c-s-<ee>c-s-<ee>c-s-<ee>c-s-<ee>c-s-<ee>c-s-<ee>c-s-<ee>c-s-<ee>c-s-<ee>c-s-<ee>c-s-<ee>c-s-<ee>c-s-<ee>c-s-<ee>c-s-<ee>c-s-<ee>c-s-<ee>c-s-<ee>c-s-<ee>c-s-<ee>c-s-<ee>c-s-<ee>c-s-<ee>c-s-<ee>c-s-<ee>c-s-<ee>c-s-<ee>c-s-<ee>c-s-<ee>c-s-<ee>c-s-<ee>c-s-<ee>c-s-<ee>c-s-<ee>c-s-<ee>c-s-<ee>c-s-<ee>c-s-<ee>c-s-<ee>c-s-<ee>c-s-<ee>c-s-<ee>c-s-<ee>c-s-<ee>c-s-<ee>c-s-<ee>c-s-<ee>c-s-<ee>c-s-<ee>c-s-<ee>c-s-<ee>c-s-<ee>c-s-<ee>c-s-<ee>c-s-<ee>c-s-<ee>c-s-<ee>c-s-<ee>c-s-<ee>c-s-<ee>c-s-<ee>c-s-<ee>c-s-<ee>c-s-<ee>c-s-<ee>c-s-<ee>c-s-<ee>c-s-<ee>c-s-<ee>c-s-<ee>c-s-<ee>c-s-<ee>c-s-<ee>c-s-<ee>c-s-<ee>c-s-<ee>c-s-<ee>c-s-<ee>c-s-<ee>c-s-<ee>c-s-<ee>c-s-<ee>c-s-<ee>c-s-<ee>c-s-<ee>c-s-<ee>c-s-<ee>c-s-<ee>c-s-<ee>c-s-<ee>c-s-<ee>c-s-<ee>c-s-<ee>c-s-<ee>c-s-<ee>c-s-<ee>c-s-<ee>c-s-<ee>c-s-<ee>c-s-<ee>c-s-<ee>c-s-<ee>c-s-<ee>c-s-<ee>c-s-<ee>c-s-<ee>c-s-<ee>c-s-<ee>c-s-<ee>c-s-<ee>c-s-<ee>c-s-<ee>c-s-<ee>c-s-<ee>c-s-<ee>c-s-<ee>c-s-<ee>c-s-<ee>c-s-<ee>c-s-<ee>c-s-<ee>c-s-<ee>c-s-<ee>c-s-<ee>c-s-<ee>c-s-<ee>c-s-<ee>c-s-<ee>c-s-<ee>c-s-<ee>c-s-<ee>c-s-<ee>c-s-<ee>c-s-<ee>c-s-<ee>c-s-<ee>c-s-<ee>c-s-<ee>c-s-<ee>c-s-<ee>c-s-<ee>c-s-<ee>c-s-<ee>c-s-<ee>c-s-<ee>c-s-<ee>c-s-<ee>c-s-<ee>c-s-<ee>c-s-<ee>c-s-<ee>c-s-<ee>c-s-<ee>c-s-<ee>c-s-<ee>c-s-<ee>c-s-<ee>c-s-<ee>c-s-<ee>c-s-<ee>c-s-<ee>c-s-<ee>c-s-<ee>c-s-<ee>c-s-<ee>c-s-<ee>c-s-<ee>c-s-<ee>c-s-<ee>c-s-<ee>c-s-<ee>c-s-<ee>c-s-<ee>c-s-<ee>c-s-<ee>c-s-<ee>c-s-<ee>c-s-<ee>c-s-<ee>c-s-<ee>c-s-<ee>c-s-<ee>c-s-<ee>c-s-<ee>c-s-<ee>c-s-<ee>c-s-<ee>c-s-<ee>c-s-<ee>c-s-<ee>c-s-<ee>c-s-<ee>c-s-<ee>c-s-<ee>c-s-<ee>c-s-<ee>c-s-<ee>c-s-<ee>c-s-<ee>c-s-<ee>c-s-<ee>c-s-<ee>c-s-<ee>c-s-<ee>c-s-<ee>c-s-<ee>c-s-<ee>c-s-<ee>c-s-<ee>c-s-<ee>c-s-<ee>c-s-<ee>c-s-<ee>c-s-<ee>c-s-<ee>c-s-<ee>c-s-<ee>c-s-<ee>c-s-<ee>c-s-<ee>c-s-<ee>c-s-<ee>c-s-<ee>c-s-<ee>c-s-<e< td=""><td>dee below. Is default: per strow. It is the window and operation of the window and operation window): just the window are the window and operation of the window and operation operation of the window and operation operation operation operation of the window and operation operation</td><td>de. nstalls and tone Email tith sever selected same (but</td><td>e-window = d activates it acs frame is al frames buffer moves</td></e<></ee></ee></ee></ee></ee></ee></ee></ee></ee></ee></ee></ee></ee></ee></ee></ee></ee></ee></ee></ee></ee></ee></ee></ee></ee></ee></ee></ee></ee></ee></ee></ee></ee></ee></ee></ee></ee></ee></ee></ee></ee></ee></ee></ee></ee></ee></ee></ee></ee></ee></ee></ee></ee></ee></ee></ee></ee></ee></ee></ee></ee></ee></ee></ee></ee></ee></ee></ee></ee></ee></ee></ee></ee></ee></ee></ee></ee></ee></ee></ee></ee></ee></ee></ee></ee></ee></ee></ee></ee></ee></ee></ee></ee></ee></ee></ee></ee></ee></ee></ee></ee></ee></ee></ee></ee></ee></ee></ee></ee></ee></ee></ee></ee></ee></ee></ee></ee></ee></ee></ee></ee></ee></ee></ee></ee></ee></ee></ee></ee></ee></ee></ee></ee></ee></ee></ee></ee></ee></ee></ee></ee></ee></ee></ee></ee></ee></ee></ee></ee></ee></ee></ee></ee></ee></ee></ee></ee></ee></ee></ee></ee></ee></ee></ee></ee></ee></ee></ee></ee></ee></ee></ee></ee></ee></ee></ee></ee></ee></ee></ee></ee></ee></ee></ee></ee></ee></ee></ee></ee></ee></ee></ee></ee></ee></ee></ee></ee></ee></ee></ee></ee></ee></ee></ee></ee></ee></ee></ee></ee></ee></ee></ee></ee></ee></ee></ee></ee></ee></ee></ee></ee></ee></ee></ee></ee></ee></ee></ee></ee></ee></ee></ee></ee></ee></ee></ee></ee></ee></ee></ee></ee></ee></ee></ee></ee></ee></ee></ee></ee></ee></de></de></de></de></ce>	dee below. Is default: per strow. It is the window and operation of the window and operation window): just the window are the window and operation of the window and operation operation of the window and operation operation operation operation of the window and operation operation	de. nstalls and tone Email tith sever selected same (but	e-window = d activates it acs frame is al frames buffer moves
window - C-u: swap - C-u C-u: delete • Move to other window • Move to specified window Ace target • Operate on specified window See also: Customize Demo: C'est la Z, video 5	<pre></pre>	v: taller v: shorter H: wider h: narrow nt w bound to	**C-x o key provides a (other-window COUN' ALL-FRAMES) **With only 2 window With 3 windows or n Type this number Type one of the form of	partially over C- <re>c-<re>c-<re>c-<re>c-<re>c-<re>c-<re>c-<re>c-<re>c-<re>c-<re>c-<re>c-<re>c-</re>c-</re>c-</re>crights: T & optional T &</re></re></re></re></re></re></re></re></re></re>	Inft right Select (r Sele	n: next p: last p: last x: swap w New: flip w Men: fli	they diffusion they divided by the condition of the condi	o: this k: skill l: all o g! quit b: next b: next l: er in their indow. y/clic order dor all fran y/our for the e function e on) wind y externa w user op ' upper le er to move cs and ten dow. y occupy acluding ju indow and iow). The e window unmber is ee original of	window buffer others window buffer buffer buffer wing of wirnes. In a little window buffer buffer buffer buffer buffer buffer buffer buffer wing of wirnes. In a little window select a package ption is selected to the table window selected buffer buffer buffer buffer buffer buffer window selected buffer window selected buffer buff	c-s- <c c-s-<do="" c-s-<le="">c-s-<ri>c-s-<ri>c-s-<do c-s-<le="">c-s-<ri>c-s-<ri>c-s-<td>s default: per trow. ce target coopen loads, in the strow and operation with the window and operation with the strong does the the mode-linge it.</td><td>el-use-acide. at one Emittith sever selected same (but one.</td><td>e-window = d activates it acs frame is al frames buffer moves t does not</td></ri></ri></do></ri></ri></c>	s default: per trow. ce target coopen loads, in the strow and operation with the window and operation with the strong does the the mode-linge it.	el-use-acide. at one Emittith sever selected same (but one.	e-window = d activates it acs frame is al frames buffer moves t does not
window - C-u: swap - C-u C-u: delete • Move to other window • Move to specified window Ace target • Operate on specified window See also: Customize Demo: C'est la Z, video 5	<pre></pre>	v: taller v: shorter H: wider h: narrow nt w bound to	** Vertically : horizontally 2: horizontally 2: horizontally 2: horizontally 2: horizontally 2: horizontally 2: horizontally 3: horizontally 3	partially over C- <re>c-<re>c-<re>c-<re>c-<re>c-<re>c-<re>c-<re>c-<re>c-<re>c-<re>c-<re>c-<re>c-</re>c-</re>c-</re>crights: T & optional T &</re></re></re></re></re></re></re></re></re></re>	lapping f Select (r Select With p This is nil. C Move to p Requ when th e other w an Ace the window her winded her vertice ally dow d binding in other nacs fram form more mpt, use windows swap be to buffer m windows swap be to buffer m windows see the wi le user-op d type us	n: next p: last p: last x: swap M-v: flip M-h:	they diffusion they divided windows with without with without with without with with without with without with with with with with with with wit	o: this k: skill l: all o g: quit b: next b: next l: all o g: quit b: next l: all o g: quit b: next lindow. y/clic order de rall fram y/clic order d	window buffer others window buffer others window buffer bu	c-s- <cccccccccccccccccccccccccccccccccc< td=""><td>see below. see below. see below. see target coopen loads, in the see the window and operation with the see that the ey does the the mode-linge it.</td><td>de. nstalls and tone Emiliate on it: tone Emiliate on it: e only with</td><td>e-window = d activates it acs frame is buffer moves t does not</td></cccccccccccccccccccccccccccccccccc<>	see below. see below. see below. see target coopen loads, in the see the window and operation with the see that the ey does the the mode-linge it.	de. nstalls and tone Emiliate on it: tone Emiliate on it: e only with	e-window = d activates it acs frame is buffer moves t does not

<u>Operation</u>	<u>Keystroke</u>	Function	<u>Note</u>
Move point to previous window can specify all frames	<f11> w 0</f11>	(pel-other-window-backward &optional N)	Select Nth previous window. n defaults to 1: meaning direct previous window. with negative n: move as (abs n) but consider all frames. If n is positive consider only current frame. This is the inverse of what does the standard (other-window). This command might be useful when ace-window is not used.
Esc-cursor keys for windmove	some other bindings, for example of the period of the peri	ample in Org-mode these keys can b ng user options to control the key bir ursor controls the <esc> bindings, it tros map C-M- bindings such as C-</esc>	is on by default on macOS and Windows, but off on Linux. M- <right> and C-M-<left> If this is not the case for your Linux system, you can activate the Esc C- bindings in replacement for the C-M- bindings you need to access several Emacs</left></right>
Move to window above	• <f11> <up> • <f1> <up> • <f1> <up> • <f2> <up> • <esc> <up> • %-<up> • <eup> • <f7><up> • <f7><up> • <up> • <up< td=""><td>(windmove-up &optional ARG)</td><td>Select the window above the current one. • With no prefix argument, or with prefix argument equal to zero, "up" is relative to the position of point in the window; otherwise it is relative to the left edge (for positive ARG) or the right edge (for negative ARG) of the current window. • If no window is at the desired location, an error is signaled. With PEL, the yu key-chord is also available when key-chord is available and active. See Key-Chords.</td></up<></up></up></up></up></up></up></up></up></up></up></up></up></up></up></up></up></up></up></up></up></up></up></up></up></up></up></up></up></up></up></up></up></up></up></up></up></up></up></up></up></up></up></up></up></up></up></up></up></up></up></up></up></up></up></up></up></up></up></up></up></up></up></up></up></up></up></up></up></up></up></up></up></up></up></up></up></up></up></up></up></up></up></up></up></up></up></up></up></up></up></up></up></up></up></up></up></up></up></up></up></up></up></up></up></up></up></up></up></up></up></up></up></up></up></up></up></up></up></up></up></up></up></up></up></up></up></up></up></up></up></up></up></up></up></up></up></up></up></up></up></up></up></up></up></up></up></up></up></up></up></up></up></up></up></up></up></up></up></up></up></up></up></up></up></up></up></up></up></up></up></up></up></up></up></up></up></up></up></up></up></up></up></up></up></up></up></up></up></up></up></up></up></up></up></up></up></up></up></up></up></up></up></up></up></up></up></up></up></up></up></up></up></up></up></up></up></up></up></up></up></up></up></up></up></up></up></up></up></up></up></up></up></up></up></up></up></up></up></up></up></up></up></up></up></up></up></up></up></up></up></up></up></up></up></up></up></up></up></up></up></up></up></up></up></up></up></up></up></up></up></up></up></up></up></up></up></up></up></f7></up></f7></eup></up></up></esc></up></f2></up></f1></up></f1></up></f11>	(windmove-up &optional ARG)	Select the window above the current one. • With no prefix argument, or with prefix argument equal to zero, "up" is relative to the position of point in the window; otherwise it is relative to the left edge (for positive ARG) or the right edge (for negative ARG) of the current window. • If no window is at the desired location, an error is signaled. With PEL, the yu key-chord is also available when key-chord is available and active. See Key-Chords.
Move to window below	• <f11> <down> • <f1> <down> • <fs> <down> • <esc> <down> • %-<down> • \$\down> • \$\down> • \$f7> <down> • bn</down></down></down></esc></down></fs></down></f1></down></f11>	(windmove-down &optional ARG)	 Select the window below the current one. With no prefix argument, or with prefix argument equal to zero, "down" is relative to the position of point in the window; otherwise it is relative to the left edge (for positive ARG) or the right edge (for negative ARG) of the current window. If no window is at the desired location, an error is signaled. With PEL, the bn key-chord is also available when key-chord is available and active. See Key-Chords.
Move to window at left	• <f11> <left> • <f1> <down> • <esc> <left> • %-<left> • \$\mathcal{C} - \left> • \$\mathcal{C} - \left> • \forall - \left></left></left></esc></down></f1></left></f11>	(windmove-left &optional ARG)	Select the window to the left of the current one. With no prefix argument, or with prefix argument equal to zero, "left" is relative to the position of point in the window; otherwise it is relative to the top edge (for positive ARG) or the bottom edge (for negative ARG) of the current window. If no window is at the desired location, an error is signaled. With PEL, the gf key-chord is also available when key-chord is available and active. See Key-Chords.
Move to window at right	<pre> <f11> <right> <f1> <right> <f1> <right> <esc> <right> %-<right> %-<right> <f7> <right> </right></f7></right></right></right></esc></right></f1></right></f1></right></f11></pre>	(windmove-right &optional ARG)	Select the window to the right of the current one. • With no prefix argument, or with prefix argument equal to zero, "right" is relative to the position of point in the window; otherwise it is relative to the top edge (for positive ARG) or the bottom edge (for negative ARG) of the current window. • If no window is at the desired location, an error is signaled. With PEL, the jk key-chord is also available when key-chord is available and active. See Key-Chords.
Exchange windows	• <f11> w x • <f7> x</f7></f11>	(ace-swap-windows)	Swap buffers of the current window with another. If 3 windows or more, a single digit shows up in the top-left corner identifying the number to type to swap to this window. Requires the <u>ace-window</u> external package. PEL downloads, install and activates it when the <u>pel-use-ace-window</u> user options is set to t.
Close/Create Windows		e used to create and remove windows to two sets of four PEL commands be	
Close this windows	• C-x 0 • <f7> 0 • <f7> d</f7></f7>	(delete-window &optional WINDOW)	This just closes the window and moves the cursor to the next window.
Kill current buffer and close window See also: Buffers	• C-x 4 0 • <f7> k</f7>	(kill-buffer-and-window)	Kill the current buffer and delete the selected window.
Close a window identified by number	<f11> w k</f11>	(ace-delete-window)	Delete a window selected by a number, a number shown in the top-left corner of the window. PEL downloads, installs and activates it when the pel-use-ace-window user options is set to t.
Close all other windows	• C-x 1 • <f7> 1 • <f7> .</f7></f7>	(delete-other-windows &optional WINDOW)	Make current window fill its frame.
Maximize one window, identified by number	<f11> w m</f11>	(ace-maximize-window) (ace-delete-other-windows)	Maximize a window. Close all windows except the window selected by number, a number shown in the top-left corner of the window. Requires the <u>ace-window</u> external package. The old versions used ace-window-maximize, but newer versions use ace-delete-maximize-windows. PEL uses the one that is available. PEL downloads, install and activates it when the <u>pel-use-ace-window</u> user options is set to t.
Create new window below	• C-x 2 • <f7> 2 • <f7> -</f7></f7>	(split-window-below &optional SIZE)	Split the selected window into two windows, one above the other. • The selected window is above. The newly split-off window is below and displays the same buffer. ► Note that Emacs default behaviour attempts to maximize the view into the current buffer when splitting the buffer into 2 windows. This means that the cursor will not be located in the same position in the new window. To change this behaviour and keep the same point in both windows, execute (setq split-window-keep-point nil). The PEL packages does that.
Create new window at right	• C-x 3 • <f7> 3 • <f7> </f7></f7>	(split-window-right &optional SIZE)	Split the selected window into two side-by-side windows. • The selected window is on the left. The newly split-off window is on the right and displays the same buffer.
Create window at cursor direction	• ESC C- <right> • ESC C-<left> • ESC C-<left> • ESC C-<down> • ESC C-<up> • <f1> C-<right> • <f1> C-<left> • <f1> C-<left> • <f1> C-<down> • <f1> C-<up> • <f11> C-<up> • <f11> C-<up> • <f11> C-<left> • <f11> C-<up> • <f11> C-<up> • <f71 c-<up=""> • <f7> C-<up> • <f7> C-<up> • <f7> C-<left> • <f7> C-<left> • <f7> C-<up> • <f7> C-<up></up></f7></up></f7></left></f7></left></f7></up></f7></up></f7></f71></up></f11></up></f11></left></f11></left></f11></left></f11></left></f11></left></f11></left></f11></up></f11></up></f11></up></f1></down></f1></left></f1></left></f1></right></f1></up></down></left></left></right>	(pel-create-window-right) (pel-create-window-left) (pel-create-window-down) (pel-create-window-up)	Create a window at the location pointed by the cursor's direction, and move point inside the new window. • The 4 different commands and shown in the same cell for convenience, one for each of the available cursors: <ri>right>, <left>, <down> and <up>. • There are 4 possible sets of bindings: • 3 sets of stand-alone commands: • Commands with <f11> prefix, always available. • Commands with ESC prefix, available when pel-windmove-on-esc-cursor user option is on (set to t). • Commands with <f1> prefix, available when pel-windmove-on-f1-cursor user option is on (set to t). • The Hydra-based commands, with the Hydra activated with any of the key sequences that use the <f7> prefix. Available when pel-use-hydra user option is set to t.</f7></f1></f11></up></down></left></ri>

<u>Operation</u>	<u>Keystroke</u>	Function	<u>Note</u>
Close a window at cursor direction	The best way to use the	em is to type them once and then use	
	The PEL package als PEL also provides the Ventered, commands car Each of the first 5 commands The Emacs default key The commands with the The commands with ES The commands with <f< td=""><td>n be issued again without any prefix. s below have 5 possible bindings: binding using the C-x prefix. e default PEL <11> prefix, always av sc prefix, available when pel-wind 1> prefix, available when pel-wind nands, with the Hydra activated with</td><td>ch can be started with one of the following commands using the <f7> prefix. Once the Hydra is</f7></td></f<>	n be issued again without any prefix. s below have 5 possible bindings: binding using the C-x prefix. e default PEL <11> prefix, always av sc prefix, available when pel-wind 1> prefix, available when pel-wind nands, with the Hydra activated with	ch can be started with one of the following commands using the <f7> prefix. Once the Hydra is</f7>
Grow window taller	• C-x ^ • <f11> w s V • ESC M-<up> • <f1> M-<up> • <f7> V</f7></up></f1></up></f11>	(enlarge-window DELTA &optional HORIZONTAL)	Grow window taller by DELTA lines (defaults to 1), specify more with C-u n (or M- n) argument prefix. • See note above for availability of various bindings.
Shrink window smaller	• <f11> w s v • ESC M-<down> • <f1> M-<down> • <f7> v</f7></down></f1></down></f11>	(shrink-window DELTA &optional HORIZONTAL)	Shrink height of window by DELTA lines (defaults to 1), specify more with C-u n (or M- n) argument prefix. • See note above for availability of various bindings.
Grow windows wider	• C-x } • <f11> w s H • ESC M-<right> • <f1> M-<right> • <f7> H</f7></right></f1></right></f11>	(enlarge-window-horizontally DELTA)	Enlarge the current window horizontally. See note above for availability of various bindings.
Shrink window narrower	• C-x { • <f11> w s h • ESC M-<left> • <f1> M-<left> • <f7> h</f7></left></f1></left></f11>	(shrink-window-horizontally DELTA)	Reduce the width of the current window. • See note above for availability of various bindings.
Make all windows the same size	• C-x + • <f11> w s = • ESC <kp-5> • <f1> <kp-5> • <f7> =</f7></kp-5></f1></kp-5></f11>	(balance-windows & optional WINDOW-OR-FRAME)	Balance the sizes of windows of WINDOW-OR-FRAME. WINDOW-OR-FRAME is optional and defaults to the selected frame. If WINDOW-OR-FRAME denotes a frame, balance the sizes of all windows of that frame. If WINDOW-OR-FRAME denotes a window, recursively balance the sizes of all child windows of that window. See note above for availability of various bindings.
Reduce current window size if buffer is smaller than window	• C-x - • <f11> w s -</f11>	(shrink-window-if-larger-than- buffer &optional WINDOW)	Shrink height of current window if its buffer doesn't need so many lines. More precisely, shrink window vertically to be as small as possible, while still showing the full contents of its buffer. Do not shrink window to less than 'window-min-height' lines. Do nothing if the buffer contains more lines than the present window height, or if some of the window's contents are scrolled out of view, or if shrinking this window would also shrink another window, or if the window is the only window of its frame.
Quick Window Layout Change	The following commands flip	the layout of 2 windows: the current	and next window between 2 horizontal windows to 2 vertical windows and vice versa.
Flip 2 horizontal windows to 2 vertical ones	• <f11> w v • <f7> M-v</f7></f11>	(pel-2-vertical-windows)	Convert 2 horizontal windows into 2 vertical windows. • Flip the orientation of the current window and its next one. • The next window is placed at the right of the current window.
Flip 2 vertical windows to 2 horizontal ones	• <f11> w h • <f7> M-h</f7></f11>	(pel-2-horizontal-windows)	Convert 2 horizontal windows into 2 horizontal windows. • Flip the orientation of the current window and its next one. • The next window is placed below the current one.
Window Layout History	_	ow you to restore a previously used w package, a package that is part of th	vindow layout. le standard Emacs. PEL activates them when pel-use-winner user option is t.
Restore an earlier window configuration	• C-c <left> • <f11> w p • <f7> p</f7></f11></left>	(winner-undo)	Switch back to an earlier window configuration saved by Winner mode. In other words, "undo" changes in window configuration.
Restore a more recent window configuration	• C-c <right> • <f11> w n • <f7> n</f7></f11></right>	(winner-redo)	Restore a more recent window configuration saved by Winner mode.
Open Buffer in another window	the other does not. Under Pl		ide another window. One command select that other window (move point to that window) and IDO version of the command when the pel-use-ido customization variable is set to t, otherwise more information at the prompt.
Select buffer in other window	• C-x 4 b • <f11> w B</f11>	(ido-switch-buffer-other-window)	Select buffer bufname in another window (switch-to-buffer-other-window). See <u>Select Buffer</u> .
Display buffer in other window, don't select the other window.	• C-x 4 C-o • <f11> w b</f11>	(ido-display-buffer) ———————————————————————————————————	Display a buffer in other window but don't select it. When <i>pel-use-ido</i> is customized to t, (ido-display-buffer) is used, which prompts and provides easy to select list of available buffer names. Otherwise the standard Emacs (display-buffer) is used prompting without showing the available buffers.

<u>Operation</u>	<u>Keystroke</u>	Function	<u>Note</u>				
Dedicated Windows	Emacs windows can be dedicated to specific buffers in such a way that future windows operations do not affect the dedicated windows. The following commands help you manage dedicated windows.						
Show dedicated status of current window	<f11> w d ?</f11>	(pel-show-window-dedicated- status)	Display the dedicated status of the current window in the echo area (the minibuffer).				
Toggle dedicated status of current window	• <f11> w d d • <f7> d</f7></f11>	(pel-toggle-window-dedicated)	Toggle the dedicated status of the current window, changing a normal window into a dedicated one and a dedicated window into a normal one.				
Follow Mode		ws mode which applies all scroll con the Emacs Wiki Scroll All Mode pag	nmands to all visible windows. To support mouse wheel or scroll bar you need to implement e.				
See also: Scrolling	Text in the first window goes to the bottom and then	continues there.	 When Emacs follow-mode is used on 2 or more windows, these windows show the text of the same buffer spread across these windows that act as a one continuous stream. Follow mode is a minor mode that combines windows into one tall virtual window. This is accomplished by two main techniques: The windows always displays adjacent sections of the buffer. This means that whenever one window is moved, all the others will follow. (Hence the name Follow mode.) Should point (cursor) end up outside a window, another window displaying that point is selected, if possible. This makes it possible to walk between windows using normal cursor movement commands. Follow mode comes to its prime when used on a large screen and two or more side-by-side windows are used. The user can, with the help of Follow mode, use these full-height windows as though they were one. 				
Toggle follow-mode See also: <u>∑ Scrolling</u>	• <f11> w f • <f11> f</f11></f11>	(follow-mode &optional ARG)	Toggle Follow mode. With a prefix argument ARG, enable Follow mode if ARG is positive, and disable it otherwise.				
Scrolling Window	For all other commands to	o scroll the window text, see the SS	crolling page.				

Windows - Reference

Topic/URL	Comment
GNU Emacs — Displaying a Buffer in a Window	Describes the Emacs features related to displaying buffers inside windows.
GNU Emacs Lisp — Displaying Buffers — The Zen of Buffer Display	Describes the rules Emacs tries to use to control the creation of new windows when they are created dynamically from commands.