Scrolling

Description	Keystroke	Function	Note
Window Scrolling	Emacs has **several** scroll modes: • One line scroll operations in any mode provided by the PEL package scroll commands. • Single window scroll, where the navigation commands for up and down are translated to scroll operations when scroll is possible. • Dual window scroll, implemented by the PEL package, where a PEL scroll command in one window is reflected in the other, associated window. • The scroll-all-mode, where all windows are scrolled together when one of the native Emacs scroll commands is used. Emacs also has the **follow mode**, described below. Note that when Emacs is running in Terminal (TTY) mode, mouse scrolling is normally disabled. PEL activates the ability to scroll with the mouse in terminal mode when the xterm-mouse-mode is activated (it can be enabled by **<f11><f12>**). See the ∑ Mouse table for details.		
Standard Emacs Scroll Commands	Emacs provides the following standard commands to scroll text in the current window. These commands scroll without having to enable a special scroll mode (just as the PEL provided single line scroll commands below). However that also work in the single window scroll and the scroll all mode. With the current implementation they do **not** scroll both windows in the dual scroll mode provided by PEL.		
Scroll up by near full screen	• **C-v** • **<page-down>**	(scroll-up-command &optional ARG)	• Use an argument number (M-number) before the C-v to specify the number of lines to scroll. • In cua-mode C-v is used for paste.
Scroll down by near full screen	• **M-v** • **<page up>**	(scroll-down-command &optional ARG)	Use an argument number (M-number) before the C-v to specify the number of lines to scroll
Smooth Scrolling	Emacs default scrolling mechanism makes the screen jumps about the size of half a window when the cursor gets close to the top or the bottom of the window. This mode was originally designed to reduce processing requirements. This behaviour can however be surprising and annoying. 📦 The smooth-scrolling package provides the smooth-scrolling external package that makes Emacs provide a much smoother scrolling experience. PEL activates smooth scrolling when the **pel-use-smooth-scolling** user option is set to **t**. You can also turn it on or off with the following command.		
Toggle smooth scrolling mode	**<f11> | s**	(**smooth-scrolling-mode** &optional ARG)	Toggle smooth scrolling mode on/off. • If a numeric argument is used, a positive one turn smooth scrolling on while a negative one turns it off. 📦 Requires the smooth-scrolling package. PEL activates it when the pel-use-smooth-scrolling user option is set to **t**.

One line scroll in any mode (PEL specific)	The following commands are provided by the PEL package.
	• The first two commands, pel-scroll-up and pel-scroll-down allow scrolling the text of current window, without moving point.
	• The other commands are used to create a group of 2 or more windows that will all scroll together.
	  • These windows will scroll together only when the **pel-scroll-up**, **pel-scroll-down** commands are used, but also when the following keys are used: **pel-home** (bound to **<home>**, and **pel-end** (bound to **<end>**). Nothing else unfortunately. See ∑ **Navigation**.
	  • 🚧 This means is not a completely supported scrolling mode as of this release of PEL: no minor-mode is involved, and all other operations that makes the current window scroll will not cause scrolling in the other windows.
	  • The current behaviour is sufficient for several tasks, though, and permit the scrolling of a group of 2 or more windows, not necessarily all windows currently used.
	
	☛ ⚠ The **<M-down>** and **<M-up>** keys do not scroll in org-mode. Use **<M-f5>** and **<M-f6>** instead.		

Scroll up **Bring text ahead into view.**	• **<M-down>** • **<M-f5>**	(**pel-scroll-up** &optional N)	Move text 1 line up (same direction as forward) toward the end of buffer. The following commands affect the behaviour of this: • (pel-toggle-scroll-sync) toggle scrolling 2 windows in sync. • (pel-toggle-numlock) toggles the numlock, to get access to the keypad keys as cursors or as digits. To use this they the keypad must be off. Repetition and inverse movement supported by numeric argument. • If N is specified it identifies a repetition count. • If N is negative it means the other direction. ⚠ In Org Mode **<M-down>** is mapped to Org-mode functionality.
Scroll down **Bring text behind into view.**	• **<M-up>** • **<M-f6>**	(**pel-scroll-down** &optional N)	Move text 1 line down (same direction as backwards), toward the top of buffer. The following commands affect the behaviour of this: • (pel-toggle-scroll-sync) toggle scrolling 2 windows in sync. • (pel-toggle-numlock) toggles the numlock, to get access to the keypad keys as cursors or as digits. To use this they the keypad must be off. Repetition and inverse movement supported by numeric argument. • If N is specified it identifies a repetition count. • If N is negative it means the other direction. ⚠ In Org Mode **<M-up>** is mapped to Org-mode functionality.

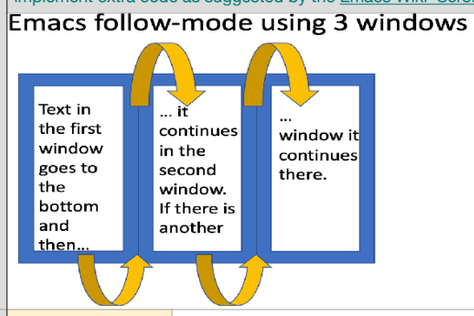
Toggle PEL window scroll mode	**<f11> | |**	(**pel-toggle-scroll-sync**)	Toggles the PEL window scroll-lock sync:
			  • if currently off, place the current and next window inside the list of windows that must be scrolled together: the PEL window scroll sync group.
			  • If currently on, turn it off.
			⚠ This command does not activate a minor-mode. There is no indication inside the mode line, but the command display a message in the echo area.

Add current window to the PEL window scroll group	**<f11> | +**	(**pel-add-window-to-scroll-sync**)	Add current window to the PEL scroll sync group. • If PEL scroll sync is currently off, it adds the current and next window to the PEL window scroll sync group.
Remove current window from the PEL window scroll group	**<f11> | -**	(**pel-remove-window-from-scroll-sync**)	Remove current window from the PEL window scroll sync group. • If less than 2 windows remain in the group, the command display a warning. • It's possible to leave only 1 window inside the PEL window scroll sync group. That allows adding another window later. If there is only one window in the group no extra scroll is ever performed. To turn it off completely re-issue to command or use the pel-toggle-scroll-sync command twice.
Scroll other window	The *other* window is the *next* window in the list of windows displayed in the Emacs frame. The following commands can be used to scroll the line of that other window.		
Scroll other window up by ~ 1 full window	• **Esc <PgDown>** • **C-M-v**	(**scroll-other-window** &optional ARG)	Scroll the text of the *other* window one near-full window down (toward the bottom of the buffer). • If a number argument is specified, that identifies the number of lines to scroll. • A negative argument inverses the direction.
Scroll other window down by ~ 1 full window	• **Esc <PgUp>** • **C-M-S-v**	(**scroll-other-window-down** &optional LINES)	Scroll the text of the *other* window one near-full window up (toward the top of the buffer). • If a number argument is specified, that identifies the number of lines to scroll. • A negative argument inverses the direction.
Scroll other window up by 1 line	**<M-S-f5>**	(**pel-scroll-up-other** &optional N)	Scroll the text of the *other* window one line down (toward the bottom of the buffer). • If a number argument is specified, that identifies the number of lines to scroll. • A negative argument inverses the direction.
Scroll other window down by 1 line	**<M-S-f6>**	(**pel-scroll-down-other** &optional N)	Scroll the text of the *other* window one line up (toward the top of the buffer). • If a number argument is specified, that identifies the number of lines to scroll. • A negative argument inverses the direction.

Description	Keystroke	Function	Note

Scrolling left/right	Scrolling horizontally can be useful for very long lines or narrow windows.
	• By default, Emacs disables these commands as they might confuse new users.
	  • The first time you try to use it Emacs will prompt, asking whether you want to enable it.
	  • If you decide to enable it forever this information will be written inside your customization data.		

scroll window (lines) left	• `C-x <` • `C-<PgDown>`	(**scroll-left** &optional ARG SET-MINIMUM)	Scroll window left. • This command is disabled by default.
scroll window (lines) right	• `C-x >` • `C-<PgUp>`	(**scroll-right** &optional ARG SET-MINIMUM)	Scroll window right. • This command is disabled by default.
1 Window Scroll	The single window scroll mode allows using normal cursor movement keys to scroll the window.		
Toggle Scroll-Lock Mode	`<f11> | l`	(**scroll-lock-mode** &optional ARG)	Toggle single window scroll mode. • This is a buffer-local minor mode for pager-like scrolling. • With a prefix argument ARG, enable the mode if ARG is positive, and disable it otherwise. • When enabled, keys that normally move point by line or paragraph will scroll the buffer by the respective amount of lines instead and point will be kept vertically fixed relative to window boundaries during scrolling.
All Windows Scroll	In this mode **all** visible windows are scrolled together.		
Toggle Scroll All Mode	`<f11> | a`	(**scroll-all-mode** &optional ARG)	Toggle shared scrolling in same-frame windows (Scroll-All mode). • With a prefix argument ARG, enable Scroll-All mode if ARG is positive, and disable it otherwise. • When Scroll-All mode is enabled, scrolling commands invoked in one window apply to all visible windows in the same frame.

Follow Mode
 	
See also: **Windows**
	Emacs has a scroll all windows mode which applies all scroll commands to all visible windows. To support mouse wheel or scroll bar you need to implement extra code as suggested by the Emacs Wiki Scroll All Mode page.
	 
			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: **Windows**	• `<f11> | f` • `<f11> w f`	(**follow-mode** &optional ARG)	Toggle Follow mode. With a prefix argument ARG, enable Follow mode if ARG is positive, and disable it otherwise.

Scrolling — Reference

Topic	URL	Note
Scroll two opened buffers in a split window at the same time	https://stackoverflow.com/questions/18092100/scroll-two-opened-buffers-in-a-split-window-at-the-same-time	Use **scroll-all-mode** I got it to work briefly (when (SL) was displayed in the mode line lighter) but have not been able to get it to work ever since. The scroll-lock-mode behaves like it is documenting in the source code; which is just to prioritize scrolling instead of moving point in the lines.
Emacs Wiki - Scroll All Mode	https://www.emacswiki.org/emacs/ScrollAllMode	
Emacs Wiki - Scroll Lock	https://www.emacswiki.org/emacs/ScrollLock	
GNU Emacs Manual - Scrolling	https://www.gnu.org/software/emacs/manual/html_node/emacs/Scrolling.html	
GNU Emacs Manual - Auto Scroll		
GNU Emacs - Follow Mode	https://www.gnu.org/software/emacs/manual/html_node/emacs/Follow-Mode.html	
Can I scroll the windows in emacs synchronously?	https://stackoverflow.com/questions/33210981/can-i-scroll-the-windows-in-emacs-synchronously	