

ibuffer-mode

Operation	Keystroke	Function	Note
ibuffer Mode ○ Mark buffers - Mark for deletion - Operate on marked - kill marked - filters ○ groups - Navigate in groups - Sort list, Other, Help Last updated on: 2025-11-19	Emacs ibuffer-mode list buffers in controllable, flexible way and provides selection, filtering, grouping commands. 🙌 It is extremely useful! - PEL provides the following extensions when the corresponding pel-use- customizable user-option is turned on:  ibuffer-vc  pel-use-ibuffer-vc Allows creating groups based on version control.  ibuffer-tramp  pel-use-ibuffer-tramp Allows creating groups based on ⓘ Tramp node With PEL you can also identify minor modes automatically activated in ibuffer by identifying them in the pel-ibuffer-activates-minor-modes user-option. See also: Intro to ibuffer A nice video presentation of ibuffer-mode. ⓘ Buffers		
Open this PDF file. See also: ⓘ Help/Info	<f12> <f1>	(pel-help-pdf ⓘoptional OPEN-WEB-PAGE)	Open this ⓘ ibuffer-mode local PDF. If the prefix argument (like C-u or M--) is used, then it opens the remote GitHub hosted raw PDF instead. If the pel-flip-help-pdf-arg user-option is set it's the other way around.
ⓘ Customize PEL ibuffer Support	<f12> <f2>	(pel-customize-pel ⓘoptional OTHER-WINDOW)	Customize PEL ibuffer support: open PEL buffer support specific group. If OTHER-WINDOW is non-nil (use C-u), display in other window.
ⓘ Customize Emacs Ibuffer support	<f12> <f3>	(pel-customize-library ⓘoptional OTHER-WINDOW)	Customize Emacs ibuffer , ibuffer-vc . When a prefix argument (like C-u) opens the buffer inside another window.
ibuffer Mode	Commands available in ibuffer window. With PEL, the C-x C-b key binding open the ibuffer window. PEL also provide extra filters and sorting commands.		
Toggle ibuffer auto-update mode	<f12> a C-c C-a	(ibuffer-auto-mode ⓘoptional ARG)	Toggle use of Ibuffer's auto-update facility (Ibuffer Auto mode). When this mode is active, ibuffer automatically updates the buffer state when they change, otherwise (by default) the state needs to be actively refreshed by pressing g in the ibuffer window.
Mark buffers	Inside the buffer-mode window, use the following commands to mark and un-mark buffer entries. m : Mark the buffer at point. t : Unmark all currently marked buffers, and mark all unmarked buffers. * c : Change the mark used on marked buffers. u : Unmark the buffer at point. DEL : Unmark the previous buffer. * M : Mark buffers by major mode. * u : Mark all "unsaved" buffers. This means that the buffer is modified, and has an associated file. * m : Mark all modified buffers, regardless of whether they have an associated file. * s : Mark all buffers whose name begins and ends with ¹*. * e : Mark all buffers which have an associated file, but that file doesn't currently exist. * r : Mark all read-only buffers. * / : Mark buffers in 'dired-mode'. * h : Mark buffers in 'help-mode', 'apropos-mode', etc. . : Mark buffers older than 'ibuffer-old-time'. % n : Mark buffers by their name, using a regexp. % m : Mark buffers by their major mode, using a regexp. % f : Mark buffers by their filename, using a regexp. % g : Mark buffers by their content, using a regexp. % L : Mark all locked buffers. U : Unmark all marked buffers. M-DEL : Unmark buffers marked with MARK.		
Mark for deletion	d : Mark the buffer at point for deletion.		
Operate on marked buffers	Once buffers are marked, S : Save the marked buffers. A : View the marked buffers in the selected frame. H : View the marked buffers in another frame. V : Revert the marked buffers. T : Toggle read-only state of marked buffers. L : Toggle lock state of marked buffers. M-s a C-s : Do incremental search in the marked buffers. M-s a C-M-s : Isearch for regexp in the marked buffers. r : Replace by regexp in each of the marked buffers. Q : Query replace in each of the marked buffers. I : As above, with a regular expression. P : Print the marked buffers. O : List lines in all marked buffers which match a given regexp (like the function 'occur'). X : Pipe the contents of the marked buffers to a shell command. N : Replace the contents of the marked buffers with the output of a shell command. ! : Run a shell command with the buffer's file as an argument. E : Evaluate a form in each of the marked buffers. This is a very flexible command. For example, if you want to make all of the marked buffers read-only, try using (read-only-mode 1) as the input form. W : As above, but view each buffer while the form is evaluated.		
Kill/close marked buffer	D : Kill the marked buffers. No impact on file (any associated file is not deleted). x : Kill all buffers marked for deletion. No impact on file (any associated file is not deleted).		
Hide marked buffers	k : Hide the marked lines from the "Ibuffer" buffer, but don't kill the associated buffer. Use g to refresh and show them again.		
Refresh	g : Regenerate the list of all buffers. Prefix arg means to toggle whether buffers that match 'ibuffer-maybe-show-predicates' should be displayed.		
Manage filters	Use the following commands to filter the buffer entries according to one or several criteria. Use the commands below to filter the list.		
Filterering commands	/ RET : Add a filter by any major mode. / m : Add a filter by a major mode now in use. / n : Add a filter by buffer name. / b : Add a filter by basename. / d : Add a filter by directory name. / i : Add a filter by modified buffers. / > : Add a filter by buffer size. / * : Add a filter by special buffers. / M : Add a filter by derived mode. / c : Add a filter by buffer content. / f : Add a filter by filename. / . : Add a filter by file extension. / e : Add a filter by an arbitrary Lisp predicate. / < : Add a filter by buffer size. / v : Add a filter by buffers visiting files.		
Manage filters	Once the filters are created you can also combine them using the following commands. / s : Save the current filters with a name. / r : Switch to previously saved filters. / a : Add saved filters to current filters. / & : Replace the top two filters with their logical AND. / : Replace the top two filters with their logical OR. / p : Remove the top filter. / ! : Invert the logical sense of the top filter. This can be applied to any filter. / M-d : Break down the topmost filter. / / : Remove all filtering currently in effect.		
Select filter by completion of filter description	<f12> c	(ibuffer-filter-chosen-by-completion)	Select and apply filter chosen by completion against available filters. - Indicates corresponding key sequences in echo area after filtering. - The completion matches against filter description text of filters in 'ibuffer-filtering-alist'.
Filter by directory name expression	<f12> d	(ibuffer-filter-by-directory QUALIFIER)	Limit current view to buffers with directory matching QUALIFIER. For a buffer associated with file '/a/b/c.d', this matches against '/a/b'. For a buffer not associated with a file, this matches against the value of 'default-directory' in that buffer.

Operation	Keystroke	Function	Note
• Filter Groups	The ibuffer mode supports creating groups of entries. The groups can be based on the filters created by the commands above and given names. The / D command creates a set of groups based on the directories of the files. The other commands provide ability to create specialized groups, saving them and deleted them.		
IBuffer Mode Filter Group commands	/ D : Group buffers by directory path names / g : Create filter group from filters. / P : Remove top filter group. / \ : Remove all active filter groups / S : Save the current groups with a name. / R : Restore previously saved groups. / X : Delete previously saved groups.		
IBuffer Mode commands to move across Filter Groups	Use the following commands to navigate through the filter groups		
	TAB : Move to the next filter group. M-n : Move to the next filter group. M-p : Move to the previous filter group.		
• Sort entries	Use the following commands to sort the list of buffer entries		
	, : Rotate between the various sorting modes. s i : Reverse the current sorting order. s a : Sort the buffers lexicographically. s f : Sort the buffers by the file name. s v : Sort the buffers by last viewing time. s s : Sort the buffers by size. s m : Sort the buffers by major mode. s p : Sort the buffers by path name.		
• Extra Filters	The following extra group filtering commands are provides by external packages that PEL can active:  ibuffer-vc allows creating groups based on version control  PEL activates it when pel-use-ibuffer-vc is set to t.  ibuffer-tramp allows creating groups based on Tramp node  PEL activates it when pel-use-ibuffer-tramp is set to t.		
Group by VC backend	<f12> v	(ibuffer-vc-set-filter-groups-by-vc-root)	Group all entries by VC back-end. • Requires ibuffer-vc activated by pel-use-ibuffer-vc
Group by Tramp connection	<f12> t	(ibuffer-tramp-set-filter-groups-by-tramp-connection)	Group all entries by Tramp connection. • Requires ibuffer-tramp activated by pel-use-ibuffer-tramp
• Other commands			
• change list format	` : Change the <i>current display format</i> . 🙌 Use this to see the complete file name when the file name is long.		
• Bury buffer	b : Bury (not kill!) the buffer on this line.		
• View Buffer	RET : View the buffer on this line. o : As above, but in another window. C-o : As both above, but don't select the new window.		
• Navigate	SPC : Move point to the next line. C-n : Move point to the next line. C-p : Move point to the previous line.		
• Diff buffer & file	= : View the differences between this buffer and its associated file.		
• Help	h : Show emacs ibuffer help ? : Show emacs ibuffer help C-h m : Show emacs ibuffer help <f1> m : Show emacs ibuffer help <f12> <f1> : Open this PEL PDF help		