




Function Keys Mappings

| Function | | shift (⇧) | control (^) | ^⇧ | meta (⌘) | ⌘⇧ | ^⌘ | ^⌘⇧ |
|--------------|--|--|-----------------------------|-----------------------------|--|---|-----------------------|-----------------------|
| Modifier Key | | S | C | C-S | M | M-S | C-M | C-M-S |
| F1 | Prefix: (help) | Prefix: (help) | unassigned | unassigned | unassigned > F6 | unassigned > C-M-p | unassigned > C-M-p | unassigned > C-M-p |
| F2 | Prefix for commands: 2C-..., bm-next , outline-... | unassigned | unassigned | unassigned | unassigned > F7 | unassigned > C-M-p | unassigned > C-M-p | unassigned > C-M-p |
| F3 | (kmacro-start-macro-or-insert-counter ARG) | (kmacro-start-macro-or-insert-counter ARG) | unassigned | unassigned | unassigned > F8 | unassigned > C-M-p | unassigned > C-M-p | unassigned > C-M-p |
| F4 | (kmacro-end-or-call-macro ARG &optional NO-REPEAT) | (kmacro-end-or-call-macro ARG &optional NO-REPEAT) | unassigned | unassigned | unassigned > F9 | unassigned > C-M-p | unassigned > C-M-p | unassigned > C-M-p |
| F5 | (repeat REPEAT-ARG) | (repeat REPEAT-ARG) and mark | (centi-assign) | unassigned | (pel-scroll-up &optional N) | (pel-scroll-up-other &optional N) | unassigned > ~ | unassigned > ~ |
| F6 | Prefix: pel:f6 | Prefix: pel:f6 | unassigned | unassigned | (pel-scroll-down &optional N) | (pel-scroll-down-other &optional N) | unassigned > ~ | unassigned > ~ |
| F7 | <ul style="list-style-type: none"> Prefix: pel-Σwnd 🍏 pel-Σnarrate  pel-use-hydra= t ,  pel-use-applescript=t | unassigned as <f19> | unassigned | unassigned | unassigned | unassigned | unassigned > ~ | unassigned > ~ |
| F8 | Prefix: projectile (activated by F11 F8  pel-use-projectile=t) | unassigned | unassigned | unassigned | unassigned | unassigned | unassigned > ~ | unassigned > ~ |
| F9 | unassigned by default. If pel-activate-f9-for-greek set, then used as prefix for Greek letters. See Σ Inserting Text | unassigned | unassigned | unassigned | unassigned | (pel-show-init-time) | unassigned > ~ | unassigned > ~ |
| F10 | (menu-bar-open &optional FRAME) | (menu-bar-open &optional FRAME) | (buffer-menu-open) | (buffer-menu-open) | (toggle-frame-maximized) | (toggle-frame-maximized) | unassigned > ~ | unassigned > ~ |
| F11 | Prefix: pel: | Prefix: pel: | unassigned | unassigned | Prefix: pel2: | (pel-scroll-up &optional N) | unassigned > ~ | unassigned > ~ |
| F12 | PEL Mode sensitive prefix | Prefix: mode sensitive | unassigned | unassigned | PEL Mode sensitive prefix Same as F12: helps typing sequences with Meta keys. The <M-f12> bindings are often major mode specialized commands for the more generic commands bound to corresponding <M-f11> key sequence. | (pel-scroll-down &optional N) | unassigned > ~ | unassigned > ~ |

Special Legend for the above table

| The colour and symbols used in the above table follow the standard legend described in >Legend with the additional conventions listed here. | |
|--|---|
| unassigned | The key is not assigned by PEL. |
| unassigned | The key with this modifier is not accessible, and does not do anything, in terminal mode. |
| unassigned > | When pressing this key with this/these modifier(s) in terminal mode the action corresponds to the key identified after the > symbol. |
| (toggle-frame-maximized) | In terminal mode the command toggle-frame-maximized is executed, but has no effect. |
| text in green | Text in green describes a PEL specific feature. Symbols are in bold. |
| PEL Mode sensitive prefix | <p>PEL provides mode sensitive keys that are globally accessible via the <f11> SPACE key sequence prefix.</p> <p>For example, when editing a buffer using the emacs-lisp-mode major mode, the <f12> prefix gives access to commands that are specific to Emacs Lisp. If you want to access the same commands while editing a buffer that use another major-mode, you can then type <f11> SPACE 1 prefix to get to the same list of commands.</p> <p>PEL provides a set of mode sensitive key binding sets for several major modes, but not for all of them. As time evolves the number of major modes supported will grow.</p> |