

```

;; ESC can be pressed and released instead of holding Alt key (Meta key)
(describe-key (kbd "C-[") ; C-[ sends the same signal as ESC

;; Place cursor after closing parenthesis, then C-x C-e to evaluate expression:
(describe-function #'eval-last-sexp) ; C-x C-e
(describe-function #'eval-print-last-sexp) ; C-j (in *scratch*, special buffers)
(describe-function #'keyboard-quit) ; C-g
(describe-function #'keyboard-escape-quit) ; ESC ESC ESC
(describe-function #'execute-extended-command) ; M-x
(describe-function #'execute-extended-command-for-buffer) ; M-X
(describe-function #'eval-expression) ; M-:, M-ESC :
(describe-function #'list-command-history) ; (default unbound)
(describe-function #'repeat-complex-command) ; C-x ESC ESC, C-x M-:
;; Minibuffer history commands: M-n and M-p
(describe-variable 'command-history)
(describe-variable 'history-length)
(describe-function #'previous-error) ; M-g p, M-g M-p
(describe-function #'next-error) ; C-x `, M-g n, M-g M-n
;; Emacs pages are between FORM FEED characters ^L (C-q C-l)
(describe-function #'quoted-insert) ; C-q
(describe-variable 'page-delimiter)
(describe-function #'insert-char) ; C-x 8 RET (M-j, or disable `fido-mode')
(describe-function #'make-string) ; (default unbound)
(describe-function #'self-insert-command) ; (bound to ordinary text characters)
(describe-function #'insert) ; (default unbound)

;; Help
(describe-variable 'text-quoting-style)
(describe-function #'view-emacs-FAQ) ; C-h C-f
(describe-function #'info-emacs-manual) ; C-h r
(describe-function #'info) ; C-h i
(describe-function #'man) ; (default unbound)
(describe-function #'shortdoc) ; (default unbound)
(describe-function #'shortdoc-display-group) ; (default unbound)
(describe-function #'help) ; (default unbound)
(describe-function #'help-for-help) ; C-h C-h, C-h ?
(describe-function #'isearch-help-map) ; C-h C-h (inside I-search)

(describe-function #'describe-bindings) ; C-h b
(describe-function #'describe-keymap) ; (default unbound)
(describe-function #'describe-mode) ; C-h m
(describe-function #'which-key-mode) ; (default unbound)
(describe-function #'describe-prefix-bindings) ; C-h, ?, F1 (after prefix key)
(describe-function #'describe-key-briefly) ; C-h c
(describe-function #'describe-key) ; C-h k

(describe-function #'describe-variable) ; C-h v
(describe-function #'describe-function) ; C-h f
(describe-function #'describe-command) ; C-h x
(describe-function #'where-is) ; C-h w
(describe-function #'describe-symbol) ; C-h o
(describe-function #'info-lookup-symbol) ; C-h S

(describe-function #'xref-find-definitions) ; M-. (make TAGS with etags, ctags)
(describe-function #'xref-find-definitions-other-window) ; C-x 4 .
(describe-function #'xref-find-references) ; M-?
(describe-function #'xref-find-apropos) ; C-M-. (can crash Emacs)
(describe-function #'apropos) ; (default unbound)
(describe-function #'apropos-variable) ; (default unbound)
(describe-function #'apropos-user-option) ; (default unbound)
(describe-function #'apropos-command) ; C-h a
(describe-function #'apropos-documentation) ; C-h d

```

```
(describe-function #'help-goto-previous-page) ; p (in help buffer)
(describe-function #'help-goto-next-page) ; n (in help buffer)
(describe-function #'backward-button) ; S-TAB (in help buffer)
(describe-function #'forward-button) ; TAB (in help buffer)
(describe-function #'help-follow-symbol) ; C-c C-c (in help buffer)
(describe-function #'help-find-source) ; C-h 4 s (in help buffer)
(describe-variable 'help-at-pt-display-when-idle)
(describe-function #'display-local-help) ; C-h .
```

```
(describe-variable 'package-archives)
(describe-function #'view-external-packages) ; C-h C-e
(describe-function #'list-packages) ; (default unbound)
(describe-function #'finder-by-keyword) ; C-h p
(describe-function #'describe-package) ; C-h P
```

```
(describe-function #'describe-categories) ; (default unbound)
(describe-function #'what-cursor-position) ; C-x =
(describe-function #'what-line) ; (default unbound)
(describe-function #'count-words) ; (default unbound)
(describe-function #'count-words-region) ; M-=
(describe-function #'count-lines-page) ; C-x l
```

;; Narrowing, widening

```
(describe-function #'narrow-to-region) ; C-x n n
(describe-function #'narrow-to-page) ; C-x n p
(describe-function #'narrow-to-defun) ; C-x n d
(describe-function #'goto-line-relative) ; C-x n g
(describe-function #'widen) ; C-x n w
```

```
(describe-function #'save-restriction) ; (default unbound)
```

;; Movement

```
(describe-variable 'next-screen-context-lines) ; affects M-v, C-v, etc.
(describe-function #'beginning-of-buffer) ; M-<
(describe-function #'end-of-buffer) ; M->
(describe-function #'backward-page) ; C-x [
(describe-function #'forward-page) ; C-x ]
(customize-variable 'scroll-error-top-bottom)
(describe-function #'scroll-down-command) ; M-v
(describe-function #'scroll-up-command) ; C-v
(describe-function #'scroll-other-window-down) ; C-M-S-v
(describe-function #'scroll-other-window) ; C-M-v
```

```
(describe-function #'scroll-left) ; C-x <
(describe-function #'scroll-right) ; C-x >
```

```
(describe-function #'mwheel-scroll) ; MouseScroll, S-MouseScroll (horizontal)
```

```
(describe-function #'reposition-window) ; C-M-l
(describe-variable 'scroll-margin)
(describe-variable 'recenter-redisplay)
(customize-variable 'recenter-positions) ; set to (top 1 2 middle bottom)
(describe-function #'recenter-top-bottom) ; C-l
(describe-function #'recenter-other-window) ; C-M-S-l
(describe-function #'move-to-window-line-top-bottom) ; M-r
(describe-variable 'line-move-visual)
(describe-variable 'track-eol)
(describe-variable 'next-line-add-newlines)
(describe-function #'previous-line) ; C-p
(describe-function #'next-line) ; C-n
(describe-function #'goto-line) ; M-g g, M-g M-g
(describe-function #'move-to-column) ; M-g TAB
(describe-function #'goto-char) ; M-g c
```

```

(describe-function #'backward-char) ; C-b
(describe-function #'forward-char) ; C-f
(describe-function #'backward-word) ; M-b
(describe-function #'forward-word) ; M-f
(describe-function #'move-beginning-of-line) ; C-a, <home>
(describe-function #'move-end-of-line) ; C-e, <end>
(describe-function #'backward-sentence) ; M-a
(describe-function #'forward-sentence) ; M-e
(describe-function #'backward-paragraph) ; M-{
(describe-function #'forward-paragraph) ; M-}

(describe-function #'backward-up-list) ; C-M-u
(describe-function #'backward-list) ; C-M-p
(describe-function #'forward-list) ; C-M-n
(describe-function #'backward-sexp) ; C-M-b
(describe-function #'forward-sexp) ; C-M-f

(describe-function #'recenter) ; (default unbound)
(describe-function #'save-excursion) ; (default unbound)

;; Regular expressions:
;; \| Alternative
;; \{ \} Repetition
;; \` \' Beginning and end of buffer
;; < > Beginning and end of word
;; \b Beginning, or end of word
;; \B Not at beginning, or end of word
;; \_< \_> Beginning and end of symbol
;; *?, +?, ?? Non-greedy *, +, ?
;; \( \) Capturing group
;; \(?: \) Shy capturing group
;; \1 to \9 Insert text from group
;; \#1 to \#9 Insert text from group as an integer
;; \? Prompt user for text input
;; \# Insert number incremented from 0 after every match
;; \& Insert whole match
;; \,(function) Call elisp function in replace portion of regexp
;; \w Word-constituent character
;; \W Not Word-constituent character
;; \sCODE Match character from code table
;; \Scode Do not match character from code table
;; Code table (can change depending on major mode):
;; - Whitespace
;; w Alnum
;; - Alnum with extra symbols: _ ! and others
;; . Punctuation characters
;; ( and ) Grouped pair: () [] {}
;; " String characters: " '
;; < and > Open/close comment characters

;; Search and replace
(describe-function #'search-backward) ; (default unbound)
(describe-function #'re-search-backward) ; (default unbound)
(describe-function #'search-forward) ; (default unbound)
(describe-function #'re-search-forward) ; (default unbound)

;; Inside I-search: M-< first match, M-> last match,
;; M-v previous off-screen match, C-v next off-screen match
(customize-variable 'isearch-allow-motion) ; turn on this option
(describe-variable 'isearch-motion-changes-direction)
(describe-variable 'isearch-yank-on-move)

```

```

(describe-function #'multi-isearch-buffers) ; (default unbound)
(describe-function #'multi-isearch-buffers-regexp) ; (default unbound)
(describe-function #'multi-isearch-files) ; (default unbound)
(describe-function #'multi-isearch-files-regexp) ; (default unbound)

(describe-function #'how-many) ; (default unbound)
(describe-function #'isearch-backward-regexp) ; C-M-r
(describe-function #'isearch-forward-regexp) ; C-M-s

(describe-variable 'search-highlight-submatches)
(describe-variable 'query-replace-highlight-submatches)
(describe-function #'isearch-highlight-regexp) ; M-s h r (inside I-search)
(describe-function #'isearch-highlight-lines-matching-regexp) ; M-s h l (inside)

(describe-function #'isearch-backward) ; C-r
(describe-function #'isearch-repeat-backward) ; C-r, C-M-r (inside I-search)
(describe-function #'isearch-end-of-buffer) ; M-s M-> (inside I-search)
(describe-function #'isearch-forward) ; C-s
(describe-function #'isearch-repeat-forward) ; C-s, C-M-s (inside I-search)
(describe-function #'isearch-beginning-of-buffer) ; M-s M-< (inside I-search)

(describe-function #'word-search-backward) ; M-s w C-r RET
(describe-function #'word-search-forward) ; M-s w RET
(describe-function #'isearch-forward-word) ; M-s w
(describe-function #'isearch-forward-symbol) ; M-s _
(describe-function #'isearch-forward-symbol-at-point) ; M-s .
;; When region is active, `isearch-forward-thing-at-point' acts on that region
;; just like # and * in Vim. If region is not active, it uses heuristic.
(describe-variable 'isearch-forward-thing-at-point)
(describe-function #'isearch-forward-thing-at-point) ; M-s M-.

(describe-function #'isearch-yank-until-char) ; C-M-z (inside I-search)
(describe-function #'isearch-yank-char) ; C-M-y (inside I-search)
(describe-function #'isearch-yank-word-or-char) ; C-w (inside I-search)
(describe-function #'isearch-yank-symbol-or-char) ; C-M-w (inside I-search)
(describe-function #'isearch-yank-line) ; M-s C-e (inside I-search)
(describe-function #'isearch-yank-kill) ; C-y (inside I-search)
(describe-function #'isearch-yank-pop) ; M-y (inside I-search)
(describe-function #'isearch-yank-pop-only) ; M-y (inside I-search)
(describe-function #'isearch-yank-x-selection) ; MOUSE2 (inside I-search)

(describe-function #'isearch-del-char) ; C-M-d (inside I-search)
(describe-function #'isearch-delete-char) ; BackSpace (inside I-search)
(describe-function #'isearch-complete) ; C-M-i, M-TAB (inside I-search)
(describe-function #'isearch-ring-retreat) ; M-p (inside I-search)
(describe-function #'isearch-ring-advance) ; M-n (inside I-search)
(describe-function #'isearch-edit-string) ; M-e (inside I-search)
;; When editing search string by pressing M-e, or MOUSE1 on search string:
(describe-function #'isearch-yank-char-in-minibuffer) ; C-f, <right>

(describe-function #'isearch-toggle-case-fold) ; M-c, M-s c (inside I-search)
(describe-function #'isearch-toggle-char-fold) ; M-s ' (inside I-search)
(describe-function #'isearch-toggle-regexp) ; M-r, M-s r (inside I-search)
(describe-function #'isearch-toggle-word) ; M-s w (inside I-search)
(describe-function #'isearch-toggle-symbol) ; M-s _ (inside I-search)
(describe-variable 'search-whitespace-regexp)
(describe-function #'isearch-toggle-lax-whitespace) ; M-s SPC (inside I-search)
(describe-function #'isearch-toggle-invisible) ; M-s i (inside I-search)
(describe-variable 'search-invisible)

```

```
(describe-function #'isearch-toggle-input-method) ; C-\ (inside I-search)
(describe-function #'isearch-toggle-specified-input-method) ; C-^ (in I-search)
(describe-function #'isearch-transient-input-method) ; C-x \ (inside I-search)
(describe-function #'isearch-char-by-name) ; C-x 8 RET (inside I-search)
(describe-function #'isearch-quote-char) ; C-q (inside I-search)
(describe-function #'isearch-printing-char) ; C-j (inside I-search)
```

```
(describe-function #'isearch-abort) ; C-g, C-g C-g (inside I-search)
(describe-function #'isearch-cancel) ; ESC ESC ESC
(describe-function #'isearch-exit) ; RET (inside I-search)
```

```
(describe-variable 'replace-lax-whitespace)
(describe-variable 'replace-regexp-lax-whitespace)
(describe-variable 'case-fold-search)
(describe-variable 'search-upper-case)
(describe-variable 'case-replace)
(describe-variable 'replace-char-fold)
(describe-variable 'query-replace-show-replacement)
(describe-function #'query-replace) ; M-%
(describe-function #'isearch-query-replace) ; M-% (inside I-search)
(describe-function #'query-replace-regexp) ; C-M-%
(describe-function #'isearch-query-replace-regexp) ; C-M-% (inside I-search)
(describe-function #'replace-string) ; (default unbound)
(describe-function #'replace-regexp) ; (default unbound)
(describe-function #'delete-duplicate-lines) ; (default unbound)
(describe-function #'keep-lines) ; (default unbound)
(describe-function #'flush-lines) ; (default unbound)
(describe-function #'kill-matching-lines) ; (default unbound)
(describe-function #'copy-matching-lines) ; (default unbound)
```

```
(describe-function #'occur) ; M-s o
(describe-function #'isearch-occur) ; M-s o (inside I-search)
(describe-function #'multi-occur) ; (default unbound)
(describe-function #'multi-occur-in-matching-buffers) ; (default unbound)
(describe-function #'imenu) ; M-i (default unbound)
```

;; Highlight

```
(describe-function #'highlight-changes-mode) ; (default unbound)
(describe-function #'highlight-lines-matching-regexp) ; M-s h l
(describe-function #'highlight-phrase) ; M-s h p
(describe-function #'highlight-symbol-at-point) ; M-s h .
(describe-function #'highlight-regexp) ; M-s h r
(describe-function #'unhighlight-regexp) ; M-s h u
(describe-function #'global-hi-lock-mode) ; (default unbound)
(describe-function #'hi-lock-mode) ; (default unbound)
(describe-function #'hi-lock-write-interactive-patterns) ; M-s h w
;; Enable `global-hi-lock-mode' before visiting file, or `hi-lock-find-patterns'
;; will not work. Patterns must be placed at the beginning of the file.
(describe-function #'hi-lock-find-patterns) ; M-s h f
```

;; Cut, copy, paste, select region

```
(describe-function #'backward-delete-char-untabify) ; BackSpace (in prog-mode)
(describe-function #'delete-backward-char) ; BackSpace (in text-mode)
(describe-function #'delete-forward-char) ; Delete
(describe-function #'delete-char) ; C-d
(describe-variable 'select-enable-clipboard)
(describe-variable 'save-interprogram-paste-before-kill)
(describe-function #'append-next-kill) ; C-M-w
(describe-function #'backward-kill-word) ; C-BackSpace, M-BackSpace
(describe-function #'kill-word) ; M-d
(describe-function #'zap-to-char) ; M-z
(describe-function #'zap-up-to-char) ; C-M-z (default unbound)
(describe-variable 'kill-transform-function)
```

```

(describe-function #'kill-line) ; C-k
(describe-function #'kill-whole-line) ; C-S-BackSpace
(describe-function #'backward-kill-sentence) ; C-x BackSpace
(describe-function #'kill-sentence) ; M-k
(describe-function #'backward-kill-sexp) ; ESC C-BackSpace, ESC C-Delete
(describe-function #'kill-sexp) ; C-M-k
(describe-function #'backward-kill-paragraph) ; (default unbound)
(describe-function #'kill-paragraph) ; (default unbound)

(describe-function #'transient-mark-mode) ; (default unbound)
(describe-function #'lost-selection-mode) ; (default unbound)
(describe-variable 'shift-select-mode)
(describe-variable 'highlight-nonselected-windows)
(describe-function #'mark-whole-buffer) ; C-x h
(describe-function #'mark-page) ; C-x C-p
(describe-function #'mark-paragraph) ; M-h
(describe-function #'mark-defun) ; C-M-h
(describe-function #'mark-sexp) ; C-M-SPC, C-M-@
(describe-function #'mark-end-of-sentence) ; (default unbound)
(describe-function #'mark-word) ; M-@
(describe-function #'set-mark-command) ; C-SPC, C-@
(describe-function #'exchange-point-and-mark) ; C-x C-x
(describe-function #'pop-global-mark) ; C-x C-SPC, C-x C-@

(describe-function #'delete-selection-mode) ; (default unbound)
(describe-function #'delete-active-region) ; (default unbound)
(describe-function #'delete-region) ; C-x m (default unbound)
(describe-function #'kill-region) ; C-w
(describe-function #'kill-ring-save) ; M-w
(describe-function #'yank) ; C-y
(customize-variable 'yank-pop-change-selection)
(describe-function #'yank-pop) ; M-y

(customize-variable 'mouse-yank-at-point)
(describe-function #'mouse-drag-region) ; MOUSE1 (hold and drag)
(describe-function #'mouse-set-point) ; MOUSE1
(describe-function #'mouse-save-then-kill) ; MOUSE3
(describe-function #'mouse-yank-primary) ; MOUSE2
(describe-function #'mouse-drag-secondary) ; M-MOUSE1 (hold and drag)
(describe-function #'mouse-start-secondary) ; M-MOUSE1
(describe-function #'mouse-secondary-save-then-kill) ; M-MOUSE3
(describe-function #'mouse-yank-secondary) ; M-MOUSE2

(describe-function #'mouse-drag-region-rectangle) ; C-M-MOUSE1 (drag)
(describe-function #'cua-rectangle-mark-mode) ; (default unbound)
(describe-function #'rectangle-mark-mode) ; C-x SPC
(describe-function #'rectangle-exchange-point-and-mark) ; C-x C-x (in rectangle)
(describe-function #'rectangle-number-lines) ; C-x r N
(describe-function #'string-insert-rectangle) ; (default unbound)
(describe-function #'string-rectangle) ; C-x r t
(describe-function #'open-rectangle) ; C-x r o
(describe-function #'clear-rectangle) ; C-x r c
(describe-function #'delete-whitespace-rectangle) ; (default unbound)
(describe-function #'delete-rectangle) ; C-x r d
(describe-function #'kill-rectangle) ; C-x r k
(describe-function #'copy-rectangle-as-kill) ; C-x r M-w
(describe-function #'yank-rectangle) ; C-x r y

;; Case
(describe-function #'capitalize-word) ; M-c
(describe-function #'downcase-word) ; M-l
(describe-function #'upcase-word) ; M-u

```

```

(describe-function #'upcase-initials-region) ; (default unbound)
(describe-function #'capitalize-region) ; (default unbound)
(describe-function #'downcase-region) ; C-x C-l
(describe-function #'upcase-region) ; C-x C-u

(describe-function #'upcase-initials) ; (default unbound)
(describe-function #'capitalize) ; (default unbound)
(describe-function #'downcase) ; (default unbound)
(describe-function #'upcase) ; (default unbound)

;; Transpose
(describe-function #'transpose-chars) ; C-t
(describe-function #'transpose-words) ; M-t
(describe-function #'transpose-lines) ; C-x C-t
(describe-function #'transpose-sentences) ; (default unbound)
(describe-function #'transpose-paragraphs) ; (default unbound)
(describe-function #'transpose-sexps) ; C-M-t
(describe-function #'transpose-regions) ; (default unbound)

(describe-function #'transpose-subr) ; (default unbound)
(describe-function #'transpose-subr-1) ; (default unbound)

;; Indentation, whitespace
(describe-function #'indent-region) ; C-M-\
(describe-function #'indent-rigidly) ; C-x TAB
(describe-function #'indent-for-tab-command) ; TAB
(describe-function #'electric-newline-and-maybe-indent) ; C-j
(describe-function #'newline) ; RET
(describe-function #'open-line) ; C-o
(describe-function #'split-line) ; C-M-o
(describe-function #'delete-blank-lines) ; C-x C-o
(describe-function #'delete-indentation) ; M-^
(describe-function #'just-one-space) ; M-SPC (in Emacs 28.2)
(describe-function #'cycle-spacing) ; M-SPC (in Emacs 30.1)
(describe-function #'delete-horizontal-space) ; M-\
(describe-function #'delete-to-left-margin) ; (default unbound)
(describe-variable 'show-trailing-whitespace)
(describe-variable 'delete-trailing-lines)
(describe-function #'delete-trailing-whitespace) ; (default unbound)

(describe-function #'set-selective-display) ; C-x $

;; Comments
(describe-variable 'comment-column)
(describe-function #'comment-indent) ; (default unbound)
(describe-function #'default-indent-new-line) ; M-j, C-M-j
(describe-function #'comment-line) ; C-x C-;
(describe-function #'comment-box) ; (default unbound)
(describe-function #'comment-dwim) ; M-;

;; Registers
(describe-function #'point-to-register) ; C-x r SPC, C-x r C-SPC, C-x r C-@
(describe-function #'copy-rectangle-to-register) ; C-x r r
(describe-function #'copy-to-register) ; C-x r s, C-x r x
(describe-function #'number-to-register) ; C-x r n
(describe-function #'increment-register) ; C-x r +

(describe-function #'insert-register) ; C-x r g, C-x r i

(describe-function #'set-register) ; (default unbound)
(describe-variable 'register-separator)
;; (setq register-separator ?+) ; example
;; (set-register register-separator "\n\n") ; example

```

```

(describe-function #'prepend-to-register) ; (default unbound)
(describe-function #'append-to-register) ; (default unbound)

;; (set-register ?s '(buffer . "**scratch**")) ; example
;; (set-register ?f '(file . "/tmp/file.el")) ; example

(describe-function #'frameset-to-register) ; C-x r f
(describe-function #'window-configuration-to-register) ; C-x r w

(describe-function #'jump-to-register) ; C-x r j

(describe-variable 'register-use-preview)
(describe-variable 'register-preview-delay)
(describe-function #'view-register) ; (default unbound)

(describe-function #'savehist-mode) ; (default unbound)
(describe-function #'desktop-save-mode) ; (default unbound)

;; Repeat last action (kill, yank, etc.), undo, redo
(describe-function #'repeat) ; C-x z
(describe-function #'undo) ; C-/, C-x u, C-_
(describe-variable 'undo-no-redo)
(customize-variable 'undo-no-redo)
(describe-function #'undo-only) ; (default unbound, same as `undo-no-redo' t)
(describe-function #'undo-redo) ; C-?, C-M-_

;; Bookmarks
(describe-function #'list-bookmarks) ; C-x r l (alias for bookmark-bmenu-list)
(describe-function #'bookmark-bmenu-list) ; C-x r l
(describe-function #'bookmark-set) ; C-x r m
(describe-function #'bookmark-set-no-overwrite) ; C-x r M
(describe-function #'bookmark-jump) ; C-x r b
(customize-variable 'bookmark-save-flag) ; set it to 1
(describe-function #'bookmark-save) ; (default unbound)
(describe-function #'bookmark-delete) ; (default unbound)
(describe-function #'bookmark-insert-location) ; (default unbound)
(describe-function #'bookmark-insert) ; (default unbound)

;; Buffers, windows, Emacs
(describe-function #'suspend-frame) ; C-z, C-x C-z
(describe-function #'make-frame-command) ; C-x 5 2
(describe-function #'display-buffer-other-frame) ; C-x 5 C-o
(describe-function #'switch-to-buffer-other-frame) ; C-x 5 b
(describe-function #'other-frame) ; C-x 5 o
(describe-function #'delete-other-frames) ; C-x 5 1
(describe-function #'delete-frame) ; C-x 5 0
(describe-function #'split-window-below) ; C-x 2
(describe-function #'split-window-right) ; C-x 3
(describe-function #'other-window) ; C-x o, M-o
(describe-function #'delete-other-windows) ; C-x 1
(describe-function #'delete-window) ; C-x 0
(describe-function #'quit-window) ; q (in special buffers)

(describe-function #'mouse-delete-other-windows) ; MOUSE2 (click on mode-line)
(describe-function #'mouse-delete-window) ; MOUSE3 (click on mode-line)

(describe-function #'mouse-select-window) ; MOUSE1 (click on mode-line)
(describe-function #'mouse-drag-mode-line) ; MOUSE1 (drag mode-line)
(describe-function #'mouse-drag-vertical-line) ; MOUSE1 (drag vertical-line)

```



```

(describe-function #'shrink-window-if-larger-than-buffer) ; C-x -
(describe-function #'balance-windows) ; C-x +
(describe-function #'shrink-window) ; (default unbound)
(describe-function #'enlarge-window) ; C-x ^
(describe-function #'shrink-window-horizontally) ; C-x {
(describe-function #'enlarge-window-horizontally) ; C-x }

(describe-function #'kill-buffer-and-window) ; C-x 4 0
(describe-function #'kill-buffer) ; C-x k
(describe-function #'previous-buffer) ; C-x <left>, C-x C-<left>
(describe-function #'next-buffer) ; C-x <right>, C-x C-<right>
(describe-function #'pop-to-buffer) ; (default unbound)
(describe-function #'switch-to-buffer) ; C-x b
(describe-function #'view-echo-area-messages) ; C-h e (*Messages* buffer)
(describe-function #'rename-buffer) ; C-x x r
(describe-function #'rename-uniquely) ; C-x x u
(describe-function #'mouse-buffer-menu) ; C-MOUSE1
(describe-function #'list-buffers) ; unbound (default C-x C-b)
(describe-function #'buffer-menu) ; unbound (may be bound to C-x C-b)
(describe-function #'ibuffer) ; C-x C-b (default unbound)
(describe-function #'copy-to-buffer) ; (default unbound)
(describe-function #'prepend-to-buffer) ; (default unbound)
(describe-function #'append-to-buffer) ; (default unbound)
(describe-function #'clone-buffer) ; C-x x n
(describe-function #'insert-buffer) ; C-x x i
(describe-function #'erase-buffer) ; (default unbound)
(describe-function #'revert-buffer) ; g (in special buffers)
(describe-function #'revert-buffer-quick) ; C-x x g
(describe-function #'font-lock-update) ; C-x x f

(describe-function #'list-directory) ; C-x C-d
(describe-function #'dired) ; C-x d
(describe-function #'dired-jump) ; C-x C-j
(describe-function #'dired-jump-other-window) ; C-x 4 C-j
(describe-variable 'default-directory)
(describe-function #'append-to-file) ; (default unbound)
(describe-function #'insert-file-literally) ; (default unbound)
(describe-function #'insert-file) ; C-x i
(describe-function #'find-file-other-frame) ; C-x 5 f, C-x 5 C-f
(describe-function #'find-file-other-window) ; C-x 4 f, C-x 4 C-f
(describe-function #'find-alternate-file) ; C-x C-v
(describe-function #'find-file) ; C-x C-f
(describe-function #'find-file-literally) ; (default unbound)
(describe-function #'find-file-read-only) ; C-x C-r
(describe-function #'read-only-mode) ; C-x C-q
(describe-function #'recover-this-file) ; (default unbound)
(describe-function #'recover-file) ; (default unbound)
(describe-function #'write-file) ; C-x C-w
(describe-function #'save-buffer) ; C-x C-s
(describe-function #'save-some-buffers) ; C-x s

(describe-function #'yes-or-no-p) ; (default unbound)
(describe-function #'y-or-n-p) ; (default unbound)
(describe-variable 'confirm-kill-processes)
(describe-variable 'confirm-kill-emacs)
(describe-variable 'kill-emacs-hook)
(describe-function #'kill-emacs) ; (default unbound)
(describe-variable 'kill-emacs-query-functions)
(describe-function #'save-buffers-kill-emacs) ; (default unbound)
(describe-function #'save-buffers-kill-terminal) ; C-x C-M-c (default C-x C-c)

```

;; Minibuffer

```
(describe-function #'minibuffer-inactive-mode) ; (default unbound)
(describe-function #'minibuffer-mode) ; (default unbound)
(describe-function #'abort-minibuffers) ; C-g (in minibuffer)
(describe-function #'minibuffer-electric-default-mode) ; (default unbound)
(describe-variable 'completions-format)
(describe-variable 'completions-sort)
(describe-function #'minibuffer-completion-help) ; ? (in minibuffer)
(describe-function #'minibuffer-complete) ; TAB (in minibuffer)
(describe-function #'minibuffer-complete-word) ; SPC (in minibuffer)
(describe-function #'switch-to-completions) ; M-v, M-g M-c (in minibuffer)
(describe-function #'minibuffer-choose-completion) ; M-RET (in minibuffer)
(describe-function #'choose-completion) ; RET (in completion list buffer)
(describe-function #'kill-current-buffer) ; z (in completion list buffer)
(describe-function #'previous-history-element) ; M-p (in minibuffer)
(describe-function #'next-history-element) ; M-n (in minibuffer)
(describe-function #'previous-matching-history-element) ; M-r (in minibuffer)
(describe-function #'next-matching-history-element) ; M-s (in minibuffer)
(describe-function #'minibuffer-complete-and-exit) ; RET (in minibuffer)
```

;; Coding systems

```
(describe-function #'universal-coding-system-argument) ; C-x RET c
(describe-function #'revert-buffer-with-coding-system) ; C-x RET r
(describe-variable 'buffer-file-coding-system)
;; (setq-default buffer-file-coding-system 'utf-8-unix) ; example
(describe-function #'set-buffer-file-coding-system) ; C-x RET f
(describe-function #'set-file-name-coding-system) ; C-x RET F
(describe-function #'recode-region) ; (default unbound)
(describe-function #'set-keyboard-coding-system) ; C-x RET k
(describe-function #'set-terminal-coding-system) ; C-x RET t
(describe-variable 'x-select-request-type)
(describe-function #'set-selection-coding-system) ; C-x RET x
(describe-function #'set-next-selection-coding-system) ; C-x RET X
(describe-function #'set-buffer-process-coding-system) ; C-x RET p

(describe-function #'list-character-sets) ; (default unbound)
(describe-function #'view-hello-file) ; C-h h
(describe-function #'toggle-input-method) ; C-\ (set russian-computer)
(describe-function #'set-input-method) ; C-x RET C-\ (set russian-computer)
(describe-function #'activate-transient-input-method) ; C-x \
```

;; Macros

```
(describe-function #'kmacro-start-macro-or-insert-counter) ; F3
(describe-function #'kmacro-end-or-call-macro) ; F4
(describe-function #'apply-macro-to-region-lines) ; C-x C-k r
(describe-function #'kmacro-start-macro) ; C-x (, C-x C-k s, C-x C-k C-s
(describe-function #'kmacro-end-macro) ; C-x )
(describe-function #'kmacro-end-and-call-macro) ; C-x e
(describe-function #'kmacro-redisplay) ; C-x C-k d

(describe-function #'kmacro-cycle-ring-next) ; C-x C-k C-n
(describe-function #'kmacro-cycle-ring-previous) ; C-x C-k C-p
(describe-function #'kmacro-to-register) ; C-x C-k x
(describe-function #'kmacro-name-last-macro) ; C-x C-k n
(describe-function #'kmacro-bind-to-key) ; C-x C-k b
(describe-function #'edit-kbd-macro) ; C-x C-k e
(describe-function #'kmacro-step-edit-macro) ; C-x C-k SPC
(describe-function #'view-lossage) ; C-h l
(describe-function #'kmacro-edit-lossage) ; C-x C-k l
(describe-function #'insert-kbd-macro) ; (default unbound)
```

;; Version control

```
(describe-function #'vc-dir) ; C-x v d
(describe-function #'vc-ignore) ; C-x v G
(describe-function #'vc-register) ; C-x v i
(describe-function #'vc-next-action) ; C-x v v
(describe-function #'vc-update) ; C-x v +
(describe-function #'vc-push) ; C-x v P
(describe-function #'vc-revert) ; C-x v u
(describe-function #'vc-insert-headers) ; (default unbound)
(describe-function #'vc-print-root-log) ; C-x v L
(describe-function #'vc-print-log) ; C-x v l
(describe-function #'vc-log-incoming) ; C-x v I
(describe-function #'vc-log-outgoing) ; C-x v O
(describe-function #'vc-update-change-log) ; C-x v a
(describe-function #'vc-root-diff) ; C-x v D
(describe-function #'vc-diff) ; C-x v =
(describe-function #'vc-revision-other-window) ; C-x v ~
(describe-function #'vc-rename-file) ; (default unbound)
(describe-function #'vc-annotate) ; C-x v g
(describe-function #'vc-create-tag) ; C-x v s
(describe-function #'vc-retrieve-tag) ; C-x v r
```

;; Abbreviations

```
(describe-function #'inverse-add-global-abbrev) ; C-x a -, C-x a i g
(describe-function #'add-global-abbrev) ; C-x a g
(describe-function #'inverse-add-mode-abbrev) ; C-x a i l
(describe-function #'add-mode-abbrev) ; C-x a l, C-x a C-a, C-x a +
(describe-function #'expand-abbrev) ; C-x ', C-x a e, C-x a '
(describe-function #'expand-jump-to-next-slot) ; C-x a n
(describe-function #'expand-jump-to-previous-slot) ; C-x a p
(describe-function #'expand-region-abbrevs) ; (default unbound)
(describe-function #'unexpand-abbrev) ; (default unbound)
```

```
(describe-function #'dabbrev-expand) ; M-/
(describe-function #'dabbrev-completion) ; C-M-/
```

;; Spelling, spellcheck

```
(describe-variable 'ispell-program-name)
(describe-variable 'ispell-dictionary)
(describe-function #'ispell-change-dictionary) ; (default unbound)
(describe-function #'ispell-hunspell-add-multi-dic) ; (default unbound)
(describe-function #'ispell) ; (default unbound)
(describe-function #'ispell-buffer) ; (default unbound)
(describe-function #'ispell-region) ; (default unbound)
(describe-function #'ispell-comments-and-strings) ; (default unbound)
(describe-function #'ispell-comment-or-string-at-point) ; (default unbound)
(describe-function #'ispell-word) ; M-$
(describe-variable 'ispell-complete-word-dict)
(describe-function #'ispell-complete-word) ; (default unbound)
(describe-function #'completion-at-point) ; C-M-i, ESC TAB, M-TAB
(describe-function #'ispell-kill-ispell) ; (default unbound)
```

```
(describe-variable 'flyspell-check-changes)
(describe-function #'flyspell-mode) ; (default unbound)
(describe-function #'flyspell-prog-mode) ; (default unbound)
(describe-function #'flyspell-buffer) ; (default unbound)
(describe-function #'flyspell-region) ; (default unbound)
(describe-function #'flyspell-auto-correct-word) ; C-., ESC TAB, C-M-i
(describe-function #'flyspell-correct-word-before-point) ; C-c $
(describe-function #'flyspell-correct-word) ; MOUSE2
(describe-function #'context-menu-mode) ; (default unbound)
```

;; Miscellaneous

```
(describe-function #'gpm-mouse-mode) ; (default unbound)
(describe-function #'xterm-mouse-mode) ; (default unbound)

(describe-function #'menu-bar-open) ; F10
(describe-function #'toggle-frame-maximized) ; M-F10
(describe-function #'toggle-frame-fullscreen) ; F11
(describe-function #'menu-set-font) ; (default unbound)
(describe-function #'mouse-appearance-menu) ; S-MOUSE1
(describe-function #'mouse-wheel-text-scale) ; C-MouseScroll
(describe-function #'text-scale-adjust) ; C-x C-0, C-x C-- , C-x C-=, C-x C-+
(describe-function #'text-scale-mode) ; enabled/disabled by `text-scale-adjust'

(describe-function #'compare-windows) ; C-x f (default unbound)
(describe-function #'ediff) ; (default unbound)
(describe-function #'diff) ; (default unbound)

(describe-function #'compose-mail) ; C-x M-m (default C-x m)

(describe-function #'edit-tab-stops) ; (default unbound)
(describe-variable 'tab-stop-list)
(describe-function #'tab-to-tab-stop) ; unbound (default M-i)

(describe-function #'async-shell-command) ; M-&
(describe-function #'shell-command) ; M-!
(describe-function #'shell-command-on-region) ; M-|

(describe-function #'universal-argument) ; C-u
(describe-function #'negative-argument) ; ESC -, C--, M--, C-M--
(describe-function #'digit-argument) ; ESC 0..9 (also C, M and C-M)

(describe-variable 'fill-prefix)
(describe-function #'set-fill-prefix) ; C-x .
(describe-variable 'fill-column)
(describe-function #'set-fill-column) ; C-x M-f (default C-x f)
(describe-function #'fill-paragraph) ; M-q
(describe-function #'fill-forward-paragraph) ; (default unbound)
(describe-function #'fill-region) ; (default unbound)
(describe-function #'fill-region-as-paragraph) ; (default unbound)

;; Unbound functions (and some rarely used bound functions)
(describe-function #'auto-fill-mode)
(describe-function #'global-display-fill-column-indicator-mode)
(describe-function #'display-fill-column-indicator-mode)
(describe-function #'global-hl-line-mode)
(describe-function #'hl-line-mode)
(describe-function #'global-whitespace-mode)
(describe-function #'whitespace-mode)
(describe-variable 'whitespace-line-column) ; exists only in `whitespace-mode'
(describe-function #'line-number-mode)
(describe-function #'column-number-mode)
(describe-function #'size-indication-mode)
(describe-function #'global-display-line-numbers-mode)
(describe-function #'display-line-numbers-mode)
(describe-variable 'display-line-numbers-type)
(describe-variable 'display-line-numbers-current-absolute)
(customize-group 'display-line-numbers)
(describe-function #'display-time-mode)
(customize-variable 'display-time-24hr-format) ; only in `display-time-mode'
(describe-variable 'display-time-use-mail-icon) ; only in `display-time-mode'
(describe-function #'display-battery-mode)
(describe-variable 'battery-mode-line-format) ; only in `display-battery-mode'
(describe-function #'global-font-lock-mode)
```

```

(describe-function #'font-lock-mode)
(describe-function #'repeat-mode)
(describe-function #'describe-repeat-maps) ; available only in `repeat-mode'
(describe-function #'follow-mode)
(describe-function #'view-mode)
(describe-function #'view-buffer)
(describe-function #'view-file)
(describe-function #'View-search-regexp-backward) ; \ (in view-mode)
(describe-function #'View-search-regexp-forward) ; / (in view-mode)
(describe-function #'View-search-last-regexp-backward) ; p (in view-mode)
(describe-function #'View-search-last-regexp-forward) ; n (in view-mode)
(describe-function #'toggle-truncate-lines) ; C-x x t
(describe-function #'global-visual-line-mode)
(describe-function #'visual-line-mode)
(describe-function #'previous-logical-line)
(describe-function #'next-logical-line)
(describe-function #'beginning-of-visual-line)
(describe-function #'end-of-visual-line)
(describe-function #'kill-visual-line)
(describe-function #'global-word-wrap-whitespace-mode)
(describe-function #'word-wrap-whitespace-mode)
(describe-function #'global-visual-wrap-prefix-mode)
(describe-function #'visual-wrap-prefix-mode)
(describe-function #'variable-pitch-mode)
(describe-function #'buffer-face-mode)
(describe-function #'fido-mode)
(describe-function #'fido-vertical-mode)
(describe-function #'icomplete-fido-ret) ; RET (in `fido-mode')
(describe-function #'icomplete-force-complete-and-exit) ; C-j (in `fido-mode')
(describe-function #'icomplete-fido-exit) ; M-j (in `fido-mode')

(describe-function #'redraw-display)
(describe-function #'redraw-frame)

(describe-function #'eval-buffer)
(describe-function #'eval-region)
(describe-function #'eval-defun)

(describe-function #'mark-beginning-of-buffer) ; (does not activate the mark)
(describe-function #'mark-end-of-buffer) ; (does not activate the mark)

(describe-function #'forward-comment)
(describe-function #'forward-line)
(describe-function #'forward-same-syntax)
(describe-function #'forward-symbol)
(describe-function #'forward-thing)
(describe-function #'forward-to-indentation)
(describe-function #'forward-to-word)
(describe-function #'forward-visible-line)
(describe-function #'forward-whitespace)
(describe-function #'forward-word-strictly)

(describe-function #'backward-delete-char)
(describe-function #'backward-prefix-chars)
(describe-function #'backward-to-indentation)
(describe-function #'backward-to-word)
(describe-function #'backward-word-strictly)

(describe-function #'clipboard-kill-region)
(describe-function #'clipboard-kill-ring-save)
(describe-function #'comment-kill)
(describe-function #'copy-region-as-kill)
(describe-function #'current-kill)

```

```

(describe-function #'isearch-yank-from-kill-ring)
(describe-function #'kill-all-abbrevs)
(describe-function #'kill-all-local-variables)
(describe-function #'kill-append)
(describe-function #'kill-backward-chars)
(describe-function #'kill-backward-up-list)
(describe-function #'kill-buffer-ask)
(describe-function #'kill-buffer-if-not-modified)
(describe-function #'kill-comment)
(describe-function #'kill-forward-chars)
(describe-function #'kill-local-variable)
(describe-function #'kill-matching-buffers)
(describe-function #'kill-new)
(describe-function #'kill-process)
(describe-function #'kill-some-buffers)
(describe-function #'kill-this-buffer)
(describe-function #'mouse-kill)
(describe-function #'mouse-kill-ring-save)
(describe-function #'mouse-kill-secondary)
(describe-function #'mouse-save-then-kill-delete-region)
(describe-function #'read-from-kill-ring)
(describe-function #'yank-from-kill-ring)

(describe-function #'sort-lines)
(describe-function #'sort-fields)
(describe-function #'sort-numeric-fields)
(describe-function #'sort-columns)
(describe-function #'sort-paragraphs)
(describe-function #'sort-regexp-fields)

(describe-function #'align)
(describe-function #'align-current)
(describe-function #'align-regexp)

(describe-function #'compile)
(describe-function #'shell)
(describe-function #'eshell)
(describe-function #'ansi-term)
(describe-function #'proced)

(describe-function #'current-time)
(describe-function #'time-convert)
(describe-function #'format-time-string)

(describe-function #'format)
(describe-function #'format-message)
(describe-function #'message) ; uses `format-message' to print strings

(describe-function #'pwd)
(describe-variable 'path-separator)
(describe-function #'cd)
(describe-function #'file-exists-p)
(describe-function #'byte-compile-file)
(describe-function #'native-compile)
(describe-function #'file-attributes)
(describe-function #'directory-files-and-attributes)
(describe-function #'directory-files)

(describe-function #'grep)
(describe-function #'grep-find)
(describe-function #'lgrep)
(describe-function #'rgrep)
(describe-function #'rzgrep)

```

```

(describe-function #'find-dired)
(describe-function #'find-name-dired)
(describe-function #'find-grep-dired)
(describe-function #'find-lisp-find-dired)

(describe-function #'fundamental-mode)
(describe-function #'enriched-mode)
(describe-function #'text-mode)
(describe-function #'electric-quote-mode)
(describe-function #'paragraph-indent-text-mode)
(describe-function #'indented-text-mode)
(describe-variable 'adaptive-fill-mode)

(describe-function #'prog-mode)
(describe-function #'electric-indent-mode)
(describe-function #'electric-indent-local-mode)
(describe-function #'subword-mode)
(describe-function #'emacs-lisp-mode)
(describe-function #'lisp-mode)
(describe-function #'tcl-mode)
(describe-function #'awk-mode)
(describe-function #'m4-mode)
(describe-function #'python-mode)

(describe-function #'reveal-mode)
(describe-function #'outline-mode)
(describe-function #'org-mode)

(describe-function #'speedbar-mode)
(describe-function #'windmove-mode)
(customize-variable 'windmove-default-keybindings) ; set to Shift
(customize-variable 'windmove-wrap-around) ; toggle on
(describe-function #'menu-bar-mode)
(describe-function #'tool-bar-mode)
(describe-function #'tab-bar-mode)
(describe-function #'scroll-bar-mode)
(describe-function #'horizontal-scroll-bar-mode)

(describe-function #'buffer-list)
(describe-function #'window-list)
(describe-function #'frame-list)

(describe-function #'modify-frame-parameters)
(describe-function #'set-frame-parameter)
(describe-function #'set-frame-height)
(describe-function #'set-frame-width)

(describe-variable 'user-real-login-name)
(describe-variable 'user-login-name)
(describe-variable 'initial-environment)
(describe-variable 'process-environment)
(describe-function #'setenv)
(describe-function #'getenv)
(describe-variable 'exec-path)
(describe-function #'quote)
(describe-function #'add-to-list)
(describe-variable 'custom-theme-load-path)
(describe-variable 'custom-theme-directory)
(describe-variable 'user-emacs-directory)
(describe-variable 'user-init-file)

```

```

(describe-function #'facemenu-menu) ; C-MOUSE2
(describe-function #'list-faces-display)
(describe-function #'list-colors-display)
(describe-function #'set-face-foreground)
(describe-function #'set-face-background)
(describe-function #'add-hook)
(describe-function #'customize)
(describe-function #'customize-face)
(describe-function #'customize-variable)
(describe-function #'customize-group)
(describe-function #'customize-create-theme)
(describe-function #'customize-themes)
(describe-variable 'w32-follow-system-dark-mode)

(describe-function #'always) ; Do nothing and return t
(describe-function #'ignore) ; Do nothing and return nil

(describe-variable 'indicate-empty-lines)
(customize-variable 'ring-bell-function) ; choose Silent
(customize-variable 'visible-bell) ; set it to nil
(customize-variable 'make-pointer-invisible) ; set it to nil
(customize-variable 'blink-cursor-mode) ; set it to nil
(customize-variable 'visible-cursor) ; has effect only on text terminals

(describe-variable 'frame-background-mode)
(describe-variable 'default-frame-alist)
(customize-variable 'default-frame-alist)

;; Recursion (to create and use recursive elisp functions):
(customize-variable 'max-specpdl-size)
(customize-variable 'max-lisp-eval-depth)
;; If you increase the value of `max-specpdl-size' too much, Emacs could run
;; out of memory trying to make the stack bigger. If you increase the value of
;; `max-lisp-eval-depth' too much, Emacs could overflow the real C stack, and
;; crash. I use the value of 8192 for both variables.

(describe-function #'deftheme)
(describe-function #'provide)
(describe-function #'require)
(describe-function #'load)
(describe-function #'load-file)
(describe-function #'load-theme)

(describe-function #'makunbound)
(describe-function #'fmakunbound)
(describe-function #'defun)
(describe-function #'defvar)
(describe-function #'defcustom)
(describe-function #'custom-set-variables)
(describe-variable 'eval-expression-print-length)
;; (custom-set-variables '(eval-expression-print-length nil)) ; example
(describe-function #'setq-default)
(describe-function #'setq)
(describe-function #'set-variable)
(describe-function #'type-of)

(describe-function #'kbd)
(describe-function #'global-unset-key)
(describe-function #'global-set-key)
(describe-function #'local-unset-key)
(describe-function #'local-set-key)
(describe-function #'define-key)

```