Emacs support for AppleScript

Description	Keystroke	Function	Note
AppleScript Support	PEL provides basic support for macOS AppleScript. • PEL enables AppleScript editing and execution support when the **pel-use-applescript** user option is set to **t**. • PEL provides help for editing AppleScript files via the 📦 **apples-mode external package**. • On macOS systems, PEL also provide a key hydra you can use to narrate any text in buffer out-loud using macOS synthesized voices and commands to execute AppleScript commands and programs. These would normally only be supported by Emacs running in Graphics mode, but PEL provides a simple functions to use them inside Emacs running in Terminal (TTY) mode as well.		
Customize PEL AppleScript Support (See also: ∑ Customize)	• `<f11> <f1> SPC a` • `<f12> <f1>`	(**pel-cfg-pkg-applescript** &optional OTHER-WINDOW)	Customize PEL AppleScript support. • If OTHER-WINDOW is non-nil (use `C-u`), display in another window. • The `<f12> <f1>` binding is available when point is in a buffer visiting an AppleScript file.
Open an AppleScript scratchpad buffer	**<f12> s**	(**apples-open-scratch**)	Open scratch buffer for AppleScript editing. It is rendered properly and allows you to write AppleScript code without committing it to a file.
Text narration	PEL provides a set of commands to narrate text taken from a prompt or directly from the current buffer at point: • word • sentence • paragraph • region When taking large section of text a regular expression filter can be applied to remove characters that would cause problem in the narration. The regular expression is stored inside the **pel-narration-filter-regexp** constant. The following commands are available.		
Narrate the text specified at prompt	`<f8> t`	(**pel-say** TEXT &optional FILTER-CHARS-REGEXP)	Say TEXT out-loud. Prompts for the text. • Use the Apple osascript to narrate the text using synthesized voice. • Quotes are not allowed inside the text. • Furthermore the optional FILTER-CHARS-REGEXP can be used to exclude text from the narration.
Read the word at point out-loud	`<f8> w`	(**pel-say-word**)	Say word at point out-loud and move to next word.
Read the sentence at point out-loud	`<f8> s`	(**pel-say-sentence**)	Say sentence at point out-loud and move to next sentence.
Read the paragraph at point out-loud	`<f8> p`	(**pel-say-paragraph**)	Say paragraph at point out-loud and move to next paragraph.
Read the selected text out-loud	`<f8> r`	(**pel-say-region** START END)	Say text between region's START end END out-loud.
Using a hydra to narrate several words, sentences, paragraphs	• 📦 With PEL user option **pel-use-hydra** set to **t**, PEL activates the **hydra external package** also creates a Hydra set of keys that can help read several words, sentences or paragraphs in the current buffer easily. • To start, first hit the `<f8> .` key sequence to activate the hydra, then type one of the following letters to read text out-loud and to move point.		
Activate the narration Hydra	`<f8> .`	(**pel-hydra-narrate**)	Activates the hydra with the following keys.
Read the word at point out-loud	`w`	(**pel-say-word**)	Say word at point out-loud and move to next word.
Read the sentence at point out-loud	`s`	(**pel-say-sentence**)	Say sentence at point out-loud and move to next sentence.
Read the paragraph at point out-loud	`p`	(**pel-say-paragraph**)	Say paragraph at point out-loud and move to next paragraph.
Move to previous word	`b`	(**backward-word** &optional ARG)	Move backward until encountering the beginning of a word.
Move to next word	`n`	(**pel-forward-word-start**)	Move point forward to beginning of next word.