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. • When editing an AppleScript file (a file with the .applescript or .scpt extension) the buffer is automatically placed in apples-mode and the **`<f12>`** key is used as the mode-specific prefix . The **`<f11> SPC`** is the global mode prefix for the AppleScript commands.		
Open local copy of this PDF file. See also: ∑ **Help/Info**	**`<f12> <f1>`**	**(pel-help-pdf)**	Open the PEL PDF file(s) for the current context: applescript. It opens the local copy of this file.
Customize PEL AppleScript support. See also: ∑ **Customize**	• **`<f11> <f2> SPC a`** • **`<f12> <f2>`**	**(pel-customize-pel** &optional OTHER-WINDOW)	Open the PEL customize group(s) for the current context: applescript. Use this to open to change PEL user option variables the activate and control the various Apple script features such as the name of the narrator voice. • When a prefix argument (like **C-u**) opens the buffer inside another window.
Customize Emacs built-in AppleScript support See also: ∑ **Customize**	**`<f12> <f3>`**	**(pel-customize-library** &optional OTHER-WINDOW)	Customize Emacs **apples** group which includes: abbrev-mode, dynamic-abbreviations,expand,hippie-expand, quickurl. • When a prefix argument (like **C-u**) opens the buffer inside another window.
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.
	
	• 🔲 📦 With PEL user option **pel-use-hydra** set to **t**, PEL activates the **hydra external package** and also creates a Hydra set of keys to help speed up narration. These keys are identified in the table below.
	• To start this Hydra, hit the **`<f7><f8>`** keys, then hit one of the following keys once or several times.
	• The keys that are in the PEL window hydra are all identified below with a **`<f7><f8>`** prefix, but when **pel-use-hydra** is set to **t**, after typing **`<f7><f8>`** once, you can hit several other window hydra keys without typing the prefix again.
	• While the PEL narrate hydra is active, you can also type other commands; the hydra will stay active. You can, for example, select a region and then type **R** to read it aloud.
	• To cancel the Hydra hit the **`<f7>`** key again.
	• A snapshot of the narrate hydra menu that shows up in the mini buffer area as soon as one of its keys is pressed is shown below.
	
	⚠️ 👆 If the pel-use-hydra is nil, then you must type the prefix before every narration command. Using the narration hydra is easier and more flexible.
	
	• The narration commands available are shown below.
	• The narrate hydra includes some navigation commands to help move around while being in narration mode. These are only showing inside the narrate hydra help menu.
	• The name of the PEL window hydra commands are not listed below. They all have a name that begins with **pel-∑narrate/** and ends with the same name as the command function listed in the Function column. For example, **pel-∑narrate/pel-say** is bound to **`<f7><f8> t`**.
	
	👆 When processing text, PEL filters out the comments strings for the buffer's major mode. It also performs a set of text translation to help the narration. These are currently not configurable.		

PEL Narration Hydra	<pre>Config
	───────────────
	`<f2>`: customize</pre>
		<pre>Read | Repeat | Type | Move to | End
		───────────── | ──────────── | ───────────── | ───────────────────── | ────────────
		w: word | **r**: last word | **t**: at prompt | **b**: previous word | **`<f7>`**: cancel
		s: sentence | | | **n**: next word |
		p: paragraph | | | **B**: previous sentence |
		R: region | | | **N**: next sentence |</pre>	

Narrate the text entered at prompt	**`<f7> <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	**`<f7> <f8> w`**	**(pel-say-word)**	Say word at point out-loud and move to next word.
Read the sentence at point out-loud	**`<f7> <f8> s`**	**(pel-say-sentence)**	Say sentence at point out-loud and move to next sentence.
Read the paragraph at point out-loud	**`<f7> <f8> p`**	**(pel-say-paragraph)**	Say paragraph at point out-loud and move to next paragraph.
Read the selected region of text out-loud	**`<f7> <f8> R`**	**(pel-say-region** START END)	Say text between region's START end END out-loud.