Smartparens Documentation

Release 1.8.0

Matúš Goljer

November 11, 2016

Contents

Contents:

CHAPTER 1

Automatic escaping

Smartparens can automatically escape quote characters (`'`, `"`) and the escape character \ (as determined by Emacs's syntax tables) when you wrap a region or insert them inside a string.

Autoescaping is specified on a per-pair basis. For convenience, there are two global options to enable or disable escaping after wrapping and after insertion of a pair.

User Option `sp-escape-wrapped-region`
If non-nil, escape special chars inside the just wrapped region.

User Option `sp-escape-quotes-after-insert`
If non-nil, escape string quotes if typed inside string.

1.1 Escaping with post hooks

To enable automatic escaping after wrapping or insertion it is necessary to add post-handlers on the pairs where you want to trigger the escaping. These are typically the `"` or `'` quote pairs. Escaping after wrapping is enabled by default.

The hooks are named after the global options—if the option is disabled the function does nothing.

Function `sp-escape-wrapped-region``()`
Escape quotes and special chars when a region is wrapped.

The wrapping is smart in the sense that it only escapes when necessary. The behaviour of wrapping a word (`bar`) which is already inside a string is summarized by the following table:

enclosing \ wrapped	‘	“
‘	‘foo \’bar\’ baz’	‘foo “bar” baz’
“	“foo ‘bar’ baz”	“foo \”bar\” baz”

The behaviour of wrapping existing text (`foo 'bar' baz` or `foo "bar" baz`) containing a string is summarized by the following table:

containing \ wrapped	‘	“
foo ‘bar’ baz	‘foo \’bar\’ baz’	“foo ‘bar’ baz”
foo “bar” baz	‘foo “bar” baz’	“foo \”bar\” baz”

Function `sp-escape-quotes-after-insert``()`
Escape quotes inserted via ‘sp-insert-pair’.

If auto-pairing is enabled and the pair was successfully inserted inside a string, this hook ensures that it is escaped properly if necessary. The same rules as for wrapping apply, that is, escaping only takes place when necessary. This hook is triggered on the `'insert` action, and so if a `:when` or `:unless` handler prevents pairing, no action is taken.

1.2 Escape action

Some users prefer not to pair quotes inside strings and instead only insert the single delimiter (this is the paredit behaviour). However, if the user disabled auto-pairing inside strings, that means the `sp-escape-quotes-after-insert` handler is never called and the quote will get inserted unescaped.

This situation can be handled by adding an `'escape` action on the pair. That tells smartparens to still escape the single inserted delimiter even if the insert action wasn't performed.

Having a separate action might seem extraneous, but it gives us better flexibility in defining the escaping rules to precisely match what is expected. Also, by the nature of the post-handlers, which are only run after the action is performed successfully, it is necessary to have an action separate from `'insert` which might get inhibited by other predicates.

As usual, actions are tried in sequence: `'wrap`, `'insert`, `'escape`, so if the pair is wrapped or inserted the escape action is skipped (and the escaping can be handled with the handlers from previous section).

The single-pair behaviour is summarized in the following table:

enclosing \ inserted	‘	“
‘foo | bar’	‘foo \’| bar’	‘foo “| bar’
“foo | bar”	“foo ‘| bar”	“foo \”| bar”

CHAPTER 2

Indices and tables

- genindex
- modindex
- search

Index