

codeEditorWidget

codeEditorWidget

```
new codeEditorWidget(x, y, language, themeopt) → {WebviewWidget}
```

Source: [widgets/codeEditorWidget.js, line 14](#)

Parameters:

Name	Type	Attributes	Default	Description
x	Integer			This is the Column that this widget should be added to.
y	Integer			This is the row of of the Column that this widget should be added to.
language	String			This is the language that CodeMirror should target when formatting the contents of this widget.
theme	String	<optional>	darcula	CodeMirror theme to use when formatting this editor. Any theme in the 'node_modules/codemirror/theme/' directory will work. Just pass the name of the css file and this does the rest.

Returns:

Returns a WebviewWidget, an instance of the Widget class which allows a user to display a remote webpage, or a local html file.

Type [WebviewWidget](#)

Extends

- [Widget](#)

Methods

```
getCol() → {Integer}
```

Source: [widgets/widget.js, line 89](#)

Inherited From: [Widget#getCol](#)

Get the position in columns(Left = 0 to Right = n) of this widget.

Returns:

colIndex - The column index of this Widget.

Type **Integer**

```
getElement() → {Element}
```

Source: [widgets/widget.js, line 81](#)

Inherited From: [Widget#getElement](#)

Get the dom element that represents this widget.

Returns:

element - The dom object for this widget.

Type **Element**

```
getRow() → {Integer}
```

Source: [widgets/widget.js, line 97](#)

Inherited From: [Widget#getRow](#)

Get the position in rows(Top = 0 to Bottom = n) of this widget.

Returns:

rowIndex - The row index of this Widget.

Type **Integer**

```
(async) init(configData) → {Promise.<any>}
```

Source: [widgets/codeEditorWidget.js, line 30](#)

Overrides: [Widget#init](#)

This function overrides the parent widget initialize function and creates a code editor to be displayed within this widget.

Parameters:

Name	Type	Description
<code>configData</code>	Object	This object contains important information about the state that this widget was in the last time the application closed. The exact value inside of the widget is passed back into it here. This object also contains information about what language this editor is editing.

Returns:

Type `Promise.<any>`

`postinit()`

Source: [widgets/codeEditorWidget.js, line 78](#)

This function is called after initialization has occurred on this widget, by this time all fields this widget references should be initialized.

`save() → {Object}`

Source: [widgets/codeEditorWidget.js, line 87](#)

This function overrides the parent widget save function. The save object returned contains the language, theme, and content of the code editor.

Returns:

Type `Object`

`setElement(element)`

Source: [widgets/widget.js, line 70](#)

Inherited From: [Widget#setElement](#)

Set the dom element that represents this widget.

Parameters:

Name	Type	Description
<code>element</code>	<code>Element</code>	The dom object for this widget.

Documentation generated by [JSDoc 3.5.5](#) on Sat Apr 13 2019 15:00:13 GMT-0400 (Eastern Daylight Time) using the [docdash](#) theme.