

# Package ‘aceEditor’

November 14, 2020

**Title** The 'Ace' Editor as a HTML Widget

**Version** 1.0.0

**Description** Wraps the 'Ace' editor in a HTML widget. The 'Ace' editor has support for many languages. It can be opened in the viewer pane of 'RStudio', and this provides a second source editor.

**License** GPL-3

**Encoding** UTF-8

**LazyData** true

**RoxygenNote** 7.1.1

**Imports** htmltools, htmlwidgets, reactR, rstudioapi, tools

**URL** <https://github.com/stla/aceEditor>

**BugReports** <https://github.com/stla/aceEditor/issues>

**NeedsCompilation** no

**Author** Stéphane Laurent [aut, cre],  
James Hriso [ctb, cph] ('react-ace' library),  
James Long and contributors [ctb, cph] ('Prettier' library),  
Zeb Zhao [ctb, cph] ('indent.js' library),  
jQuery contributors [ctb, cph] ('jQuery' library),  
Tristan Edwards [ctb, cph] ('sweetalert2' library),  
Limon Monte [ctb, cph] ('sweetalert2' library)

**Maintainer** Stéphane Laurent <laurent\_step@outlook.fr>

**Repository** CRAN

**Date/Publication** 2020-11-14 10:00:02 UTC

## R topics documented:

aceDiffEditor	2
aceEditor	3
getAceModes	5
getAceThemes	5

<b>Index</b>	<b>6</b>
--------------	----------

---

 aceDiffEditor

*Ace diff editor*


---

### Description

Open the Ace diff editor.

### Usage

```
aceDiffEditor(
  file1,
  file2,
  mode = NULL,
  theme = NULL,
  fontSize = 14,
  tabSize = NULL,
  autoCompletion = TRUE,
  snippets = FALSE,
  width = NULL,
  height = NULL,
  elementId = NULL
)
```

### Arguments

file1, file2	paths to files
mode	the language of the files; if NULL, the mode is guessed from the extension of the first file; run <a href="#">getAceModes</a> to get the list of available modes
theme	the theme of the editor; if NULL, the theme is set to the theme currently used in RStudio; run <a href="#">getAceThemes</a> to get the list of available themes
fontSize	font size
tabSize	number of spaces for the indentation (usually 2 or 4); if NULL, it is set to the one used in RStudio
autoCompletion	logical, whether to enable the auto completion
snippets	logical, whether to enable the snippets (for example, there is a snippet for <code>switch</code> in JavaScript)
width, height	dimensions; the default values are nice for usage in the RStudio viewer pane
elementId	a HTML id for the container; this is useless for common usage

### Note

The diff editor does not correctly reacts when one resizes the RStudio viewer pane. You have to resize it before opening the diff editor.

**Examples**

```
file <- system.file("htmlwidgets", "aceEditor.css", package = "aceEditor")
aceDiffEditor(file, file)
```

---

aceEditor

*Ace editor*


---

**Description**

Open the Ace editor.

**Usage**

```
aceEditor(
  contents,
  mode = NULL,
  theme = NULL,
  fontSize = 14,
  tabSize = NULL,
  autoCompletion = TRUE,
  snippets = FALSE,
  width = NULL,
  height = NULL,
  elementId = NULL
)
```

**Arguments**

contents	this can be the path to a file, NULL to open an empty editor, or missing to open the file currently open in RStudio
mode	the language of the contents; if NULL and the contents are read from a file, the mode is guessed from the extension of the file; run <a href="#">getAceModes</a> to get the list of available modes
theme	the theme of the editor; if NULL, the theme is set to the theme currently used in RStudio; run <a href="#">getAceThemes</a> to get the list of available themes
fontSize	font size
tabSize	number of spaces for the indentation (usually 2 or 4); if NULL, it is set to the one used in RStudio
autoCompletion	logical, whether to enable the auto completion
snippets	logical, whether to enable the snippets (for example, there is a snippet for <code>switch</code> in JavaScript)
width, height	dimensions; the default values are nice for usage in the RStudio viewer pane
elementId	a HTML id for the container; this is useless for common usage

## Examples

```
# in RStudio, `aceEditor()` opens the current file:
aceEditor()

# opens a new JavaScript file:
aceEditor(NULL, mode = "javascript")

# opens an existing file:
aceEditor(system.file("htmlwidgets", "aceEditor.css", package = "aceEditor"))

# two editors side-by-side:
library(aceEditor)
library(htmltools)

ed1 <- aceEditor(
  width = "100%", height = "calc(100vh - 10px)"
)
ed2 <- aceEditor(
  width = "100%", height = "calc(100vh - 10px)"
)

if(interactive()){
  browsable(
    div(
      div(ed1, style="position: fixed; left: 1vw; right: 51vw;"),
      div(ed2, style="position: fixed; left: 51vw; right: 1vw;")
    )
  )
}

# two stacked editors:
library(aceEditor)
library(htmltools)

ed1 <- aceEditor(
  height = "calc(50vh - 10px)", width = "100%"
)
ed2 <- aceEditor(
  height = "calc(50vh - 10px)", width = "100%"
)

if(interactive()){
  browsable(
    tagList(
      tags$style(HTML(
        ".editor {",
        "  position: fixed;",
        "  left: 1vw;",
        "  width: 98vw;",
        "}"
      ))
    )
  )
}
```

```
   )),  
    div(  
      div(ed1, class = "editor", style = "bottom: calc(50vh - 25px);"),  
      div(ed2, class = "editor", style = "top: calc(50vh);")  
    )  
  )  
}  
}
```

---

getAceModes

*Ace modes*

---

### Description

Get the available modes of the Ace editor.

### Usage

```
getAceModes()
```

---

getAceThemes

*Ace themes*

---

### Description

Get available themes of the Ace editor.

### Usage

```
getAceThemes()
```

# Index

`aceDiffEditor`, [2](#)  
`aceEditor`, [3](#)

`getAceModes`, [2](#), [3](#), [5](#)  
`getAceThemes`, [2](#), [3](#), [5](#)