Sphinx Copybutton Documentation

Executable Books Project

Jul 03, 2021

CONTENTS

Sphinx-copybutton does one thing: add a little "copy" button to the right of your code blocks. That's it! It is a lightweight wrapper around the excellent (and also lightweight) Javascript library ClipboardJS.

Here's an example

And here's a code block, note the copy button to the right!

```
copy me!
```

If the code block overlaps to the right of the text area, you can just click the button to get the whole thing.

```
12345678912345678912345678912345678912345678912345678912345678912345678912345678912345678912345678912
```

You can configure `sphinx-copybutton` to detect *input prompts* in code cells, and then both remove these prompts before copying, as well as skip lines that *don't* start with prompts (in case they are output lines).

For example, this site has been configured to strip Python prompts (">>> "). Try copy-pasting the code block below.

```
>>> a = 2
>>> print(a)
2

>>> b = 'wow'
>>> print(b)
wow
```

CHAPTER

ONE

INSTALLATION

Note: `sphinx-copybutton` only works on Python >= 3.6

You can install `sphinx-copybutton` with `pip`:

```
pip install sphinx-copybutton
```

Or with `conda` via `conda-forge`:

```
conda install -c conda-forge sphinx-copybutton
```

Here's a link to the sphinx-copybutton GitHub repository.

CHAPTER

TWO

USAGE

In your `conf.py` configuration file, add `sphinx_copybutton` to your extensions list. E.g.:

```
extensions = [
    ...
    'sphinx_copybutton'
    ...
]
```

When you build your site, your code blocks should now have little copy buttons to their right. Clicking the button will copy the code inside!

CHAPTER

THREE

CUSTOMIZATION

Sphinx-copybutton was designed to work with the default Sphinx theme, Alabaster. If you use a theme that doesn't play nicely with sphinx-copybutton's CSS, you can always add your own CSS rules!

3.1 Customize the CSS

To customize the display of the copy button, you can add your own CSS files that overwrite the CSS in the sphinx-copybutton CSS rules. Just add these files to `_static` in your documentation folder, and it should overwrite sphinx-copybutton's behavior.

3.2 Strip and configure input prompts for code cells

By default, `sphinx-copybutton` will copy the entire contents of a code block when the button is clicked. For many languages, it is common to include **input prompts** with your examples, along with the outputs from running the code.

`sphinx-copybutton` provides functionality to both strip input prompts, as well as *only* select lines that begin with a prompt. This allows users to click the button and *only* copy the input text, excluding the prompts and outputs.

To define the prompt text that you'd like removed from copied text in your code blocks, use the following configuration value in your `conf.py` file:

```
copybutton_prompt_text = "myinputprompt"
```

When this variable is set, `sphinx-copybutton` will remove the prompt from the beginning of any lines that start with the text you specify. In addition, *only* the lines that contain prompts will be copied if any are discovered. If no lines with prompts are found, then the full contents of the cell will be copied.

For example, to exclude traditional Python prompts from your copied code, use the following configuration:

```
copybutton_prompt_text = ">>> "
```

3.2.1 Using regexp prompt identifiers

If your prompts are more complex than a single string, then you can use a regexp to match with.

Note: Keep in mind that the RegExp you are writing is evaluated in JavaScript and not in Python. In some edge cases this might lead to different results.

If you enclose your regexp in a raw string (`r""`), you can easily test that your RegExp matches all the wanted prompts, i.e. at RegEx101.

For example this documentation uses the following configuration:

```
copybutton_prompt_text = r">>> |\.\.\. |\$ |In \[\d*\]: | {2,5}\.\.\.: | {5,8}: "
copybutton_prompt_is_regexp = True
```

Which matches the following prompts and their continuations if they exist:

Prompt Name	RegEx Pattern	Matched String Examples
Python Repl + continuation	`r'>>> |\.\.\. '`	`'>>> ','... '`
Bash	`r'\$ '`	`'$ '`
`ipython` and `qtconsole` + continuation	`r'In \[\d*\]: | {2,5}\.\.\.: '`	`'In []: ', 'In [999]: ',' ...: ',' ...: '`
`jupyter-console` + continuation	`r'In \[\d*\]: | {5,8}: '`	`'In []: ', 'In [999]: ',' ...: ',' ...: '`

An example usage would be the `ipython`-directive:

```
``ipython`` and ``qtconsole`` style:

.. code-block:: ipython

   In [1]: first
      ...: continuation
   output
   In [2]: second

``jupyter`` style:

.. code-block:: ipython

   In [1]: first
         : continuation
   output
   In [2]: second
```

`ipython` and `qtconsole` style:

```
In [1]: first
   ...: continuation
output
In [2]: second
```

`jupyter` style:

```
In [1]: first
      : continuation
output
In [2]: second
```

If you want a detailed explanation how the RegEx's work you can also use RegEx101 and read the `Explanation` sidebar.

3.2.2 Configure whether *only* lines with prompts are copied

By default, if sphinx-copybutton detects lines that begin with code prompts, it will *only* copy the text in those lines (after stripping the prompts). This assumes that the rest of the code block contains outputs that shouldn't be copied.

To disable this behavior, use the following configuration in `conf.py`:

```
copybutton_only_copy_prompt_lines = False
```

In this case, all lines of the code blocks will be copied after the prompts are stripped.

3.2.3 Configure whether the input prompts should be stripped

By default, sphinx-copybutton will remove the prompt text from lines according to the value of `copybutton_prompt_text`.

To disable this behavior and copy the full text of lines with prompts (for example, if you'd like to copy *only* the lines with prompts, but not strip the prompts), use the following configuration in `conf.py`:

```
copybutton_remove_prompts = False
```

3.2.4 Configure whether *empty* lines are copied

By default, sphinx-copybutton will also copy / pass through empty lines, determined by `line.trim() === ''`.

To disable copying empty lines, use the following configuration in `conf.py`:

```
copybutton_copy_empty_lines = False
```

3.2.5 Honor line continuation characters when copying multline-snippets

Sometimes you may wish to copy a code block like this one:

```
$ datalad download-url http://www.tldp.org/LDP/Bash-Beginners-Guide/Bash-Beginners-
↪Guide.pdf \
--dataset . \
-m "add beginners guide on bash" \
-O books/bash_guide.pdf
```

Assuming that you specified `$` as your prompt, sphinx-copybutton will only copy the first line by default.

To copy all lines above, you can use the following configuration:

```
copybutton_line_continuation_character = "\\"
```

Note that if we want to define \ as the line continuation character, we have to "escape" it with another \, as with any Python string that should carry a literal \.

Next, this configuration will make the code look for lines to copy based on the rules above, but if one of the lines to be copied contains a line continuation character, then the next line will be automatically copied, regardless of whether it matches the other rules.

3.2.6 Honor HERE-document syntax when copying multline-snippets

HERE-documents are a form of multiline string literals in which line breaks and other whitespace (including indentation) is preserved. For example:

```
$ cat << EOT > notes.txt
   This is an example sentence.
       Put some indentation on this line.

  EOT
```

Executing this codeblock in the terminal will create a file `notes.txt` with the exact text from line two of the codeblock until (but not including) the final line containing `EOT`.

However, assuming that you specified `$` as your prompt, sphinx-copybutton will only copy the first line by default.

sphinx-copybutton can be configured to copy the whole "HERE-document" by using the following configuration:

```
copybutton_here_doc_delimiter = "EOT"
```

This will continue to look for lines to copy based on the rules above, but if one of the lines to be copied contains the defined delimiter (here: `EOT`), then all following lines will be copied literally until the next delimiter is encountered in a line.

3.2.7 Use a different copy button image

To use a different image for your copy buttons, do the following:

1. Place the image in the `_static/` folder of your site.
2. Set the `copybutton_image_path` variable in your `conf.py` to be the path to your image file, **relative to** `_static/`.

3.2.8 Configure the CSS selector used to add copy buttons

By default, `sphinx-copybutton` will add a copy button to all elements that match the following selection:

```
div.highlight pre
```

To change this selector, use the following configuration in `conf.py`:

```
copybutton_selector = "your.selector"
```

In this case, all elements that match `your.selector` will have a copy button added to them.

CHAPTER

FOUR

DEVELOPMENT

If you'd like to develop or make contributions for sphinx-copybutton, fork the repository here:

https://github.com/ExecutableBookProject/sphinx-copybutton

pull to your computer and install locally with `pip`:

```
pip install -e /path/to/sphinx_copybutton
```

Pull requests and **Issues** are absolutely welcome!

The package is tested for three things (see `.github/workflows/integration.yml`):

4.1 code style

To adhere to this code style install the package with pre-commit:

```
$ pip install .[code_style]
```

Then you can run:

```
$ pre-commit run --all
```

Or setup pre-commit to run on code commits:

```
$ pre-commit install
```

4.2 JavaScript unit testing

Install the test dependencies with npm:

```
$ npm install ci
```

Then run the tests:

```
$ npm test
```

Note: NodeJS >= 12 is required

4.3 Documentation builds

Install the package:

```
$ pip install .
```

Then run the docs build:

```
$ cd doc
$ make html
```

4.3.1 A nested page for reference

To make sure that the images / svg still works!

```
e = mc^2
```