pandoc-emphasize-code

Contents

Usage

Often when working with code examples in documentation, printed or web hosted, or in presentation slideshows, you might want to emphasize parts of a code snippet.

You can get away with manually writing the target markup, in LaTeX or raw HTML, but if you want to render the same document in multiple output formats, this gets really tedious. Also, having to write the markup by hand can be error prone.

This filter lets you specify *ranges* of a code block to emphasize, and have the filter generate the appropriate markup for you. It recognizes code blocks with the `emphasize` attribute present:

````
```{.haskell emphasize=2:3-2:14,3:3-3:12}
myFunc = do
  newStuffHere
  andThisToo notThis
  notSoRelevant
```
````

Currently, the following output formats are supported:

- HTML (`html` and `html5`)
- LaTeX (`latex` and `beamer`)
- GitHub-Flavored Markdown (`markdown_github`)

Syntax

The value of the `emphasize` attribute is a comma-separated list of *ranges*. A *range* consists of two positions, separated by a dash. A *position* consists of a *line number* and a *column number*, separated by a colon.

The syntax can be described in EBNF, like so:

```
line number     = natural number;
column number   = natural number;
position        = line number, ":", column number;
range           = position, "-", position;
ranges          = range, { ",", range) };

(* definition of natural number excluded for brevity *)
```

Rendering to HTML

The code block above would render the following HTML output:

```
<pre class="haskell"><code>myFunc = do
  <em>newStuffHere</em>
  <em>andThisToo</em> notThis
  notSoRelevant</code></pre>
```

When rendering to HTML, the markup can be styled using CSS:

```
code em {
  color: red;
  font-style: italic;
}
```

By default, if no custom styling is applied, it will look something like this:

```
myFunc = do
  newStuffHere
  andThisToo notThis
  notSoRelevant
```

Note that the there is no additional syntax highlighting when emphasizing code and rendering to HTML, as there is no way to use Pandoc's highlighter and

embed custom HTML tags. You might be able to add that using a Javascript highlighter running on the client.

Rendering with LaTeX

When rendering using LaTeX, two things are required:

- The `listings` package needs to be included.
- You need to define a `CodeEmphasis` command, styling the emphasized code in `lstlisting`s.

If you're not using a custom LaTeX template, you can use the YAML front matter in a Markdown source file to add the requirements:

```
header-includes:
  - \usepackage{listings}
  - \lstset{basicstyle=\ttfamily}
  - \newcommand{\CodeEmphasis}[1]{\textcolor{red}{\textit{#1}}}
```

NOTE: When rendering as Beamer slides, any frame including an emphasized block must be marked as `fragile`:

```
## My Slide {.fragile}

```{.haskell emphasize=2:3-2:14,3:3-3:12}
myFunc = do
  newStuffHere
  andThisToo notThis
  notSoRelevant
```
```

Regular Highlighting

You can still use regular Pandoc highlighting (the *skylighting* library):

```
``` {.css}
.hello {
  world: yes;
}
```
```

It gives you all the nice colors:

```css
.hello {
  world: yes;
}
```

The drawback is that you have two different highlighting systems now, one for emphasized code, one for regular code blocks.

Install

Executables for Linux and macOS are available in the Releases page.

From Hackage

If you'd rather install using `cabal` or `stack`, you can use the following command:

```
cabal install pandoc-emphasize-code
```

The package is available at Hackage.

Build

Requirements:

- Cabal or Stack, either works.

To install from sources, run:

```
git clone git@github.com:owickstrom/pandoc-emphasize-code.git
cd pandoc-emphasize-code
stack setup
stack install
```

Run

If you have installed from sources, and you have `~/.local/bin` on your `PATH`, you can use the filter with Pandoc like so:

```
pandoc --filter pandoc-emphasize-code input.md output.html
```

Changelog

- **0.1.0**
  - First release
  - Support for multiple ranges
  - Rendering support for HTML, Markdown, and LaTeX

License

Mozilla Public License Version 2.0