Slides in Literate Haskell with Pandoc

Dom De Re

7 July, 2014

Slides in Literate Haskell

Pandoc

You will need Pandoc

Pandoc gets the Literate Haskell source file (or Markdown file) and marks it up into slides.

Installing **Pandoc**

- On **Ubuntu**, you can install it with the package manager: `sudo apt-get install pandoc`
- However if you are rendering a Literate Haskell file, you are likely to be familiar with `cabal` and `ghc`, in which case you can install it like you would any other Haskell binary off Hackage.

LaTeX and Beamer

Installing LaTeX and Beamer

If you wish to build the `beamer` target, you will need to install the `latex-beamer` package with:

```
sudo apt-get install latex-beamer
```

Write Slides

You can write your slides in **Github flavoured Markdown** mixed with or without **Literate Haskell**

Github Flavoured Markdown Example

A Python excerpt:

```
def erp(r, x):
    print("erpderp")
```

Literate Haskell 1 (Bird Style)

Ordinarily in Markdown, a $>$ character at the beginning of a line would signify a quote block.

When using Literate Haskell however, a $>$ at the beginning of a line will signify a line of **Haskell** code that you want both marked up in the slides and compiled into the module

```
foldr' :: (a -> b -> b) -> b -> [a] -> b
foldr' _ y []     = y
foldr' f y (x:xs) = f x (foldr' f y xs)
```

Literate Haskell 2

If you want to write broken code without it stopping the module from compiling, there are two ways to do it.

You can start the line with a $<$ char instead of a $>$ char:

```
-- | This wouldn't compile
foo :: a -> b
foo x = x
```

Or you can embed `Haskell` code the way you already would in Github Flavoured Markdown:

```
-- | This would not compile either
foo2 :: (s -> a) -> (s -> b -> t) -> (a -> b) -> s -> t
foo2 g s w x = g s
```

Building The Slides

`pandoc` can output a wide variety of formats, right now, the included `Makefile` only builds a small subset of them

Reveal.js

Build the slides with:

```
make revealjs
```

By default it uses the `sky` theme and sets the `slide-level` to 2

Output goes to `revealjs/index.html`

Beamer

Build the slides with:

```
make beamer
```

It uses the default theme and also sets the `slide-level` to 2

Output goes to `beamer/talk.pdf`