

WHAT I LEARNED BY LEARNING HASKELL

-  Christoph Welcz
-  18.04.2018
-  @ChristophWelcz
-  github.com/enolive/haskell-intro



WHY HASKELL?



- force myself to use FP
- leave my comfort zone
- understand libraries like Ramda or Vavr.

A dramatic close-up of a man's face. He has short, dark hair and a beard. His eyes are wide open, looking upwards and slightly to the right with a shocked or intense expression. A bright, overexposed light source is visible behind his head, creating a lens flare effect. In the background, other people are visible but out of focus.

Welcome to FP Club!

Rule #1: You do not talk about the m-word

Rule #2: You DO NOT talk about the m-word



- Pure functional language
- Statically typed
- First version 1990
- For mathematicians

Tool	Description
GHC	Glasgow Haskell Compiler
GHCi	GHC interactive, aka REPL
Stack	project templates, dependency management, build
Hoogle	documentation database
HSpec	write and run tests with a BDD syntax similar to RSpec
HLint	the linter, d'oh!

First Class Functions Infix Syntax
Recursion Function Composition Guards
Pattern Matching List
Comprehensions Lazy/Strict
Evaluation Partial
Application (Un-)Currying Eta
Conversion (Un-)Folding

Haskell Stack Commands

```
# creates a new project using the default template
stack new my-project name
# starts the REPL
stack ghci
# starts the local hoogle documentation server
stack hoogle server
# runs all tests
stack test
```

BASH

IMPORTANT GHCI COMMANDS

Command	Meaning
:r	Reloads all modules
:t TYPE	Shows type information
:m MODULE	loads the given module (public-only)
:m *MODULE	loads the given module (all)
:I FILE	loads the given file into the module

HELLO WORLD

```
module HelloWorld
  ( greet
  ) where

greet :: String -> String
greet name = "Hello, " ++ name ++ "!"
```

Haskell

FACTORIAL

```
fac :: Int -> Int  
fac 1 = 1  
fac n = n * fac (n - 1)
```

Haskell

FIBONACCI SEQUENCE

```
-- see also
-- https://stackoverflow.com/questions/1105765/generating-fibonacci-numbers
fibs :: [Int]
fibs = 1 : 1 : zipWith (+) fibs (tail fibs)

fib :: Int -> Int
fib n = fibs !! n
```

Haskell

USEFUL STUFF

- Haskell Stack
- IntelliJ-Haskell
- Hoogle
- Learn You a Haskell for Great Good!
- Exercism Haskell Track