

# notes

Curtis Carter

September 24, 2014

## Contents

<b>1</b>	<b>Purpose</b>	<b>2</b>
<b>2</b>	<b>Pre-reqs</b>	<b>2</b>
<b>3</b>	<b>Setup (What I did before the first commit)</b>	<b>2</b>
<b>4</b>	<b>Scotty</b>	<b>3</b>
<b>5</b>	<b>Next</b>	<b>3</b>

## 1 Purpose

The purpose of this repository is for users somewhat new to Haskell to have an idea of how to go about creating a basic HTTP service with the following features:

- Basic HTTP concepts (routing, serving requests)
- Configuration
- Database access
- Logging
- Monitoring
- Structuring a Haskell application

## 2 Pre-reqs

- Recent version of GHC. At the time of this project 7.8.2
- Recent version of Cabal. At the time of this project 1.20.0.2

## 3 Setup (What I did before the first commit)

- Cabal update
- Make the directory where our project will live.
  - make sub-directory called src in this dir.
- Make a file called Main.hs with the main function in it inside the src directory. This is only temporary to get our project compiling/running.

```
module Main where

main :: IO ()
main = putStrLn "wee"
```

- cd into the project directory
- run: cabal init

- Take defaults except for the Library | Executable question. Pick Executable.
- Modify your .cabal file to include the following in the Executable section
  - \* hs-source-dirs: src
- CRITICAL!: run (cabal sandbox init) This creates a sandbox.
- Your application should now build (cabal build) and run (cabal run)

## 4 Scotty

- Add scotty as a dependency in your .cabal
  - build-depends: base >=4.7 && <4.8 , scotty
- Install scotty and it's dependencies (cabal install --only-dependencies)

## 5 Next

If you want to see the path I took to get to the current state, I tried to keep the commits feature oriented, but there are some refactoring commits you may want to look at as well.