

Getting Started with Haskell

Tim Sheard

Installing GHC

- We will use Haskell Platform 2012.4
- Goto haskell.org
- Click on the Haskell platform

Haskell - HaskellWiki - Mozilla Firefox

File Edit View History Bookmarks Tools Help

overstrike in html - Goog... color text in html - Goog... Tim Sheard's home page Haskell - HaskellWiki

www.haskell.org/haskellwiki/Haskell

Log in / create account

The Haskell Programming Language

Haskell is an advanced purely-functional programming language. An open-source product of more than twenty years of cutting-edge research, it allows rapid development of robust, concise, correct software. With strong support for integration with other languages, built-in concurrency and parallelism, debuggers, profilers, rich libraries and an active community, Haskell makes it easier to produce flexible, maintainable, high-quality software.

Learn Haskell

- [What is Haskell?](#)
- [Try Haskell in your browser](#)
- [Learning resources](#)
- [Books & tutorials](#)
- [Library documentation](#)

Use Haskell

- [Download Haskell](#)
- [Language specification](#)
- [Hackage library database](#)
- [Applications and libraries](#)
- [Hoogle and Hayoo API search](#)

Join the Community

- [Haskell on Reddit, Stack Overflow](#)
- [Mailing lists, IRC channels](#)
- [Wiki \(how to contribute\)](#)
- [Communities and Activities Reports](#)
- [Haskell in industry, research and education.](#)
- [Planet Haskell](#), [The Monad Reader](#)

Headlines

- 2012:
 - [The Haskell Platform 2012.4 is now available](#)
 - [GHC 7.6 is released](#)
 - [The Haskell Platform 2012.2 is now available](#)
 - [Yesod 1.0 is now available](#)
 - [GHC 7.4 is released](#)
 - [O'Reilly have announced a forthcoming book on Parallel and Concurrent Haskell](#)

Upcoming Events

Recent Events

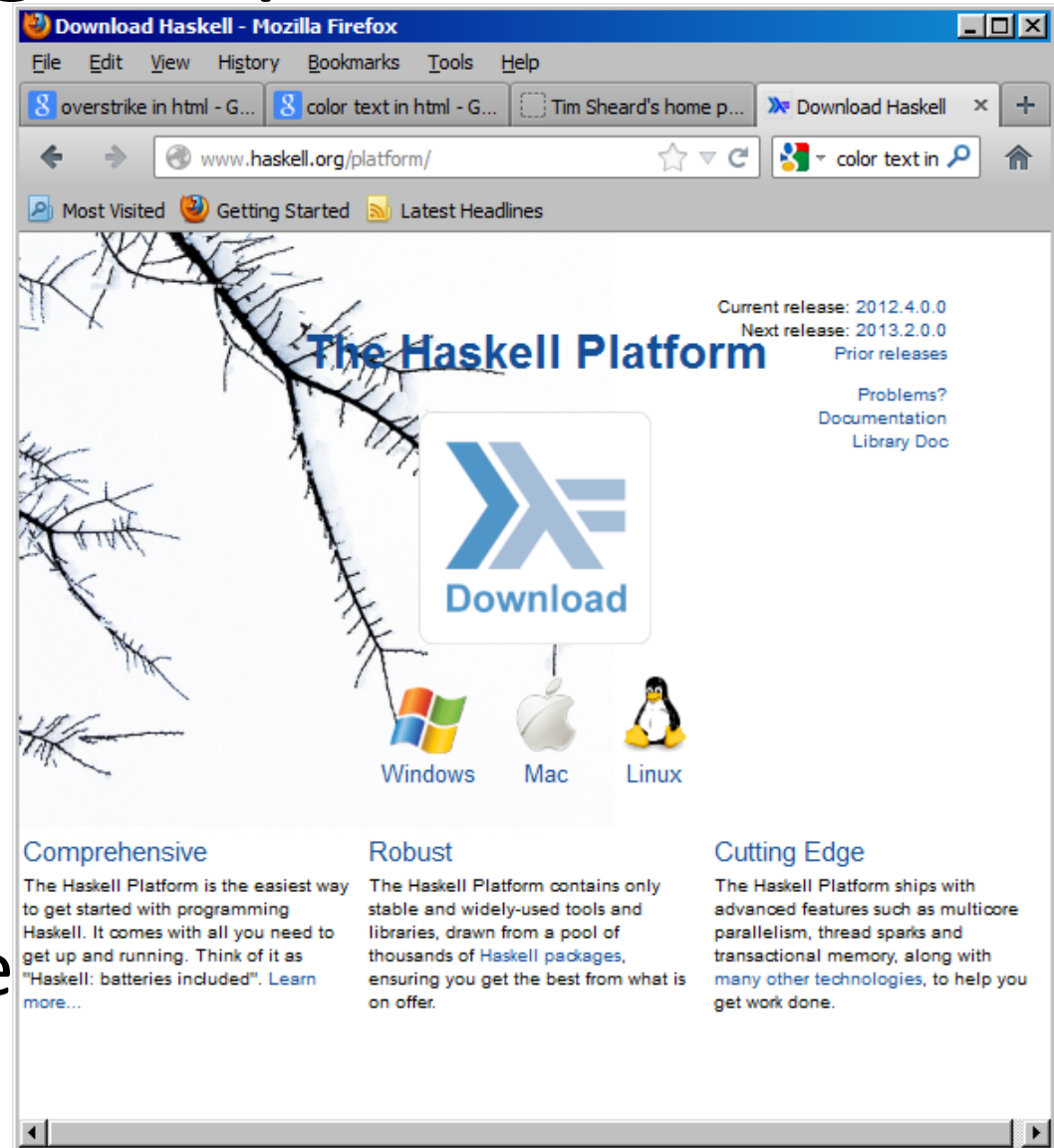
- [ACM SIGPLAN Haskell Symposium 2012](#)
September 13, 2012, Copenhagen, Denmark
- [CUFP 2012, with tutorials including two-days of Haskell introductory training](#)
September 13-14, 2012, Copenhagen, Denmark
- [24th Symposium on Implementation and Application of Functional Languages.](#)
August 30 - September 1, 2012, Oxford, UK
- [4th Summerschool on Applied Functional Programming](#)
August 20-31, 2012, Utrecht, The Netherlands
- [ICFP Programming Contest 2012](#)

Recent Package Updates

[See here](#)

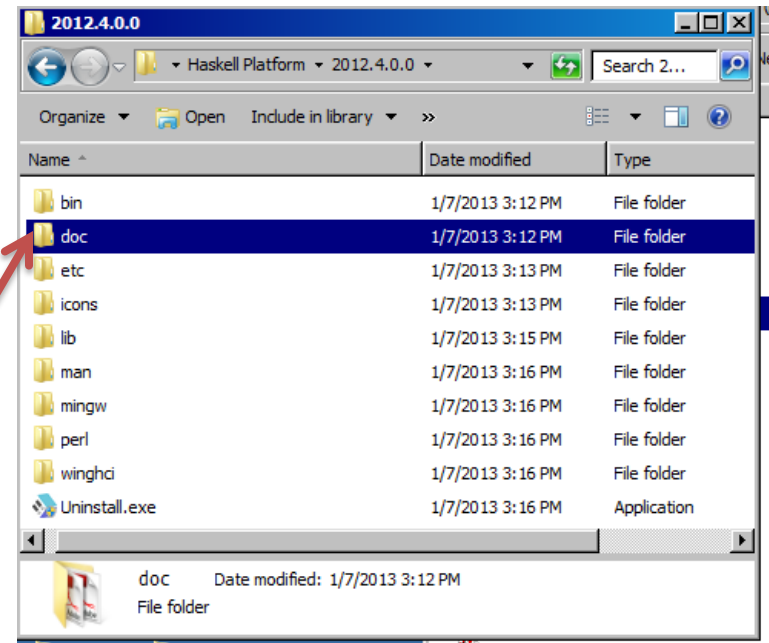
Installing the platform

- Choose your operating system
 - Windows
 - Mac
 - Linux
 - On Debian
 - `apt-get install ghc6`
- Follow directions
- Ask me if you have problems

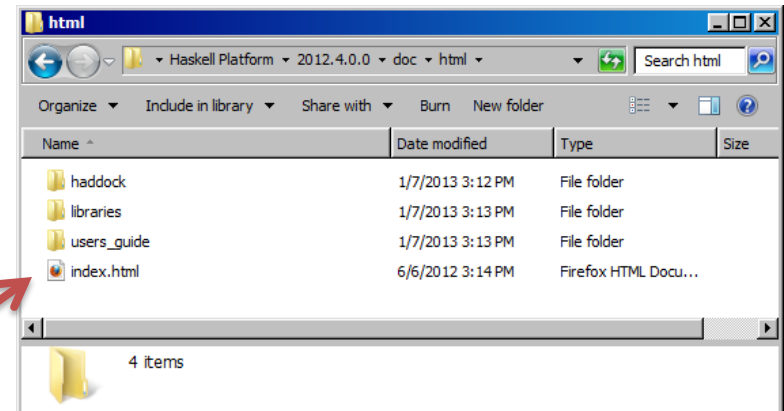


Getting Help

- On Windows, in the directory where GHC is installed, there is a directory called “doc”



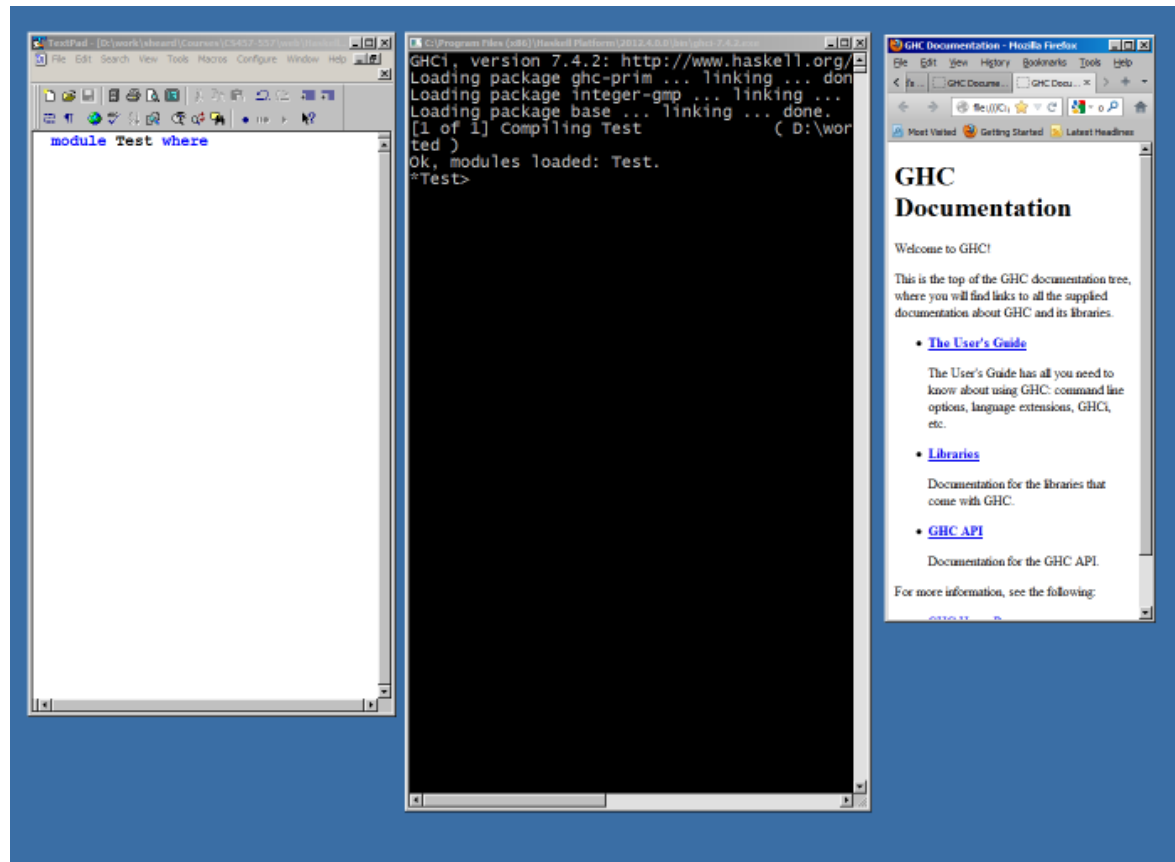
- In this directory is another directory called html, and here there is a file called “index.html”



Advice

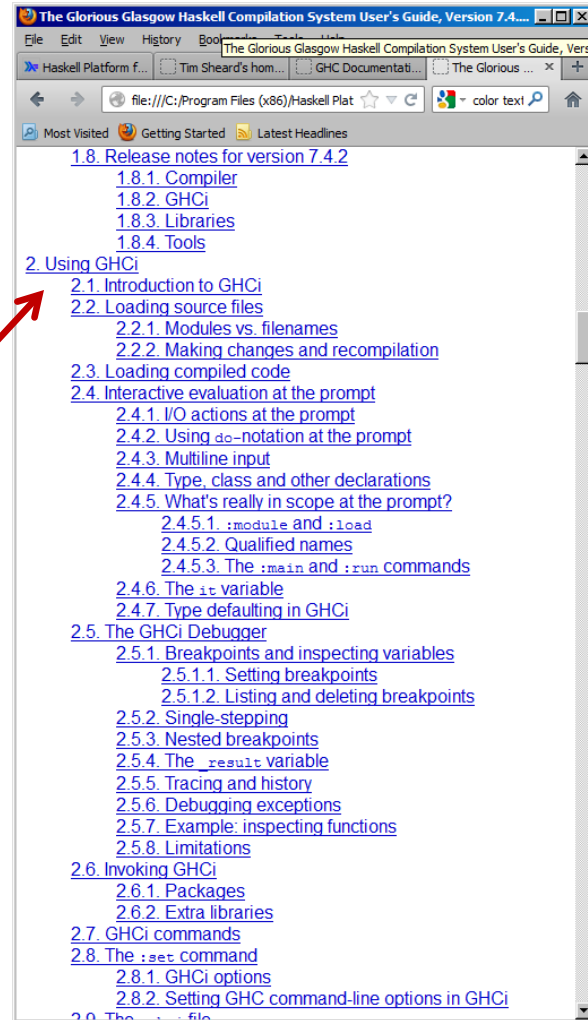
- Arrange your screen with side by side windows

- Editor
- GHCi
- Help



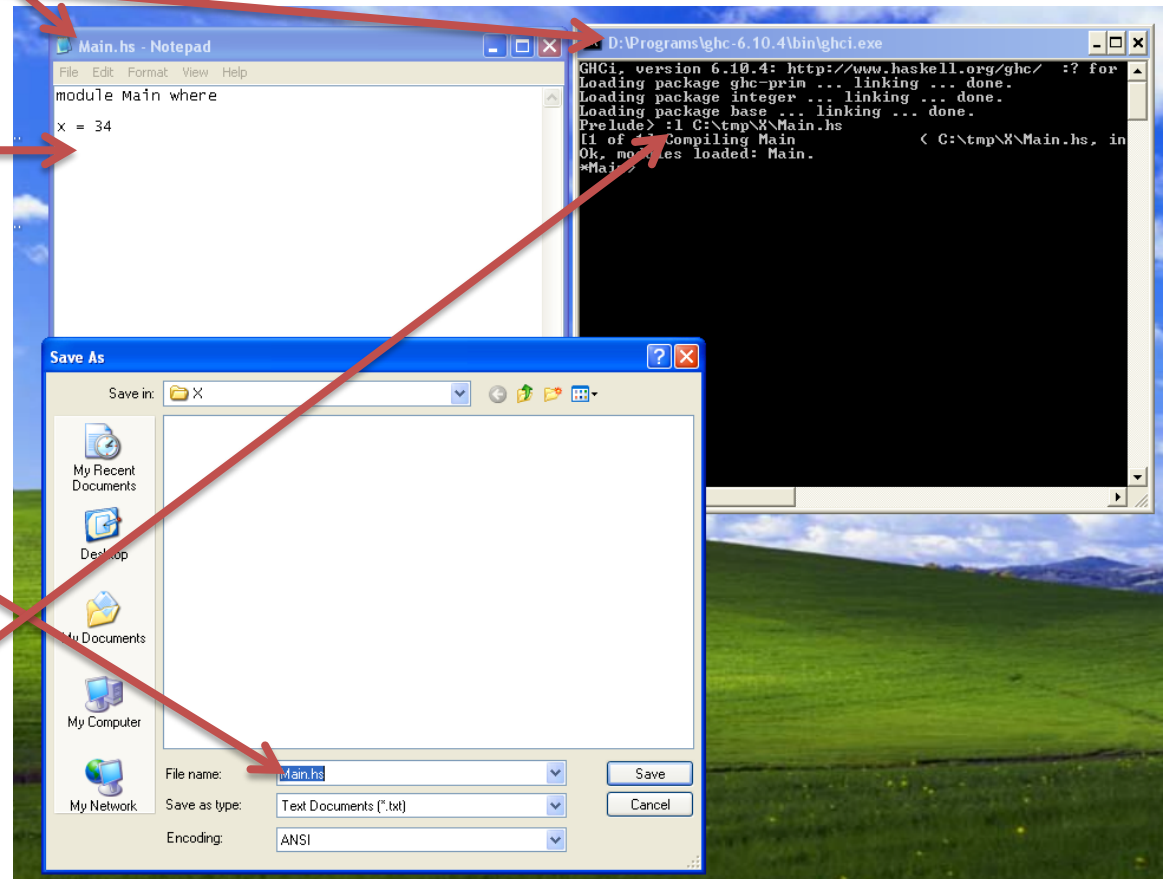
Learning what's possible

- From the index page click on “The Users Guide”
- Scroll down to “Using GHCi”
- Explore



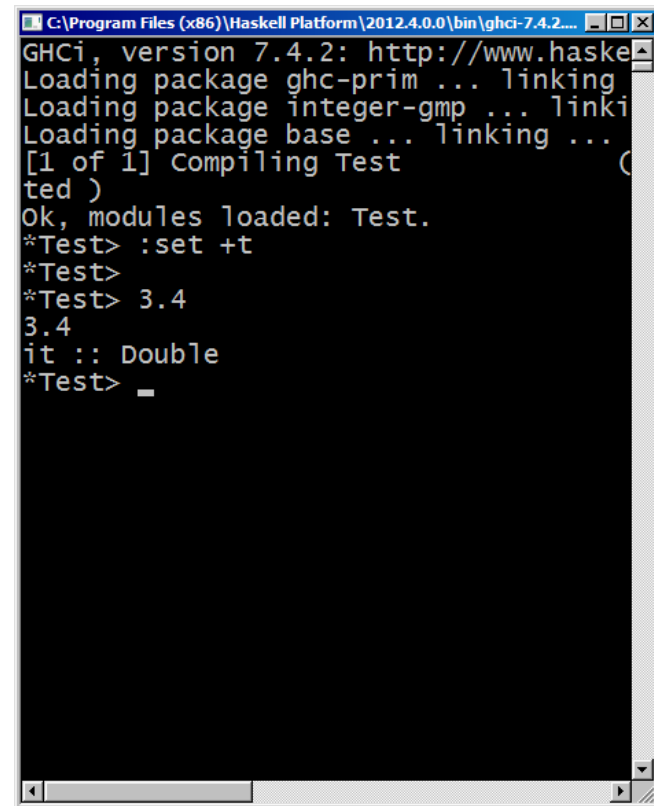
Writing a program

- Start up an editor
- Start up ghci
- Write a program in the editor,
- save the file
- “open” the file in ghci by using
`:l C:\tmp\X\Main.hs`



Setting Options

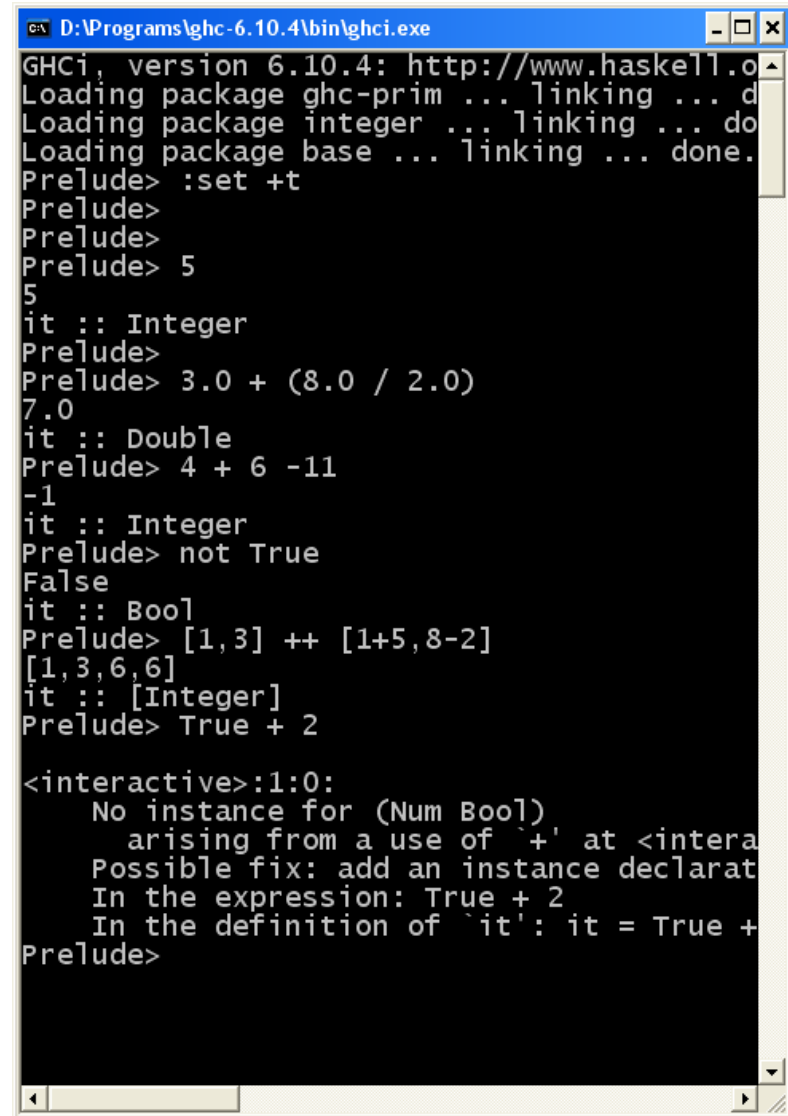
- GHCi allows the user to set many options.
- I recommend use of the “show type option” especially for new users
 - `:set +t`



```
C:\Program Files (x86)\Haskell Platform\2012.4.0.0\bin\ghci-7.4.2...
GHCi, version 7.4.2: http://www.haskell.org/ghci/
Loading package ghc-prim ... linking ... done
Loading package integer-gmp ... linking ... done
Loading package base ... linking ... done
[1 of 1] Compiling Test
Test
Ok, modules loaded: Test.
*Test> :set +t
*Test>
*Test> 3.4
3.4
it :: Double
*Test> _
```


Interaction in Ghci

- Type an expression
- GHCi checks that it is well formed
 - Well-typed
- Evaluates the expression
 - Carries out the computation
- Prints the result
- And the Type



```
D:\Programs\ghc-6.10.4\bin\ghci.exe
GHCi, version 6.10.4: http://www.haskell.org/ghc/
Loading package ghc-prim ... linking ... done.
Loading package integer ... linking ... done.
Loading package base ... linking ... done.
Prelude> :set +t
Prelude>
Prelude>
Prelude> 5
5
it :: Integer
Prelude>
Prelude> 3.0 + (8.0 / 2.0)
7.0
it :: Double
Prelude> 4 + 6 -11
-1
it :: Integer
Prelude> not True
False
it :: Bool
Prelude> [1,3] ++ [1+5,8-2]
[1,3,6,6]
it :: [Integer]
Prelude> True + 2
<interactive>:1:0:
  No instance for (Num Bool)
    arising from a use of `+' at <interactive>:1:0:
  Possible fix: add an instance declaration
  In the expression: True + 2
  In the definition of `it': it = True + 2
Prelude>
```