



Learn You a Haskell for Great Good!

1. Introduction

[About this tutorial](#)

[So what's Haskell?](#)

[What you need to dive in](#)

2. Starting Out

[Ready, set, go!](#)

[Baby's first functions](#)

An intro to lists

Texas ranges

I'm a list comprehension

Tuples

3. Types and Typeclasses

Believe the type

Type variables

Typeclasses 101

4. Syntax in Functions

Pattern matching

Guards, guards!

Where!?

Let it be

Case expressions

5. Recursion

Hello recursion!

Maximum awesome

A few more recursive functions

Quick, sort!

Thinking recursively

6. Higher Order Functions

Curried functions

Some higher-orderism is in order

Maps and filters

Lambdas

Only folds and horses

Function application with \$

Function composition

7. Modules

Loading modules

Data.List

Data.Char

Data.Map

Data.Set

Making our own modules

8. Making Our Own Types and Typeclasses

Algebraic data types intro

Record syntax

Type parameters

Derived instances

Type synonyms

Recursive data structures

Typeclasses 102

A yes-no typeclass

The Functor typeclass

Kinds and some type-foo

9. Input and Output

Hello, world!

Files and streams

Command line arguments

Randomness

Bytestrings

Exceptions

0. Functionally Solving Problems

Reverse Polish notation calculator

Heathrow to London

1. Functors, Applicative Functors and Monoids

Functors redux

Applicative functors

The newtype keyword

Monoids

2. A Fistful of Monads

Getting our feet wet with Maybe

The Monad type class

Walk the line

do notation

The list monad

Monad laws

3. For a Few Monads More

Writer? I hardly know her!

Reader? Ugh, not this joke again.

Tasteful stateful computations

Error error on the wall

Some useful monadic functions

Making monads

4. Zippers

Taking a walk

A trail of breadcrumbs

Focusing on lists

A very simple file system

Watch your step

This work is licensed under a [Creative Commons Attribution-Noncommercial-Share Alike 3.0 Unported License](https://creativecommons.org/licenses/by-nc-sa/3.0/) because I couldn't find a license with an even longer name.

