## Week 1

square :: Int -> Int

square x = x \* x

`mod` `div`

import Data.Char

toUpper toLower length

tail （除了第一个元素之外的列表）

\*收藏了一个看上去不错的列表操作网站

## Week 2

fst返回元组的第一个元素 fst(8,11),返回8

snd返回元组的最后一个元素

Head tail

Init last

注意括号使用，head list为一个输入值时打括号

注意type一致（可加空列表空str）

import Data.List

insert 某一元素插进列表中第一个比它大的前面（可核实）

drop x list列表去掉前x位

## Week 3

max取两数中大的，min

注意对function中对空列表的定义

## Week 4

Type 套 type

data Rhtype = Pos | Neg deriving (Show, Eq)

data ABOtype = A | B | AB | O deriving (Show, Eq)

data Bloodtype = Type ABOtype Rhtype deriving (Show, Eq)

data RhType = Pos | Neg

data ABOType = A | B | AB | O

data BloodType = BloodType ABOType RhType

patient1BT :: BloodType

patient1BT = BloodType A Pos

## Week 5

fac1 :: IO()

fac1 = do

putStr "Enter a number which is greater or equal than 0:"

num <- getLine

putStr "The factorial of this number is "

putStrLn (show (fac (read num)))

read只能读字符串

Haskell不更新变量的值