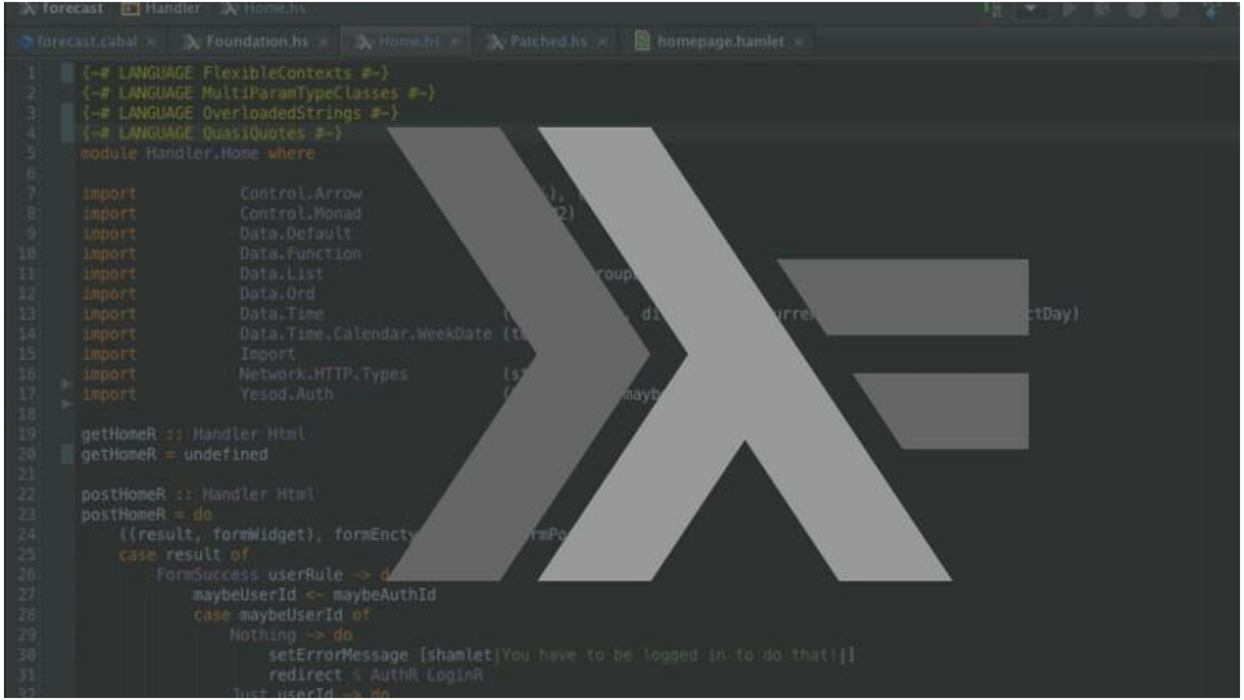
Higher Order Functions Problems



```
{-# LANGUAGE FlexibleContexts #-}
{-# LANGUAGE MultiParamTypeClasses #-}
{-# LANGUAGE OverloadedStrings #-}
{-# LANGUAGE QuasiQuotes #-}
module Handler.Home where

import           Control.Arrow
import           Control.Monad
import           Data.Default
import           Data.Function
import           Data.List
import           Data.Ord
import           Data.Time
import           Data.Time.Calendar.WeekDate (t
import           Import
import           Network.HTTP.Types          (s
import           Yesod.Auth

getHomeR :: Handler Html
getHomeR = undefined

postHomeR :: Handler Html
postHomeR = do
    ((result, formWidget), formEncty
    case result of
        FormSuccess userRule -> d
            maybeUserId <- maybeAuthId
            case maybeUserId of
                Nothing -> do
                    setErrorMessage [shamlet|You have to be logged in to do that!|]
                    redirect $ AuthR LoginR
                Just userId -> do
```

Lucas Bazilio - Udemy

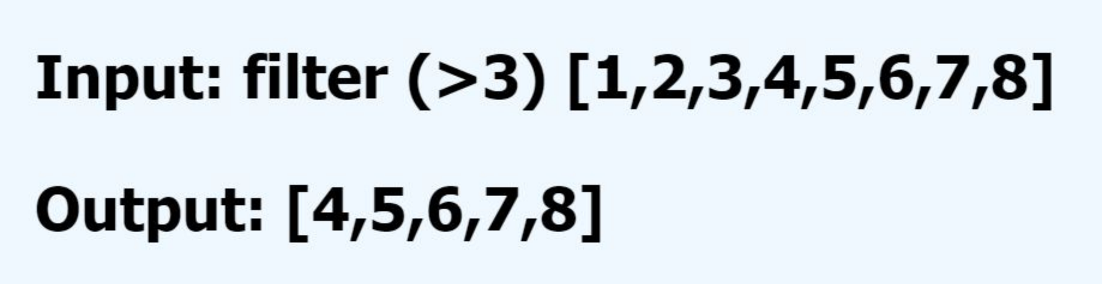
Problem 11

Define a function *countIf :: (Int -> Bool) -> [Int] -> Int* that, given a predicate on integers and a list of integers, returns the number of elements in the list that satisfy the predicate

Input	Output
`countIf (>5) [1..10]`	`-> 5`

Lucas Bazilio - Udemy

filter



Input: filter (>3) [1,2,3,4,5,6,7,8]

Output: [4,5,6,7,8]

Instructor Youtube Channel: Lucas Science