99 questions/Solutions/2

From HaskellWiki

< 99 questions | Solutions

(*) Find the last but one element of a list.

```
myButLast :: [a] -> a
myButLast = last . init

myButLast' x = reverse x !! 1

myButLast'' [x,_]  = x
myButLast'' (_:xs) = myButLast'' xs

myButLast''' (x:(_:[])) = x
myButLast''' (_:xs) = myButLast''' xs

myButLast'''' = head . tail . reverse

lastbut1 :: Foldable f => f a -> a
lastbut1 = fst . foldl (\(a,b) x -> (b,x)) (err1,err2)
  where
    err1 = error "lastbut1: Empty list"
    err2 = error "lastbut1: Singleton"

lastbut1safe :: Foldable f => f a -> Maybe a
lastbut1safe = fst . foldl (\(a,b) x -> (b,Just x)) (Nothing,Nothing)

myButLast''''' [] = error "Empty list"
myButLast''''' [x] = error "Too few elements"
myButLast''''' (x:xs) =
               if length xs == 1 then x
               else myButLast''''' xs
```

Retrieved from "https://wiki.haskell.org/index.php?title=99_questions/Solutions/2&oldid=60352"
Category:

- Programming exercise spoilers

- This page was last modified on 15 November 2015, at 20:35.
- Recent content is available under a simple permissive license.