

λ List Functions λ

Verwenden Sie ghci und <http://www.haskell.org/hoogle> um diese Liste zu vervollständigen.

Signatur	<code>(++) :: [a] -> [a] -> [a]</code>
Beispiel	<code>[1,2] ++ [3,4,5] ~> [1,2,3,4,5]</code> <code>"Hallo " ++ show 12 ~> "Hallo 12"</code>
Beschreibung	Hängt zwei Listen aneinander. Wird entsprechend verwendet um Strings zu konkatenieren.

Signatur	
Beispiel	<code>take 3 ['a','b','c','d','e'] ~> ['a','b','c']</code>
Beschreibung	

Signatur	
Beispiel	
Beschreibung	Wirft die ersten n Elemente weg.

Signatur	<code>(!!) :: [a] -> Int -> a</code>
Beispiel	
Beschreibung	

Signatur	
Beispiel	
Beschreibung	Gibt das letzte Element zurück.

Signatur	
Beispiel	<code>init ['a','b','c','d'] ~> ['a','b','c']</code>

Beschreibung	
---------------------	--

Signatur	
Beispiel	
Beschreibung	Dreht eine Liste um.

Signatur	<code>elem :: Eq a => a -> [a] -> Bool</code>
Beispiel	
Beschreibung	

Signatur	
Beispiel	<code>maximum [1,4,3] ~> 4</code> <code>minimum [1,4,3] ~> 1</code>
Beschreibung	

Signatur	
Beispiel	
Beschreibung	Gibt die Summe/ das Produkt zurück. Die Listenelemente müssen von einem Zahlen Typen sein (Num).

Signatur	<code>zip :: [a] -> [b] -> [(a,b)]</code>
Beispiel	
Beschreibung	

Signatur	
Beispiel	<code>concat [[1],[2,3],[4]] ~> [1,2,3,4]</code> <code>concat ["abc","def"] ~> "abcdef"</code>

Beschreibung	
---------------------	--

Signatur	
Beispiel	zipWith (+) [1,2,3] [10,11,12] ~> [11,13,15] zipWith (++) ["Ha","Ec"] ["llo","ho"] ~> ["Hallo","Echo"]
Beschreibung	