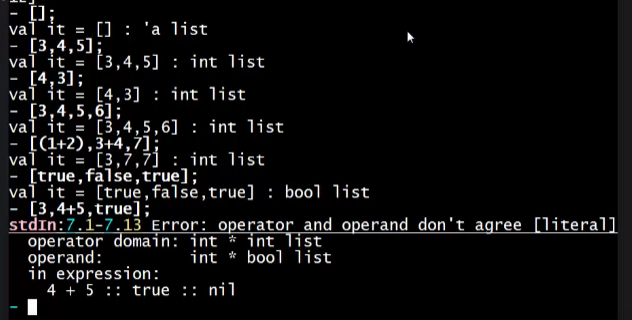
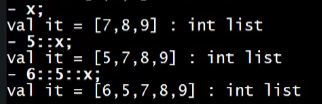


List Examples:

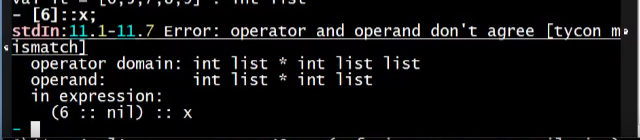


* All elements should have the same type



* This reads as 5 “cons to” x -> add 5 in the front of the list
* “::” adds the left element to the right element/list

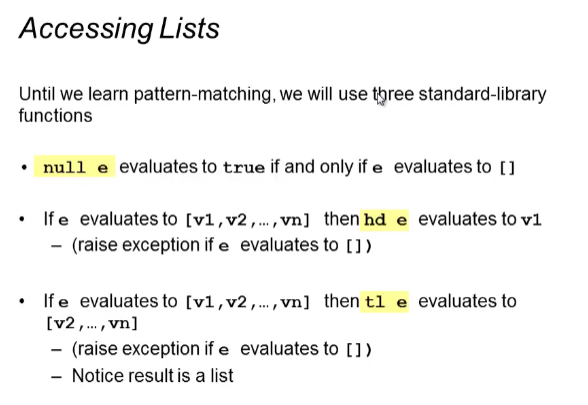
But you can’t “cons” a list to an element/list



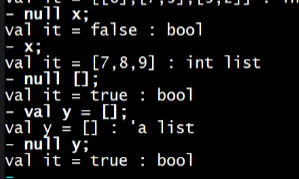
* Because we are adding a [int] in front of an [int]. this will result to [ [int], int, int…] which will not have the same type anymore!

You can add that [6] to a [ [int], …] type:



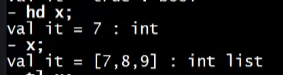


null <list>



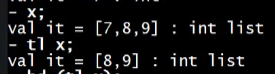
hd <list> (head)

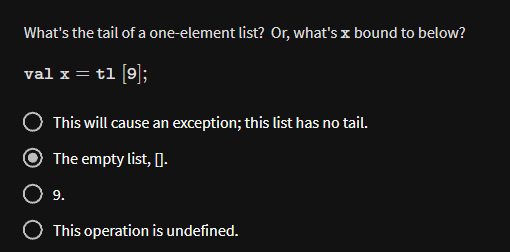
* FIRST element of the list

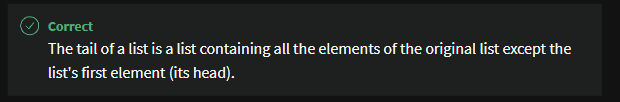


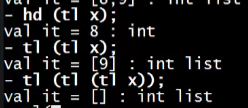
tl <list> (tail)

* REST of the elements minus the FIRST

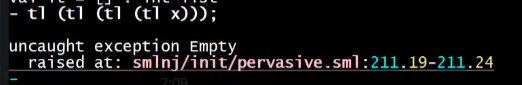


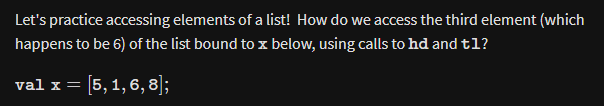


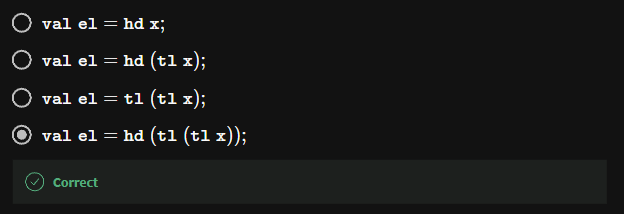


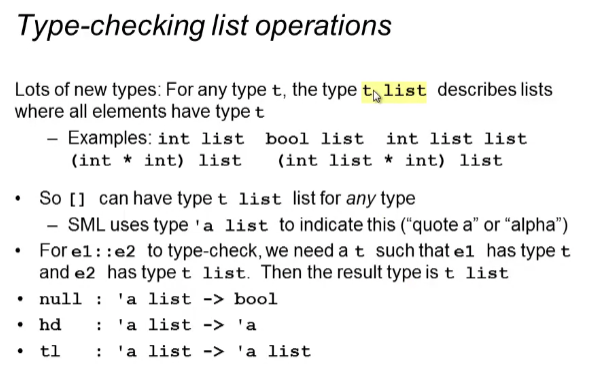


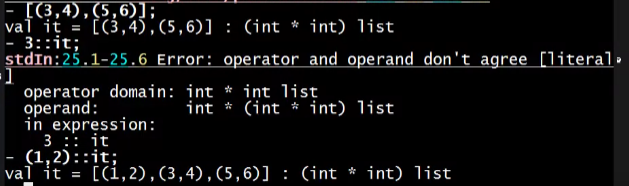
When you are trying to get the hd or tl of an EMPTY list:





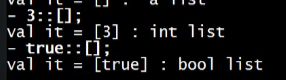






Type of an empty list:



* Type of ‘a (alpha) list
  + This means that you can replace that ‘a to any type you want once you add an element to the list
  + bool list, int list …
  + 
  + Means that you can cons to ANY type of element into an empty list



* A function here in SML for checking if a list is empty

