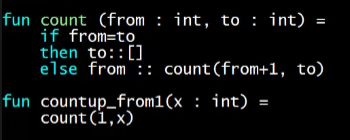
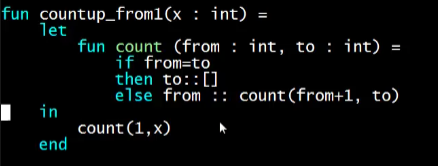


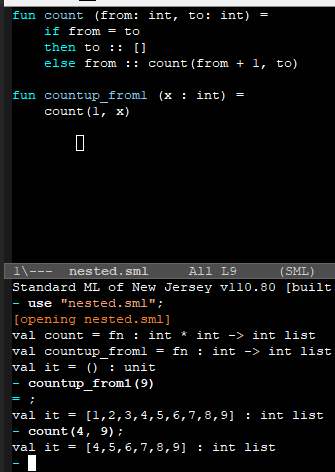
Example:



But I want my count function to be used only by countup\_from1 (private), so we use let expressions

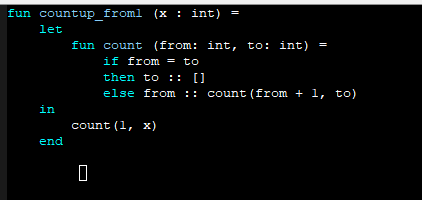


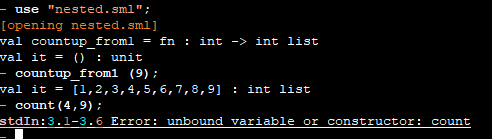
Behavior if count is outside:



We can still use count() function here

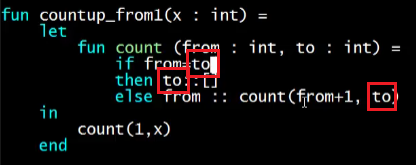
Behavior if count is a let expression



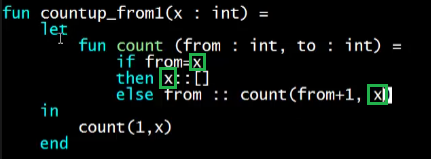


Count is not accessible already. It is only used inside countup\_from1

Refactoring countup\_from1



Since count is already inside the countup\_from1 function, it will always take “x” as “to”, better yet use “x” for all the instances of “to”



Then looking at our function, we do not need the parameter “to” anymore, so we delete it. And also, check the recursive call (and all other calls to count), we need to also delete the 2nd argument since count now only takes one argument after delete “to” parameter

