

```
fun append (xs,ys) =  
  if xs=[]  
  then ys  
  else (hd xs)::append(tl xs,ys)  
  
fun map (f,xs) =  
  case xs of  
    [] => []  
  | x::xs' => (f x)::(map(f,xs'))  
  
val a = map (increment, [4,8,12,16])  
val b = map (hd, [[8,6],[7,5],[3,0,9]])
```

Programming Languages

Dan Grossman
2013

Introduction to Section 4: Remaining ML Topics

Remaining Topics

- Type Inference
 - Mutual Recursion
 - Module System
 - Equivalence
-
- No homework assignment focused on this material
 - But some will be on the midterm exam

Next section:

Start using Racket for more programming-languages concepts