

```
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

Optional: Are All PLs the Same?

All cars are the same

- To make it easier to rent cars, it is great that they all have steering wheels, brakes, windows, headlights, etc.
 - Yet it is still uncomfortable to learn a new one
 - Can you be a great driver if you only ever drive one car?
- And maybe PLs are more like cars, trucks, boats, and bikes
- So are all PLs really the same...

Are all languages the same?

Yes:

- Any input-output behavior implementable in language X is implementable in language Y [Church-Turing thesis]
- Java, ML, and a language with one loop and three infinitely-large integers are “the same”

Yes:

- Same fundamentals reappear: variables, abstraction, one-of types, recursive definitions, ...

No:

- The human condition vs. different cultures
(travel to learn more about home)
- The primitive/default in one language is awkward in another
- Beware “the Turing tarpit”