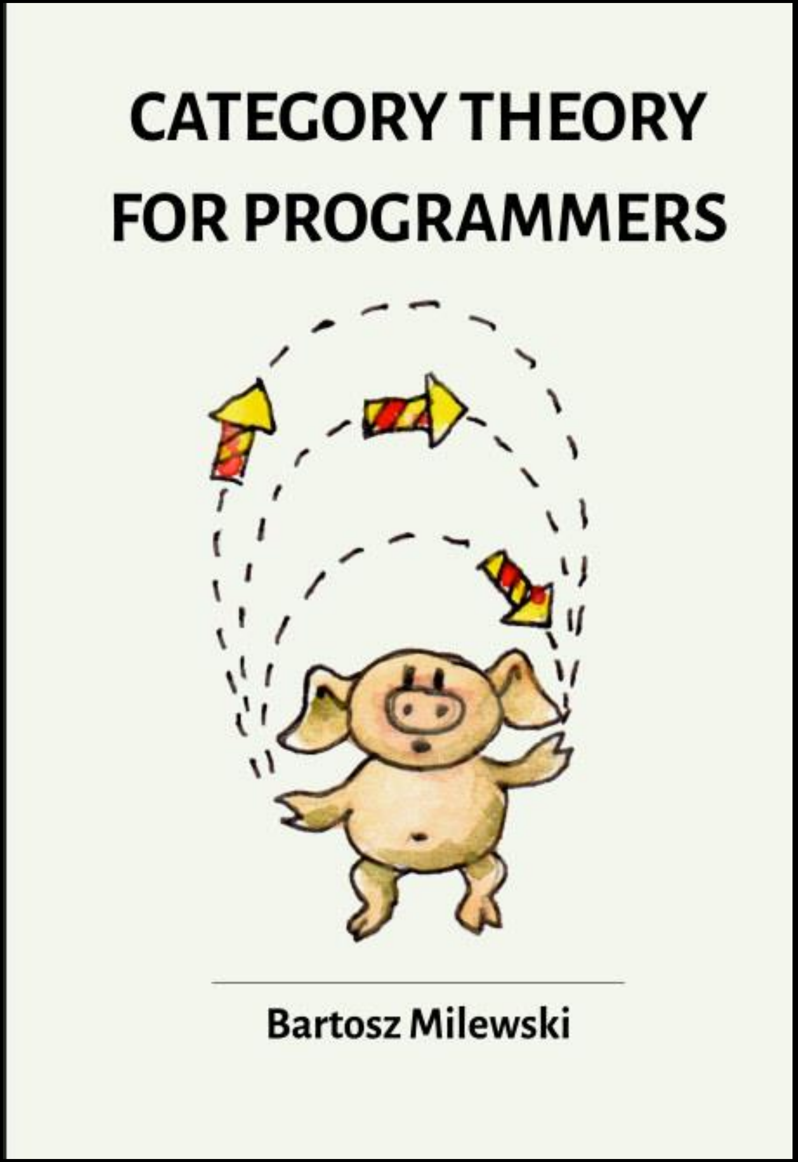
Java

Meetup

Friendly Environment Policy

Berlin Code of Conduct

DISCORD



Category Theory for Programmers

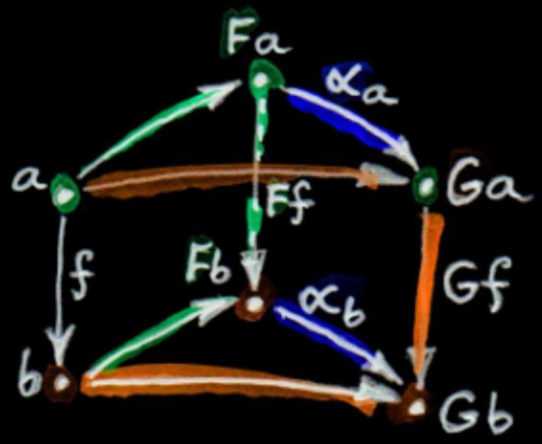
Chapter 17:

It's All About Morphisms

If I haven't convinced you yet that category theory is all about morphisms then I haven't done my job properly. Since the next topic is adjunctions, which are defined in terms of isomorphisms of hom-sets, it makes sense to review our intuitions about the building blocks of hom-sets. Also, you'll see that adjunctions provide a more general language to describe a lot of constructions we've studied before, so it might help to review them too.

17.3 Natural Transformations

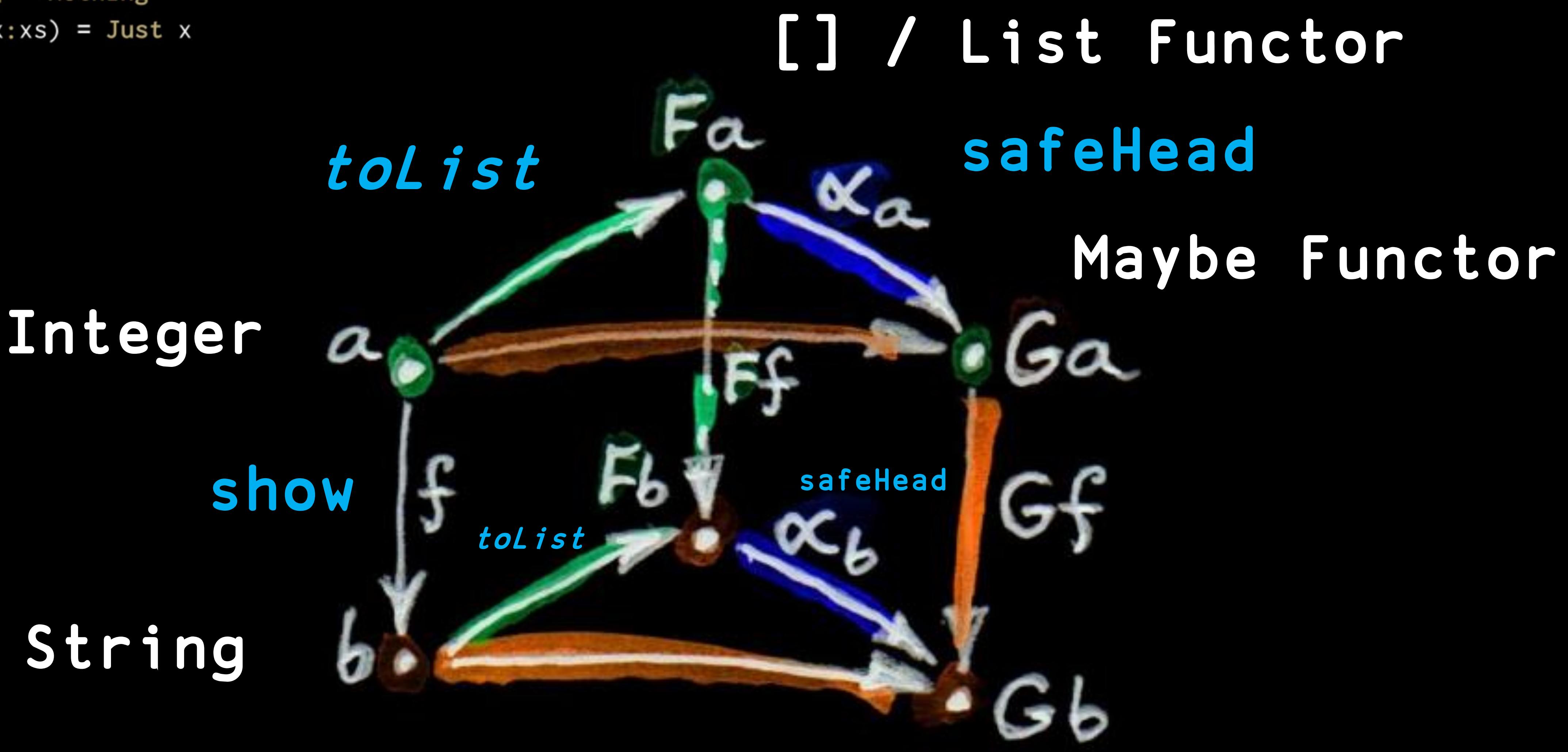
In general, natural transformations are very convenient whenever we need a mapping from morphisms to commuting squares. Two opposing sides of a naturality square are the mappings of some morphism f under two functors F and G. The other sides are the components of the natural transformation (which are also morphisms).



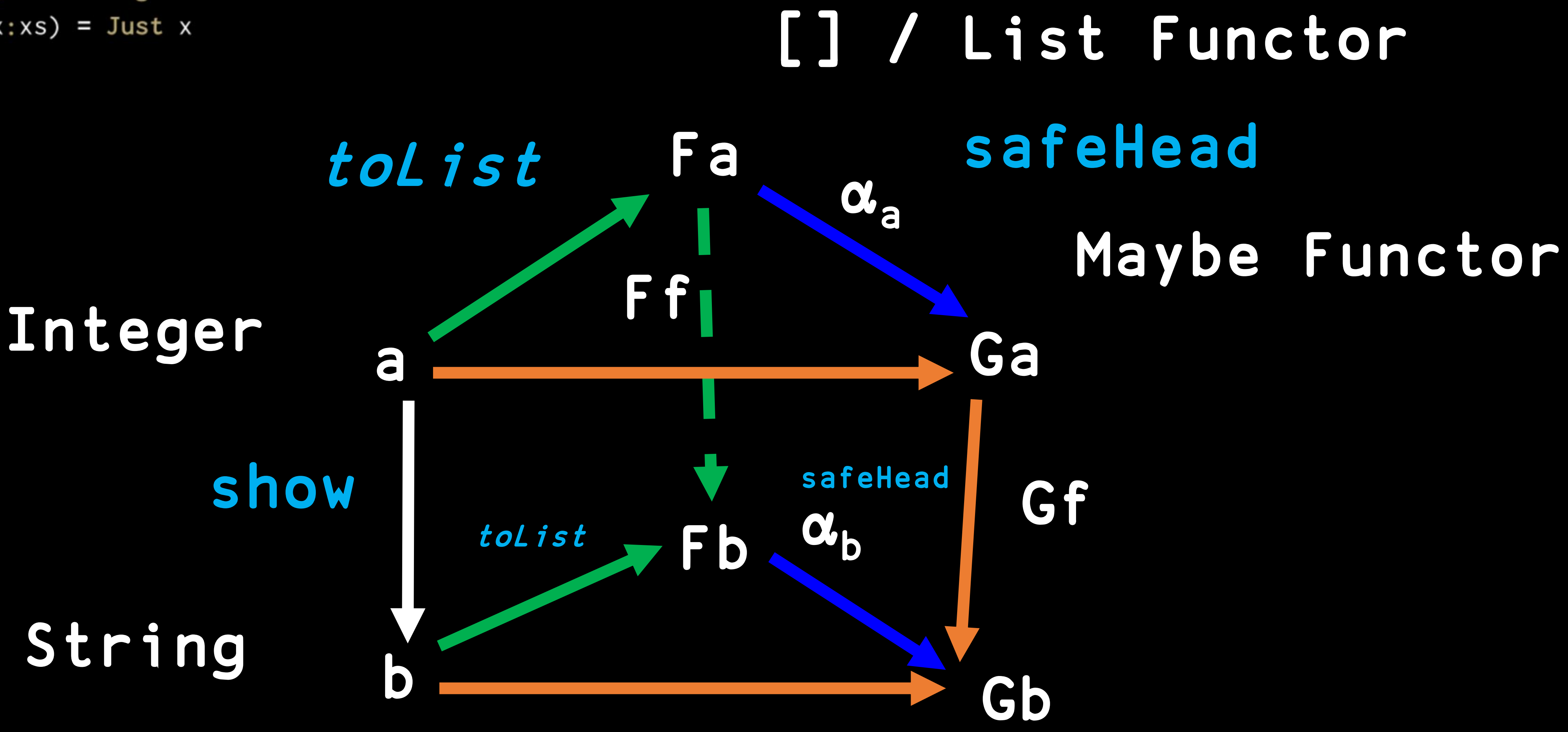
17.8 Challenges

1. Consider some degenerate cases of a naturality condition and draw the appropriate diagrams. For instance, what happens if either functor F or G map both objects a and b (the ends of $f :: a \to b$) to the same object, e.g., $Fa = Fb$ or $Ga = Gb$? (Notice that you get a cone or a co-cone this way.) Then consider cases where either $Fa = Ga$ or $Fb = Gb$. Finally, what if you start with a morphism that loops on itself — $f :: a \to a$?

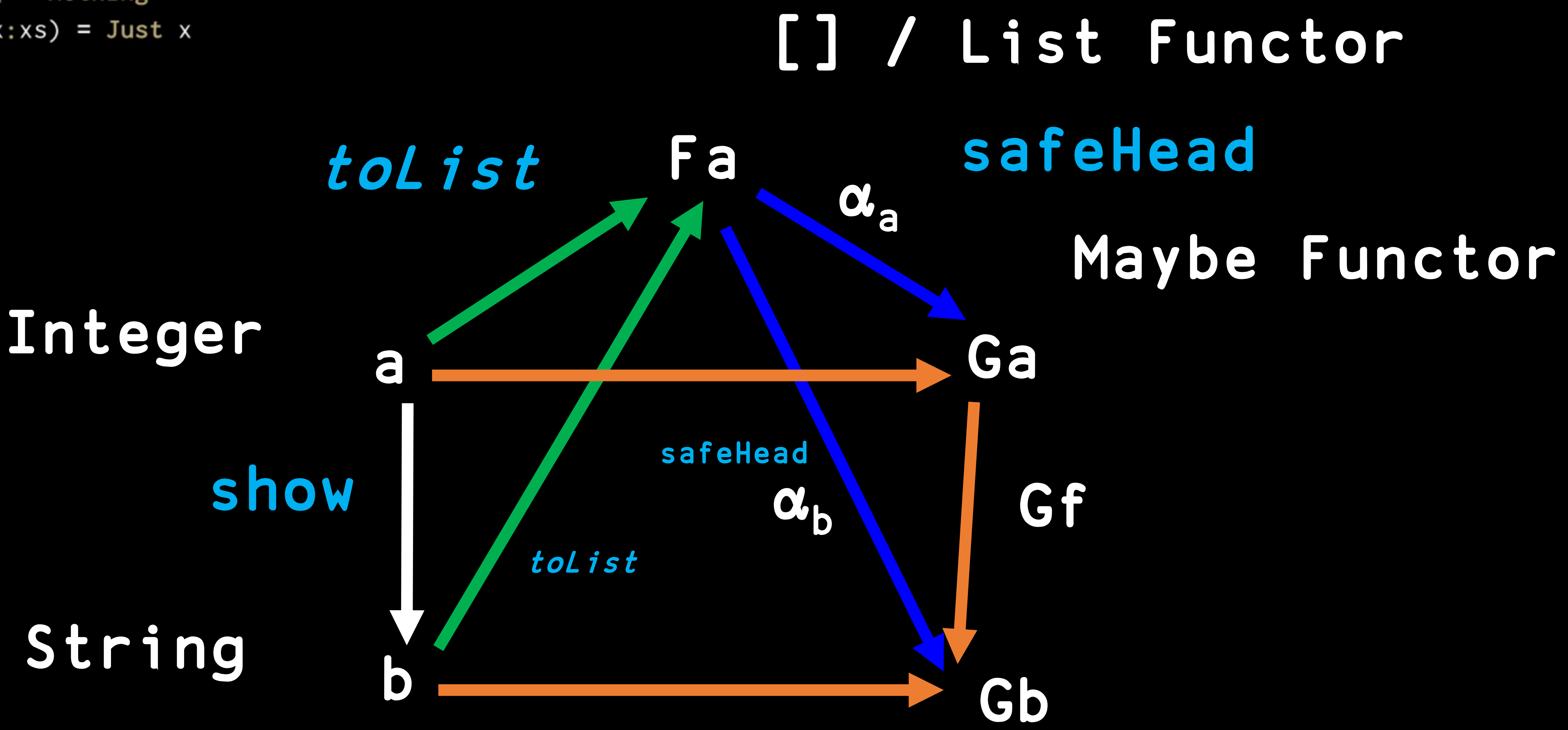
```haskell
safeHead :: [a] -> Maybe a
safeHead [] = Nothing
safeHead (x:xs) = Just x
```



```haskell
safeHead :: [a] -> Maybe a
safeHead [] = Nothing
safeHead (x:xs) = Just x
```



```haskell
safeHead :: [a] -> Maybe a
safeHead [] = Nothing
safeHead (x:xs) = Just x
```



Java

Meetup