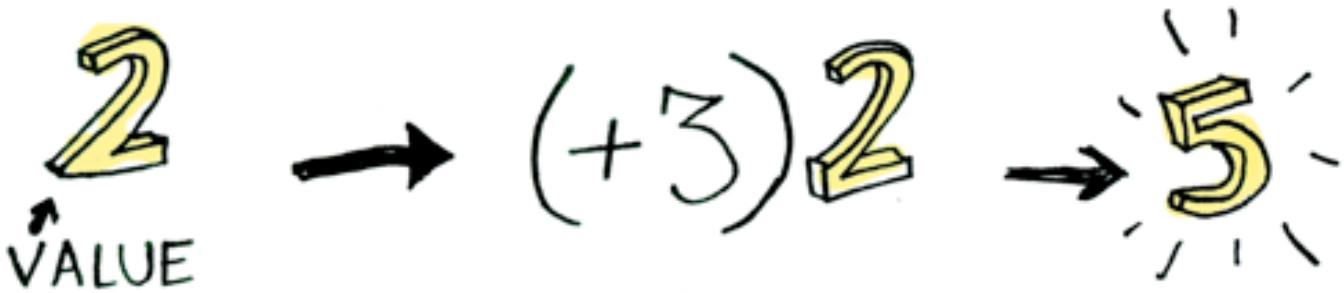
FIT2102 PASS - Week 8

Nicholas Cheng

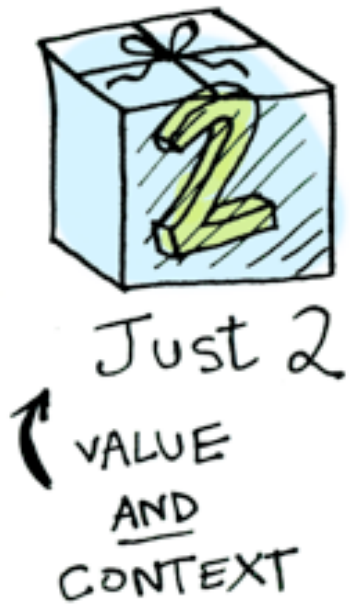
October 3, 2020

Function application



Functors

The **main** difference now is that values are inside of a *context*.



Functors

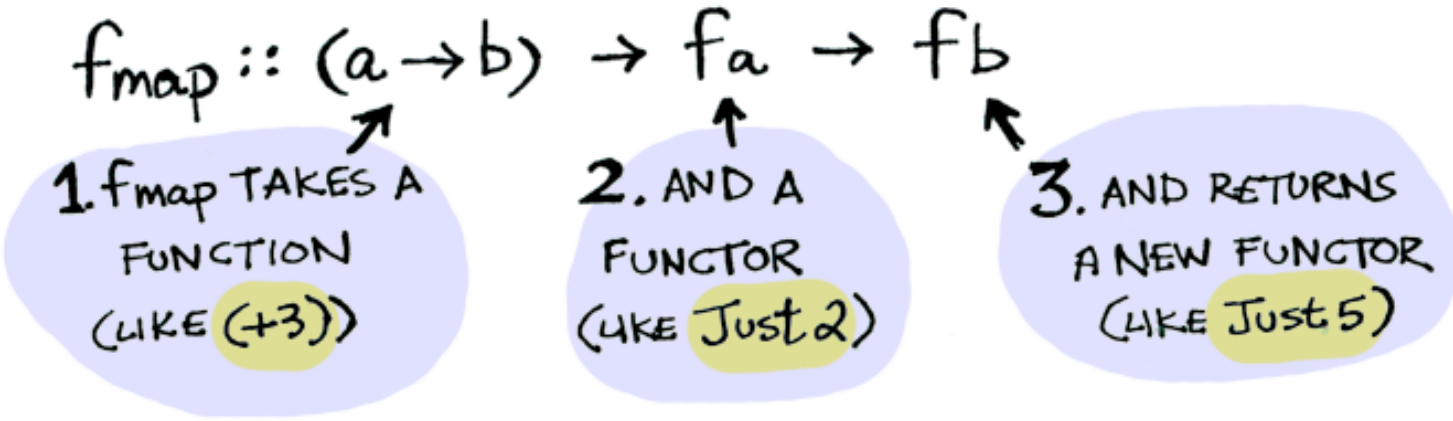
Does this make a difference? **Yes**, but **no**. Lemme explain.

Functors

Ideally, we'd want something that does this.

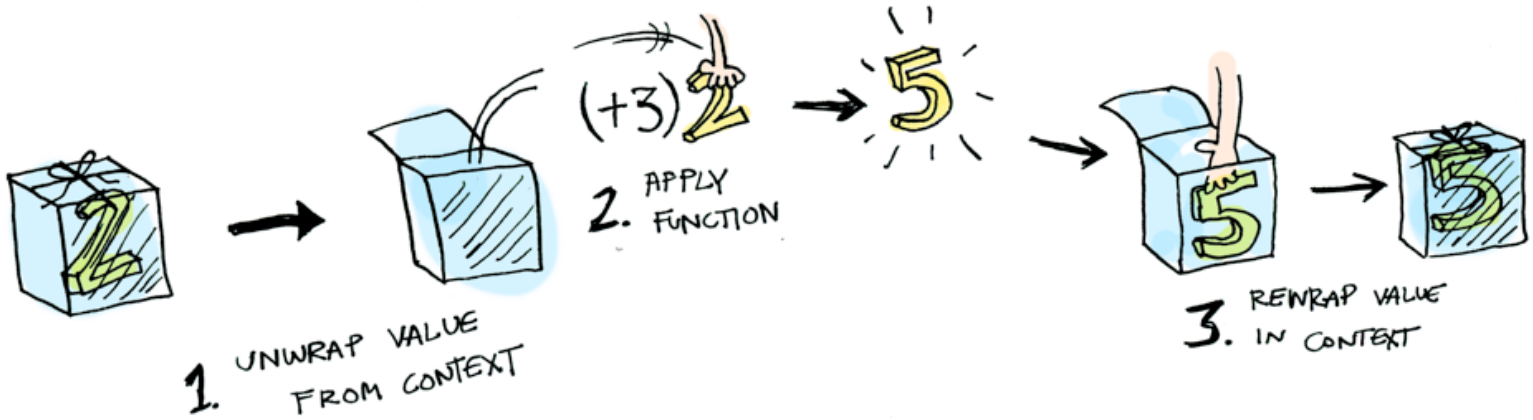
Functors

What the fmap?!



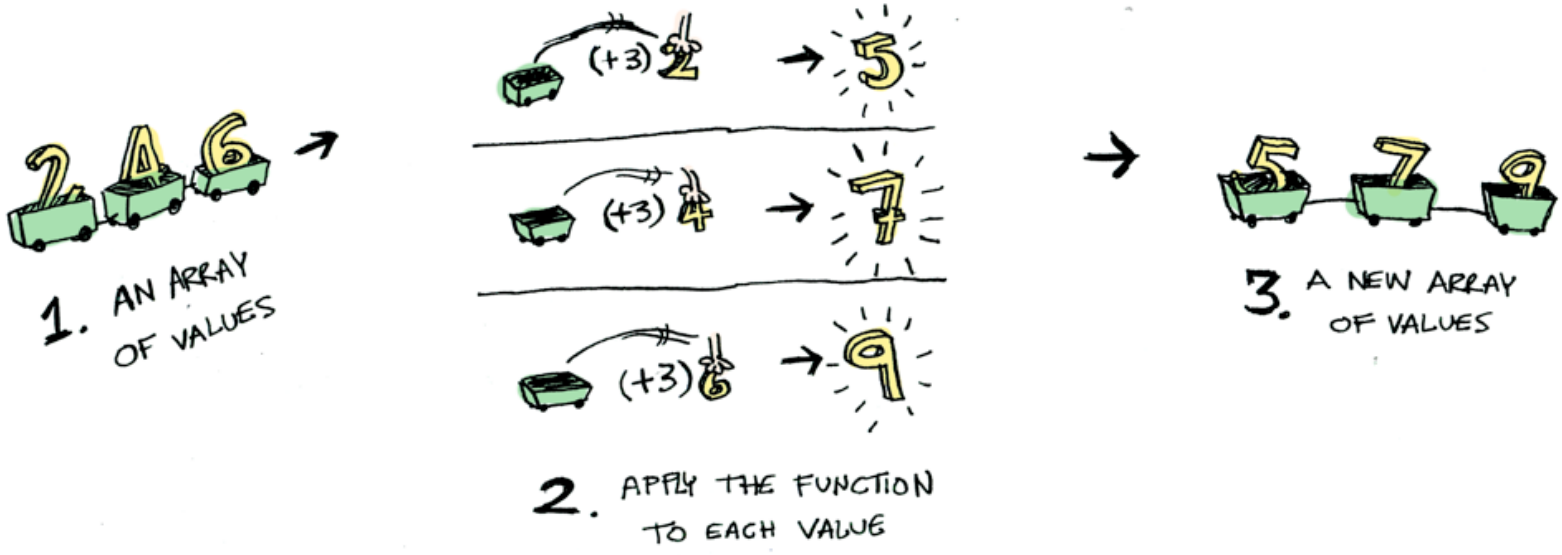
Functors

Stop wasting my time and just tell me what to do!



Say, this looks awfully familiar.

Functors

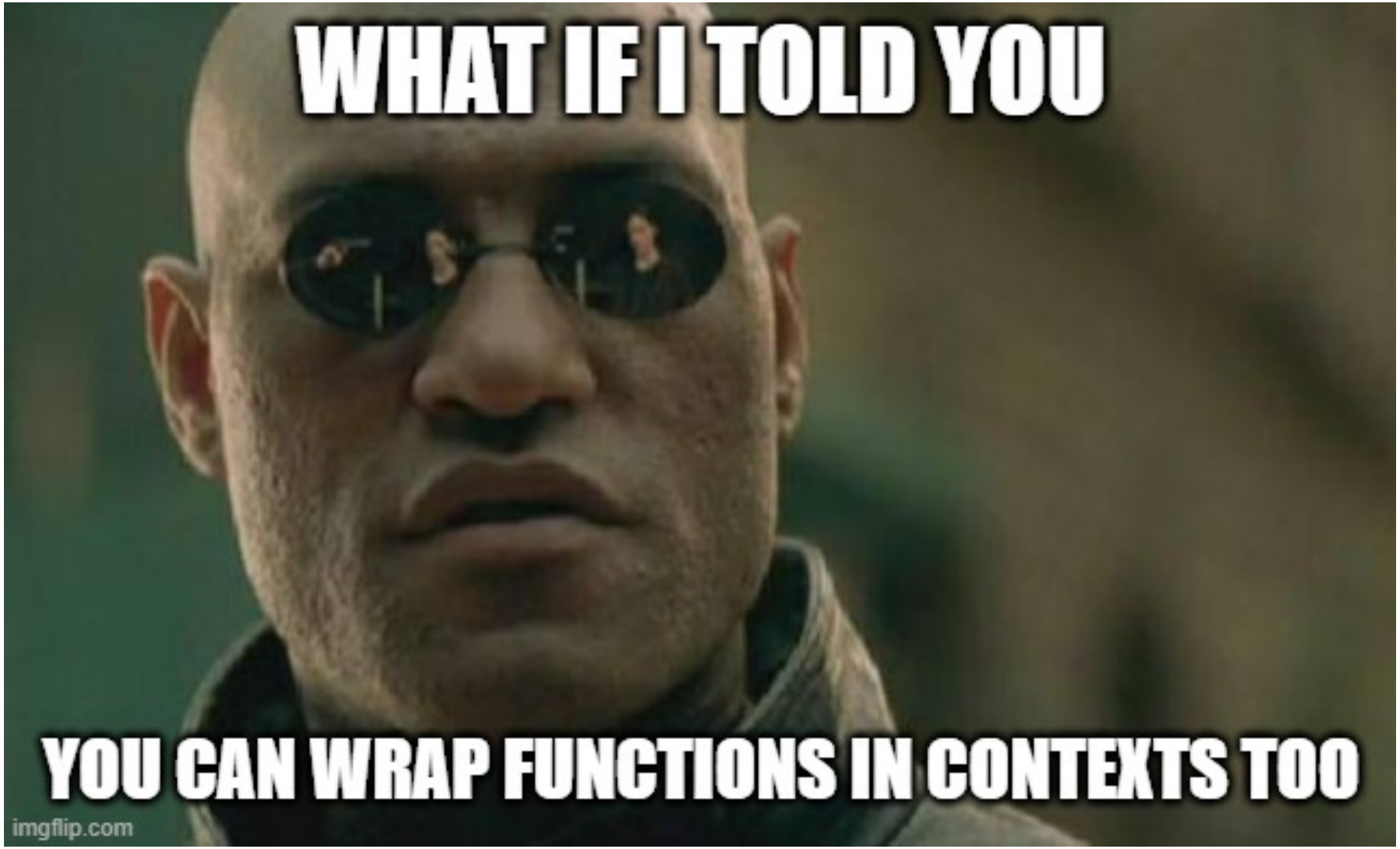


And lists too?

Functors

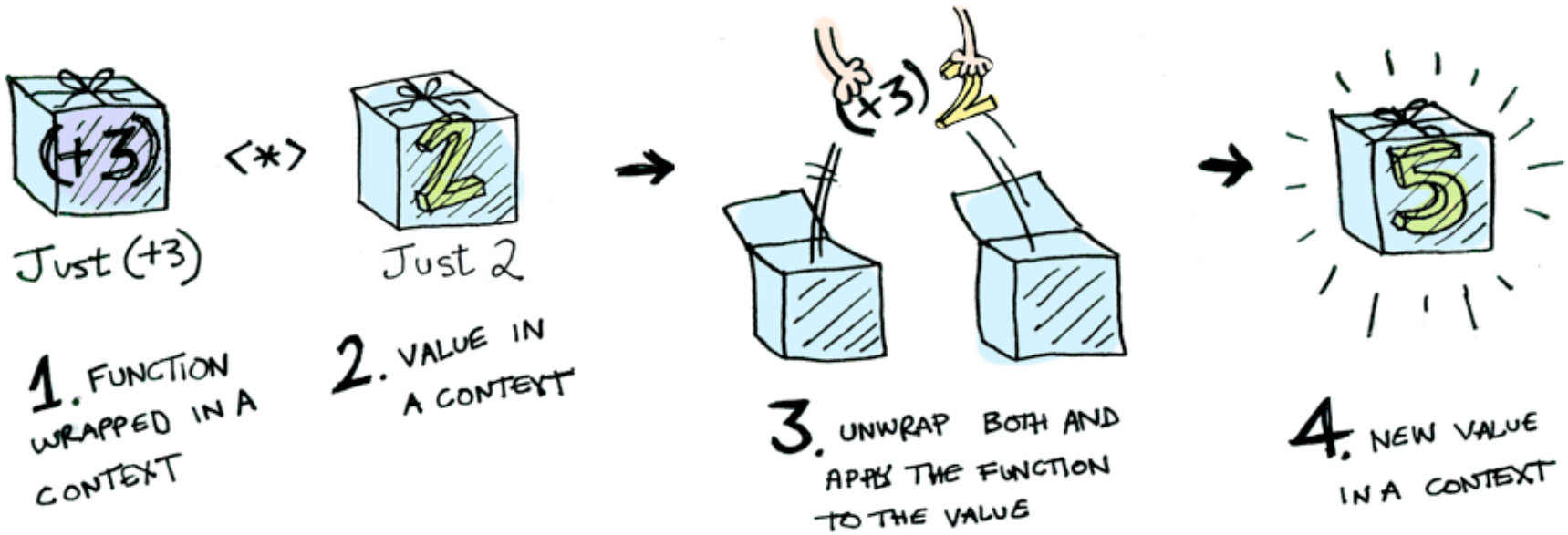
Applicatives

Red pill, blue pill or purple pill?



Applicatives

Red pill it is. . .



Point-free code

In the land of the free. . .

`f x = 1 + x` vs. `f = (1+)`

Point-free code

Eta conversion

```
f x = g x
  f = g
```

Point-free code

Operator sectioning

```
x + y = (+) x y
x + y = ((+) x) y
x + y = (x+) y
```

Point-free code

Composition

```
(f . g) x = f (g x)
```

Bibliography

[1] Aditya Bhargava. *Functors, Applicatives,And Monads In Pictures*. 2013. URL: http://adit.io/posts/2013-04-17-functors,_applicatives,_and_monads_in_pictures.html.

[2] Tim Dwyer. *Functor and Applicative*. 2020. URL: https://tgdwyer.github.io/haskell3/.