Immutability

and friends

What’s immutability?

Property of a “thing” which indicates such a “thing”, once crated, cannot be modified/changed/altered.

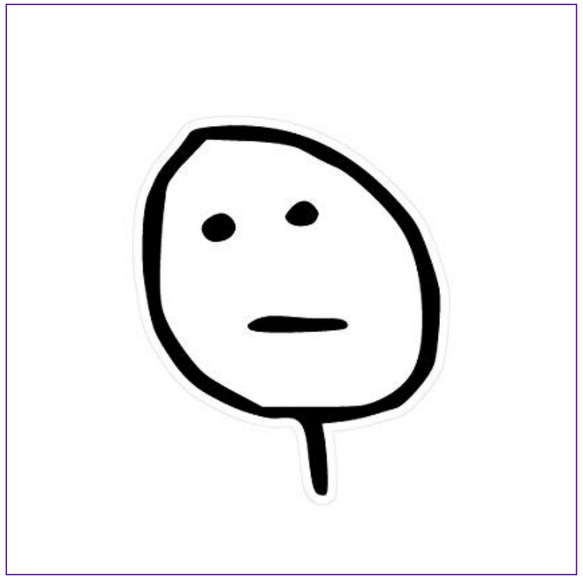
Where can I find immutability?

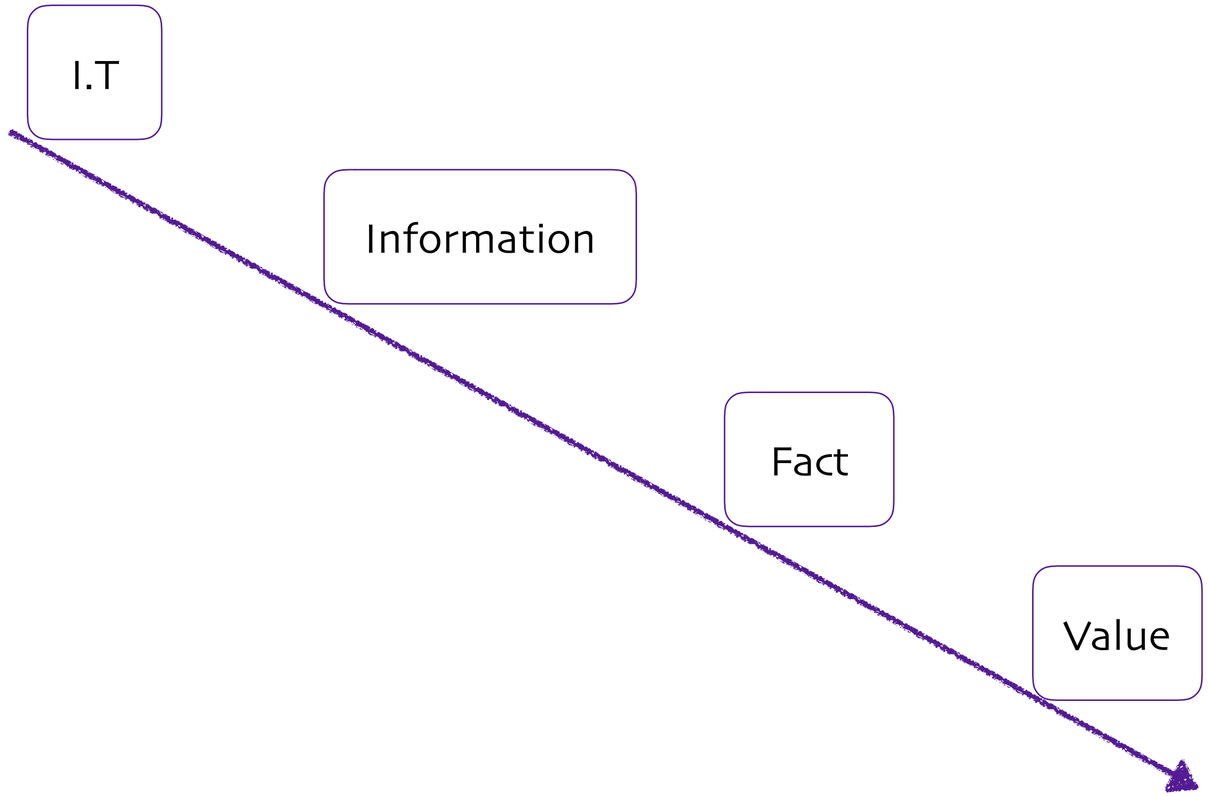
- All primitive types
- Objects
- Data structures
- Infrastructure

then, Why mutation?

Mainly because of:

- Performance
- **Resources**
- That's how we learnt (unfortunately)





The value of **values**

Why prefer immutability

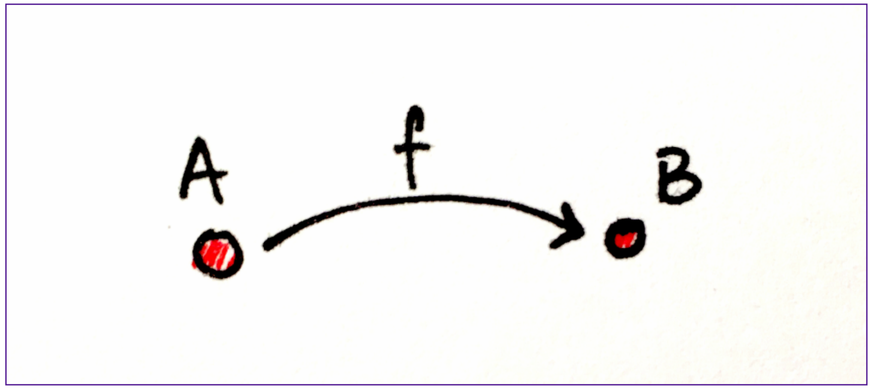
- Time traveling
- Undo/Redo
- Performance
- Security
- Thread safety

- Immutable structures are simpler to understand and reason about
- Help us achieve referential transparency

f-ing me

What's a *f*unction?

- A function is a computation that takes in an **A** as argument and returns a **B** as result.
- From category theory, a function is represented as a morphism (arrow).



Properties of a *function*

functions are:

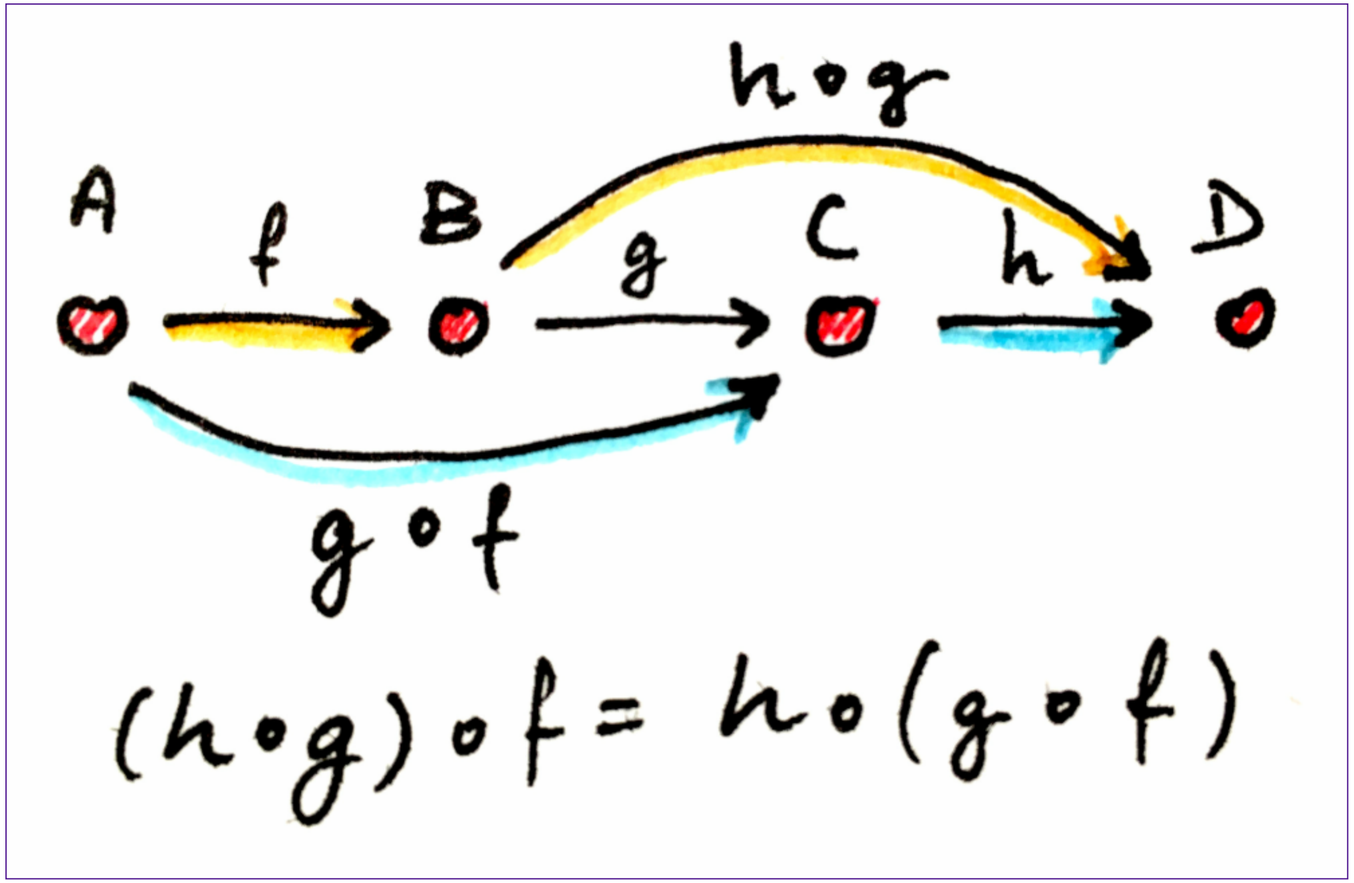
1) Total: for every input, they return and output.

2) Deterministic: for the same input, they return the same output.

3) Pure: they do nothing more than computing their output.

Super *f*

- Composition
  - Identity
  - Associativity
- Higher order



Okay, cool ‘n all but, Where are the examples? 🧐

Examples

1)No variables, just values

2)Working with immutable objects

3)Optics

4)Immutable data structures

5)Streams*

Special *effects*

Immutability and *f* riends

- Pure functions
- Referential transparency

Demo time!

Unfold API model.

Bye!

and don’t mutate!