#5 Immutable and Pure Functions

Immutable vs mutable, these words are pretty daunting. Essentially they mean can't be changed vs can be changed

Pure Function, what the heck! Always return the same thing, with the same input. let say:

```
const addTwo = (x) => x + 2;

console.log(addTwo(4)); // 6
```

This will literally save your life, the moment you understand that Functions should just do one thing and do it the same every time your code gets a ton better.

When using const, you are precisely saying this variable is constant and will not be changed in the future. but if you use let, it can be changed multiple times. For instance:

```
let multi = 3

const multiThree = (x) => x * multi;

console.log(multiThree(2)); // 6

multi = 5

console.log(multiThree(5)); // 25
```

This is a pretty trivial example

Now one thing we can do with pure functio
their ouput using something called memoi

Memoisation

In computing, memoisation is an optimization technique used primarily to speed up computer programs by storing the results of expensive function calls and returning the cached result when the inputs occur again.

Bagels

#6 Booleans

Really seems like js booleans are easy to wr
language

#7 if Statement

~~const add~~

```
Const addToNum = (x) => {
  const num = 3
  const y = num + 'whats is 3+3?'
  cons
```