**1. Pure Functions**

To be considered pure, functions must fulfil the following criteria:

* they must be predictable
* they must have no side effects

**Pure functions must be predictable**

Identical inputs will always return identical outputs, no matter how many times a pure function is called. In other words: we can run a pure function as many times as we like, and given the inputs remain constant, the function will always predictably produce the same output. Kind of like when you're a pizza-loving person with lactose intolerance. No, this time won't be different, so stop ogling that 16-incher your flatmate ordered.

**Pure functions must have no side-effects.**

A side-effect is any operation your function performs that is not related to computing the final output, including but not limited to:

* Modifying a global variable
* Modifying an argument
* Making HTTP requests
* DOM manipulation
* Reading/writing files

A pure function must both be predictable and without side-effects. If either of these criteria is not met, we're dealing with an impure function.

An impure function is kind of the opposite of a pure one - it doesn't predictably produce the same result given the same inputs when called multiple times, and may cause side-effects. Let's have a look at some examples.

Examples will be in js file (pureVSimpureFunctions.js)