Constructor Pattern

- We call a constructor function that because it is intended to be called with the `new` operator, otherwise its just a regular JS function
- We capitalize the functions name, which is not a syntax requirement, but a convention to reveal intent to be used as constructor
- If we call a function without the `new` operator, `this` will be pointing to the global object and you'll define properties and functions on it, not an instance

When we call a function with the `new` operator, the following happens:

1. A new object is created
2. `this` in the function is set to point to the new object
3. The code in the function is executed
4. `this` is returned if the constructor doesn't explicitly return an object (must be an object!)
   - It can be a subtle bu gif you don't take not of the implication of not returning an object