JS Functions, Day 9, 18 June 2014

- From this point forward, use `"use strict";` to the top of your JS scripts.
- For more clean and robust JS, run your JS through http://jslint.com/ **Caution: hurt feelings!**
- Using `console.log()` to log stuff to the console is OK, but it's not the same as `return`.
- Use `return` at the end of your functions in order for the function to return a value.
- functions without a return return `undefined`.

The value of returning values from functions

```
// Log true if the input is equal to 21, otherwise log false.
function isBlackJack(input) {
    if (input == 21) {
        console.log(true);
    } else {
        console.log(false);
    }
}

var outcome = isBlackJack(21);
if (outcome == true) {
    alert('Black-Jack 21, you WIN!');
} else {
    alert('YOU LOSE!');
}
```

Defining Functions

```
function square(number) {
    return number * number;
}

// The function definition defines the order of parameters.
function divide(numerator, divisor) {
    return numerator / divisor;
}
```

Calling Functions

`square(5);` This function returns 25 but does not console.log it (unless you typed it in your console.)

`console.log(square(5))` evaluates `square(5)` and then console logs the that returned value.

Copy and paste the divide function definition above into your browser's JS console. Then run the following:

```
var someNumerator = 1;
var someDenominator = 2;

divide(someNumerator, someDenominator);

divide(someDenominator, someNumerator);
```

FUNCTIONS SHOULD DO ONE THING. THEY SHOULD DO IT WELL. THEY SHOULD DO IT ONLY.