ANONYMOUS FUNCTIONS

EXAMPLE: SYNTACTIC SUGAR

```
function something() {
    console.log("regular function");
}
var something2 = function() {
    console.log("anonymous function");
}

// these two functions are essentially the same
something();
something2();
```

EXAMPLE: FUNCTION AS A PARAMETER

```
function performOperation(x, y, operation) {
    console.log("performing operation with inputs: " + x + " " + y);
    var z = operation(x, y);
    console.log("result: " + z);
}

var multiply = function(a, b) {
    return a * b;
}

var getGreaterNumber = function(a, b) {
    if(a > b) {
        return a;
    } else {
        return b;
    }
}
```

EXAMPLE: FUNCTION AS A PARAMETER (2)

- WHAT IS PRINTED HERE?

```
performOperation(2, 3, multiply);
performOperation(10, 10, multiply);
performOperation(5, 6, getGreaterNumber);
performOperation(5, -6, getGreaterNumber);
```

EXAMPLE 2: FUNCTION AS A PARAMETER

```
function identity(x) {
    console.log("x is: " + x);
}

function negative(x) {
    var y = -x;
    console.log("-x is: " + y);
}

function doSomeThings(callback) {
    callback(3);
    callback(4);
    callback(5);
}
```

EXAMPLE 2: FUNCTION AS A PARAMETER (2)

- WHAT IS PRINTED HERE?

```
doSomeThings(identity);
doSomeThings(negative);

doSomeThings(function(x) {
    var squared = x * x;
    console.log("x squared is: " + squared);
});
```

EXAMPLE 3

```
function doWork(onSuccess, onError) {
    function contrivedFunction() {
        return Math.random() < .5;
    }

    var didSucceed = contrivedFunction();

    if (didSucceed) {
        onSuccess();
    } else {
        onError();
    }
}
```

EXAMPLE 3 (2)

```
doWork(
    function() {
        console.log("woo! success!");
    },
    function() {
            console.log("error :c");
    }
)
```

EXERCISE

- CREATE A FUNCTION `findSum` THAT TAKES IN TWO PARAMETERS AND RETURNS THE SUM OF THOSE PARAMETERS.
- CREATE A FUNCTION `findProduct` THAT TAKES IN TWO PARAMETERS AND RETURNS THE DIFFERENCE OF THOSE PARAMETERS.

EXERCISE (CONTINUED)

- CREATE A FUNCTION `threeOperation` THAT TAKES IN TWO PARAMETERS. NAMED `x` AND `operation`. THE FIRST PARAMETER IS A NUMBER. THE SECOND PARAMETER IS A FUNCTION.
  - `threeOperation` SHOULD CALL THE `operation` PARAMETER AS A FUNCTION. IT SHOULD PASS THE NUMBER 3 ALONG WITH THE `x` PARAMETER TO THAT FUNCTION.

EXERCISE (CONTINUED)

- CALL `threeOperation` WITH THE VALUES OF `4` AND `findSum`. CHECK THAT YOUR ANSWER IS `7`.
- CALL `threeOperation` WITH THE VALUES OF `5` AND `findSum`. CHECK THAT YOUR ANSWER IS `8`.
- CALL `threeOperation` WITH THE VALUES OF `4` AND `findProduct`. CHECK THAT YOUR ANSWER IS `12`.
- CALL `threeOperation` WITH THE VALUES OF `5` AND `findProduct`. CHECK THAT YOUR ANSWER IS `15`.

WHAT IS THIS FOR?