8 How to Define and Use Functions

Functions are central to coding in JavaScript. Defining a function in JavaScript is simple. For example, the formula for finding the length of the hypoteneuse of a right triangle is to take the square root of the sum of the squares of the lengths of the two sides. E.g, if the sides are 3 and 4, then this JavaScript would make that calculation:

```
Math.sqrt(3 * 3 + 4 * 4)
```

Open up the JavaScript Console in your browser and enter the above. You should see 5 returned.

We can turn this into a function that types two arguments, A and B, :

```
function hyp(a, b) {
   return Math.sqrt(a*a + b*b);
}
```

The keyword **function** says "define a function." **hyp** is the name of the function, and **a** and **b** are the parameters. The keyword **return** says that the function returns a value, and that value is whatever the expression that follows calculates.

Enter the above into your JavaScript Console, then enter

```
hyp(3, 4)
```

This says "call the **hyp(a, b)** function, with **a** = 3 and **b** = 4." Again, you should see 5 returned. Try some other values.

Arrow Functions

More recent versions of JavaScript support another syntax for defining functions. Instead of writing

```
function hyp(a, b) {
   return Math.sqrt(a*a + b*b);
}
```

you can write this instead:

```
var hyp = (a, b) => Math.sqrt(a*a + b*b);
```

The most common "arrow" syntax is *variable-list* => *return-value*. This syntax is particularly handy when creating functions to pass to array functions like **map()** and **forEach()**.

More details on this syntax can be found at MDN.