**JavaScript Functions — Understanding the Basics**

What is a Function?

A function is a subprogram designed to perform a particular task.

Using a function is like hiring someone who is skilled to do a job. If you hire a painter, you explain how you want it, the painter does the job, and you get the result.

There are a few different ways to define a function.

1. A Function Declaration defines a named function. When using function declarations, the function definition is hoisted, thus allowing the function to be used before it is defined.

Function name(parameters) {

Statements

}

1. A Function Expressions defines a named or anonymous function.

Function Expressions are not hoisted and therefore cannot be used before they are defined.

Let name = function(parameters) {

Statements

}

1. An Arrow Function Expression is a shorter syntax for writing function expressions.

let name = (parameters) => {

statements

}

Functions are executed when they are called, and this is a known as invoking a function. You can invoke a function by referencing the function name, followed by an open and close parenthesis: ().

Values can be passed into functions and used within the function. The name of the value is called a parameter. The actual value itself is called an argument.

Functions always return a value. In JavaScript, if no return value is specified, the function will return undefined by default.

Functions are objects. Objects in JavaScript are extremely versatile. Because of this, we can even pass a function as a parameter into another function. – it is called a higher order function.