

Stefano Schmidt
Launch School
Introduction to Programming With JavaScript
Functions – Exercises

1: This exercise asks us what the following code logs to the console and whether executing the function foo affects the output:

```
let bar = 1;
function foo() {
  let bar = 2;
}

foo();
console.log(bar);
```

The answer is that this code will log the value 1 to the console. Executing the function foo will not affect the output; the bar variable declared in the function body has scope only inside the function body, while the bar variable declared at the top of the code has global scope and it's stored value is what will be logged.

2: This exercise asks us to modify the greeter.js program from previous chapter by writing a function that can ask a user for first and last names in separate invocations, and then use the return values of these function calls to greet the user. I did this as follows:

```
function getName(which) {
  let rlSync = require('readline-sync');
  return rlSync.question(`What is your ${which} name? `);
}

console.log(`Hello, ${getName('first')} ${getName('last')}!`);
```

3: This exercise asks us to write a program multiply.js that defines a function multiply that takes in two numbers as arguments and returns their product, and then to write some code that asks the user for two numbers and then uses the function to output the result of multiplying the two numbers given by the user. I did this as follows:

```
function multiply(num1, num2) {
  return num1*num2;
}

function getNum(which) {
  return Number(require('readline-sync').question(`Enter the ${which} number: `));
}

let firstNum = getNum('first');
let secondNum = getNum('second');
console.log(`${firstNum} * ${secondNum} = ${multiply(firstNum, secondNum)}`);
```

4: This exercise asks us to use the times function shown below to write a program factorial.js that logs the value of 1!, 2!, 3!, 4!, and 5! to the console. I did this as follows:

```
function times(number1, number2) {
  let result = number1 * number2;
  console.log(result);
  return result;
}

times(times(times(times(times(1,1), 2), 3), 4), 5);
```

5: This exercise asks us to explain what the following code will log to the console:

```
function scream(words) {
  words = words + '!!!!';
  return;
  console.log(words);
}

scream('Yipeee');
```

The answer is that it will not log anything to the console because the scream() function has the return statement before the console log, and hence the console log will not be executed.

6: This exercise asks us to explain what the following code will log to the console:

```
function scream(words) {  
  return words + '!!!!';  
}  
  
scream('Yipeee');
```

The answer is that it will not log anything to the console because there is no call to the `console.log()` method anywhere in this snippet of code.