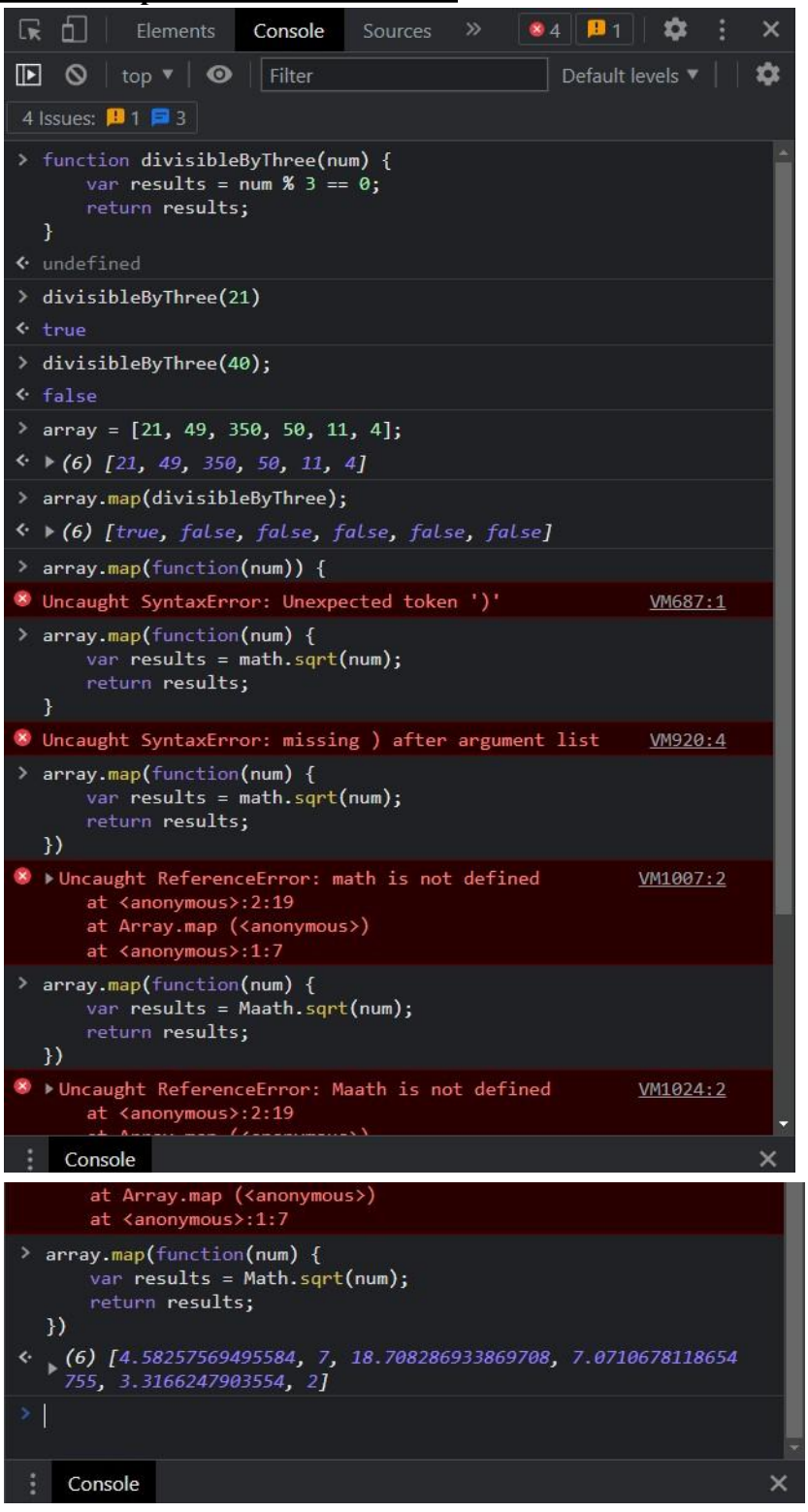
Pichy Jumpholwong & Zac Nakamura
Lab 8: Anon Functions and Callbacks
5/3/22

Task 1: Create an index.html for Lab 8

```
<!DOCTYPE html>
</html>
<html>
    <head>
        <title>Lab 8 - Call Functions and Callbacks</title>
        <script src="js/lab.js"></script>
        <link rel="stylesheet" type="text/css" href="../css/site.css">
        <link rel="stylesheet" type="text/css" href="css/lab.css">
    </head>
    <body>
        <h1>Lab 8: Call Functions and Callbacks</h1>
        <h2>Challenges</h2>
        <p>When writing our function in the javascript file, we did not know why (function(num)) did not work and
        realized that javascript is not like python, therefore the closing parentheses should go after the curly brace.
        </p>
        <h2>Problems</h2>
        <p>We had problems with understanding callback functions and task x because although we were following
        the steps on W3Schools, we did not understand how it could work for our code.</p>
        <h2>Results</h2>
        <p>Check out the console for our results!</p>
        <script src="js/lab.js"></script>
    </body>
</html>

CRLF  UTF-8  HTML  master  Fetch  GitHub  Git (3)
```

Screenshot of our Atom source code

Task 2: Experiment in the Console



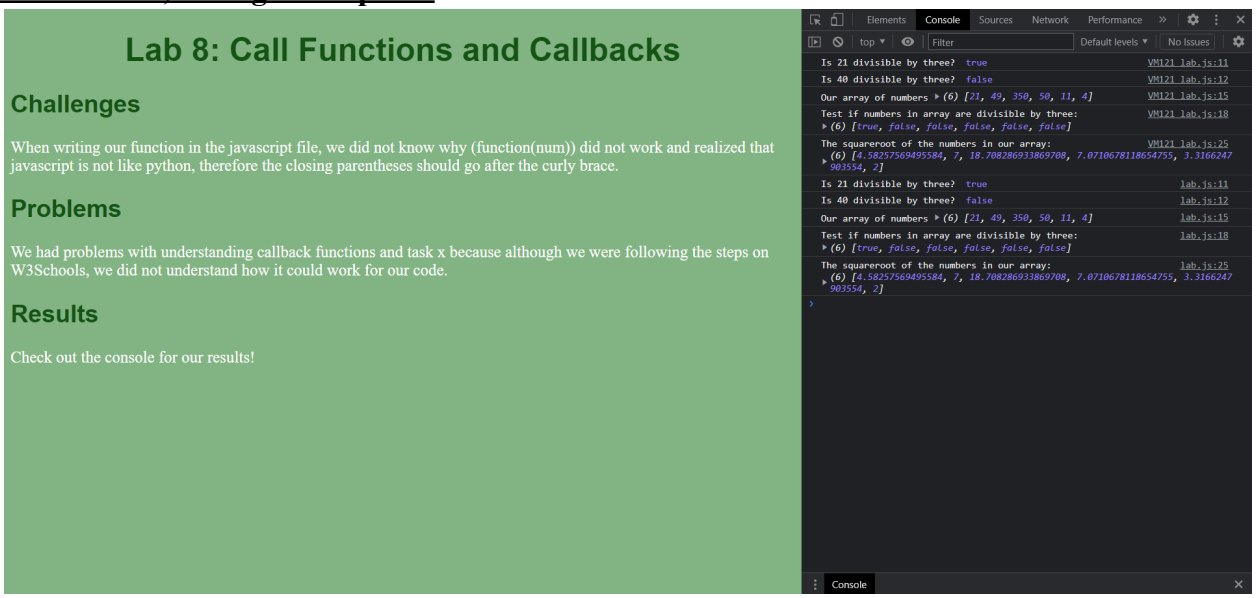
Screenshot of our console results

Task 3: Create a JavaScript file

```
// Authors: Pichy Jumpholwong and Zac Nakamura
// Created: 3 May 2022
// License: Public Domain

function divisibleByThree(num) {
    var results = num % 3 == 0;
    return results;
}

// test function
console.log("Is 21 divisible by three? ", divisibleByThree(21));
console.log("Is 40 divisible by three? ", divisibleByThree(40));

array = [21, 49, 350, 50, 11, 4];
console.log("Our array of numbers", array);

var result = array.map(divisibleByThree);
console.log("Test if numbers in array are divisible by three:", result)

var result = array.map(function(num) {
    var result = Math.sqrt(num);
    return result;
})

console.log("The squareroot of the numbers in our array:", result);

```

Screenshot of our JavaScript file in Atom

Task 4: Test, Debug and Upload



Screenshot of our working file with our console open

Self-Evaluation Rubric:

Self Evaluation Rubric (Pichy Jumpholwong and Zac Nakamura)						
Did you complete the assignment and did you complete it on time?	Submitted on time	Up to 1 day late	Up to 2 days late	Up to 3 days late	4 days late or more	Do you need to clarify?
	✅	☐	☐	☐	☐	
Did you collaborate with a partner?	Worked with partner			Worked alone		Do you need to clarify?
	✅			☐		
Did you put in earnest effort and provide an articulate summary of your experience?	Excellent	Pretty good	About average	Could be improved	Not this time	What supports this?
	✅	☐	☐	☐	☐	
Was the assignment complete, with minimal errors, correct output, and good style?	Excellent	Pretty good	About average	Could be improved	Not this time	What supports this?
	✅	☐	☐	☐	☐	
How much EXTRA effort did you put into the assignment?	A lot of extra effort		Some extra effort		Not this time	What supports this?
	✅		☐		☐	

Summary of your evaluation/efforts:

We forgot at times that javascript needs semicolons, however, instead of going back, we decided to make another note and add semicolons for testing our functions. We put in a lot of extra effort when we were trying to figure out Task X, the Bonus section. We looked at examples on W3Schools and other websites as well to see if we could get different examples and descriptions. Although we could not figure out Task X, we decided to leave our code in the javascript file and comment it out so that we can remember it, go back to it, or just simply have it as a reference. In addition, we also asked another classmate if they had figured it out as well. Although they did not, we still thought it was a great idea to ask them for potential help.