

[Lesson](#)[Tuesday](#)

Introduction to Programming
(/introduction-to-programming)
/ Arrays and Looping (/introduction-to-programming/arrays-and-looping)
/ Separation of Logic: Fixing a Bug in
Text Analyzer

[Text](#)[Cheat sheet](#)

Examples of Bad Separation of Logic

Don't Access/Manipulate the DOM from your Business Logic

The following code is bad:

```
// This is bad! Don't put any logic to alter the DOM in your function!

function numberOfOccurrencesInText(word, text) {
  if (word.trim().length === 0) {
    // I'm directly altering the DOM from my business logic! This is bad.
    document.getElementById("total-count").innerText = 0;
  }
  const textArray = text.split(" ");
  let wordCount = 0;
  textArray.forEach(function(element) {
    if (element.toLowerCase().includes(word.toLowerCase()))
  {
    wordCount++;
  }
});
  // I'm directly altering the DOM from my business logic! This is bad.
  document.getElementById("total-count").innerText = wordCount;
}
```

Don't Include a Message to the User in the Return Values of Your Business Logic Functions

Instead, you should be returning just the value, without a message to the user. Any messages to the user should be in the UI logic. Don do this:

```
// Not so good either, but for less obvious reasons.

function numberOfOccurrencesInText(word, text) {
  if (word.trim().length === 0) {
    return "You need to enter a word!";
  }
  const textArray = text.split(" ");
  let wordCount = 0;
  textArray.forEach(function(element) {
    if (element.toLowerCase().includes(word.toLowerCase()))
  {
    wordCount++;
  }
  });
  return "There are " + wordCount + " total matches!";
}
```

Tests

Here are all the tests we wrote for both `wordCounter()` and `numberOfOccurrencesInText()`. **This set of tests now includes the test we wrote to fix the bug that was caused when the `word` variable was an empty string `""`.**

These should provide a good sense both of what a pseudocode test should look like and a general progression from simplest behavior to more complex behavior.

Describe: wordCounter()

Test: "It should return 1 if a passage has just one word."

Code:

```
const text = "hello";
```

```
wordCounter(text);
```

Expected Output: 1

Test: "It should return 2 if a passage has two words."

Code:

```
const text = "hello there";
```

```
wordCounter(text);
```

Expected Output: 2

Test: "It should return 0 for an empty string."

Code: wordCounter("");

Expected Output: 0

Test: "It should return 0 for a string that is only space s."

Code: wordCounter(" ");

Expected Output: 0

Test: "It should not count numbers as words."

Code: wordCounter("hi there 77 19");

Expected Output: 2

Describe: numberOfOccurrencesInText()

Test: "It should return 0 occurrences of a word for an empty string."

Code:

```
const text = "";
```

```
const word = "red";
```

```
numberOfOccurrencesInText(word, text);
```

Expected Output: 0

Test: "It should return 1 occurrence of a word when the word and the text are the same."

Code:

```
const text = "red";
const word = "red";
numberOfOccurrencesInText(word, text);
Expected Output: 1
```

Test: "It should return 0 occurrences of a word when the word and the text are different."

Code:

```
const text = "red";
const word = "blue";
numberOfOccurrencesInText(word, text);
Expected Output: 0
```

Test: "It should return the number of occurrences of a word."

Code:

```
const text = "red blue red red red green";
const word = "red";
numberOfOccurrencesInText(word, text);
Expected Output: 4
```

Test: "It should return a word match regardless of case."

Code:

```
const text = "red RED Red green Green GREEN";
const word = "Red";
numberOfOccurrencesInText(word, text);
Expected Output: 3
```

Test: "It should return a word match regardless of punctuation."

Code:

```
const text = "Red! Red. I like red, green, and yellow.";
const word = "Red";
numberOfOccurrencesInText(word, text);
Expected Output: 3
```

Test: "If an empty string is passed in as a word, it should return 0."

Code:

```
const word = "";
const text = "red RED Red!";
```

```
numberOfOccurrencesInText(word, text);
```

Expected Output: 0

[Previous \(/introduction-to-programming/arrays-and-looping/separation-of-logic-adding-a-ui-to-text-analyzer\)](#)

[Next \(/introduction-to-programming/arrays-and-looping/separation-of-concerns-in-text-analyzer-boldpassage-ui-function\)](#)

Lesson 29 of 50

Last updated February 28, 2023

[disable dark mode](#)



© 2023 Epicodus (<http://www.epicodus.com/>), Inc.