MIS 304: Using and Managing Information Systems

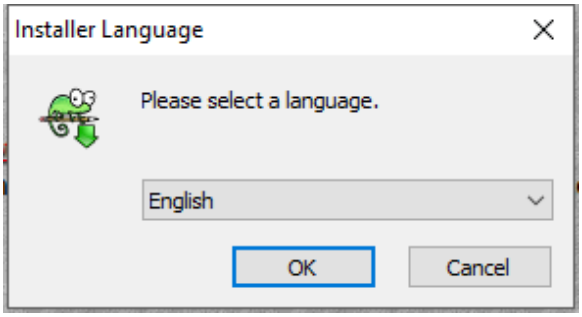
Lab Session 1: Web Programming - Animates Your Name

The goal of this lab session is to help you get started with web programming. We will write a program that animates your name. When you move your mouse over your name, bubbles will scatter away and then reassemble.

We will first learn JavaScript, a programming language. Then we'll apply what we've learned to write this program.

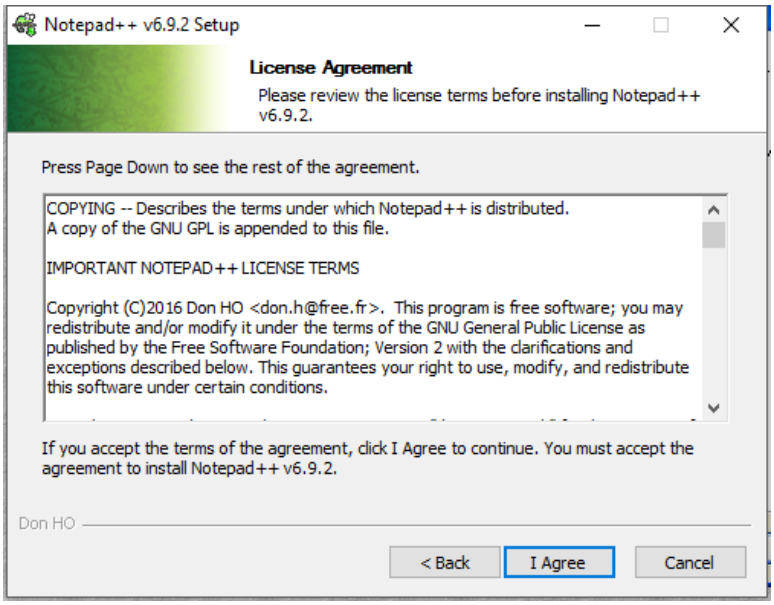
1. We will need a text editor. Notepad++ is a free text editor running in the Windows environment.
   (1) Download the latest version: https://notepad-plus-plus.org/

   (2) Double-click on the file that was downloaded above. You should see “Installer Language”. You may have a question asking you if you really want to run the file. Click “OK”.

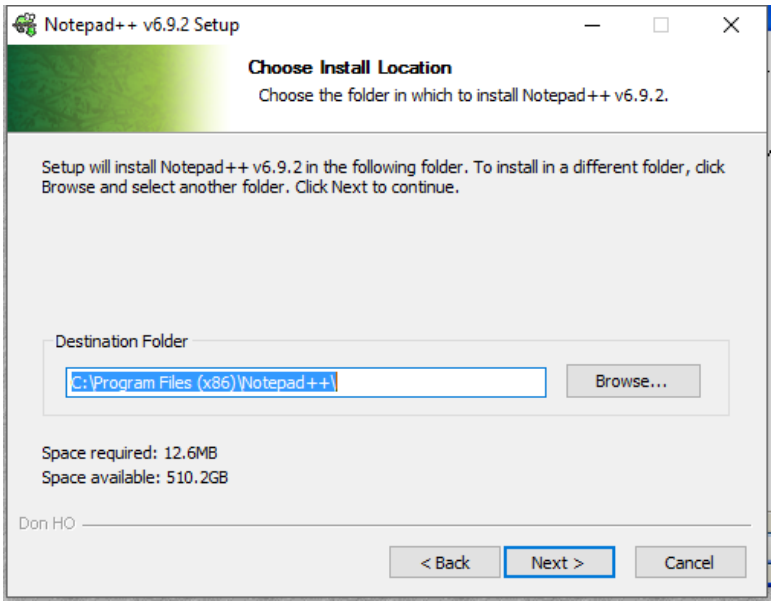


   (3) You're ready to install. Click “Next”.

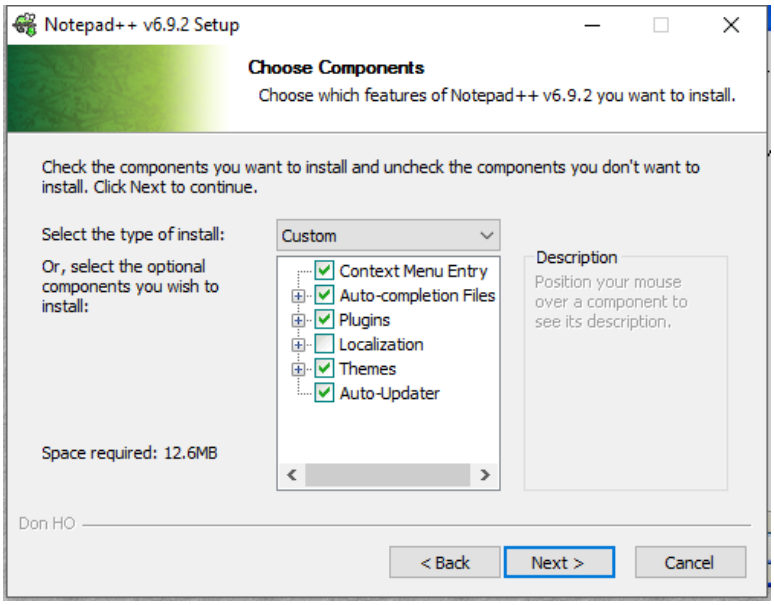
(4) Click "I Agree".



(5) Change where you install Notepad++ if you want. Click "Next" to continue.

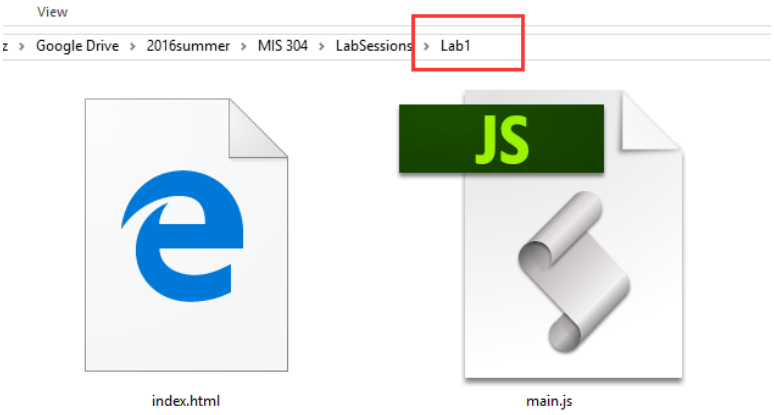


(6) The default setting is sufficient. Click "Next".



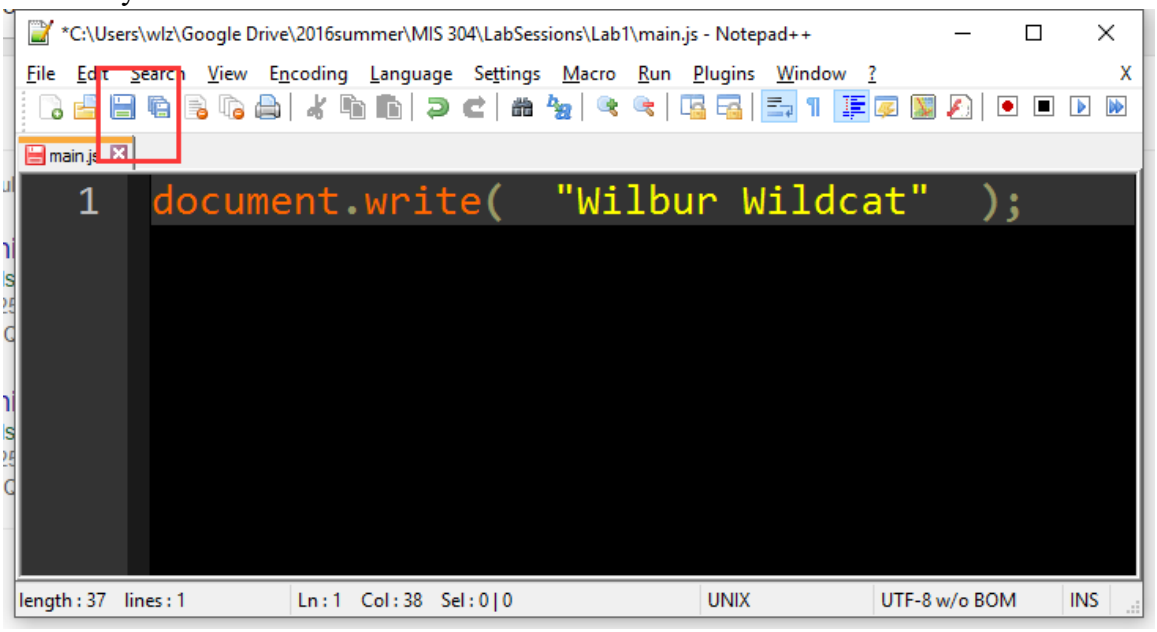
(7) Run Notepad++.

2. Create a file folder "Lab1". Download and save **index.html** and **main.js** in this folder from Blackboard.

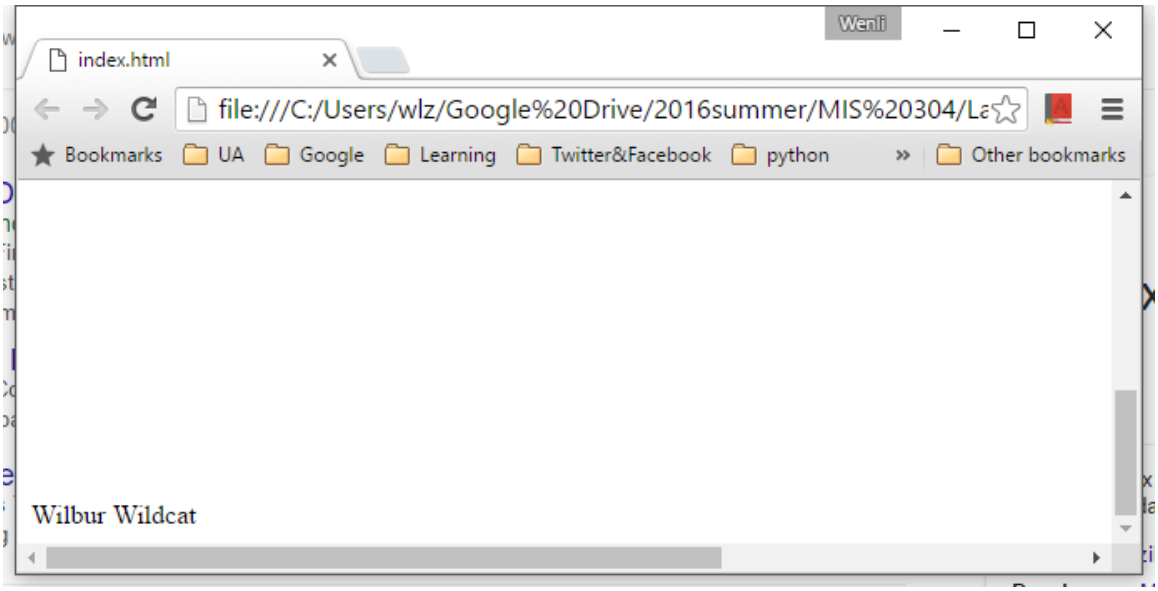


3. **Right** click on **main.js**, click "Edit with Notepad++"
4. In order to write a program that animates your name, we need to learn the programming language JavaScript. Let's get started by getting to know each other. What's your name?

   Write this line in **main.js** document.write("Wilbur Wildcat ");
   Save your file.

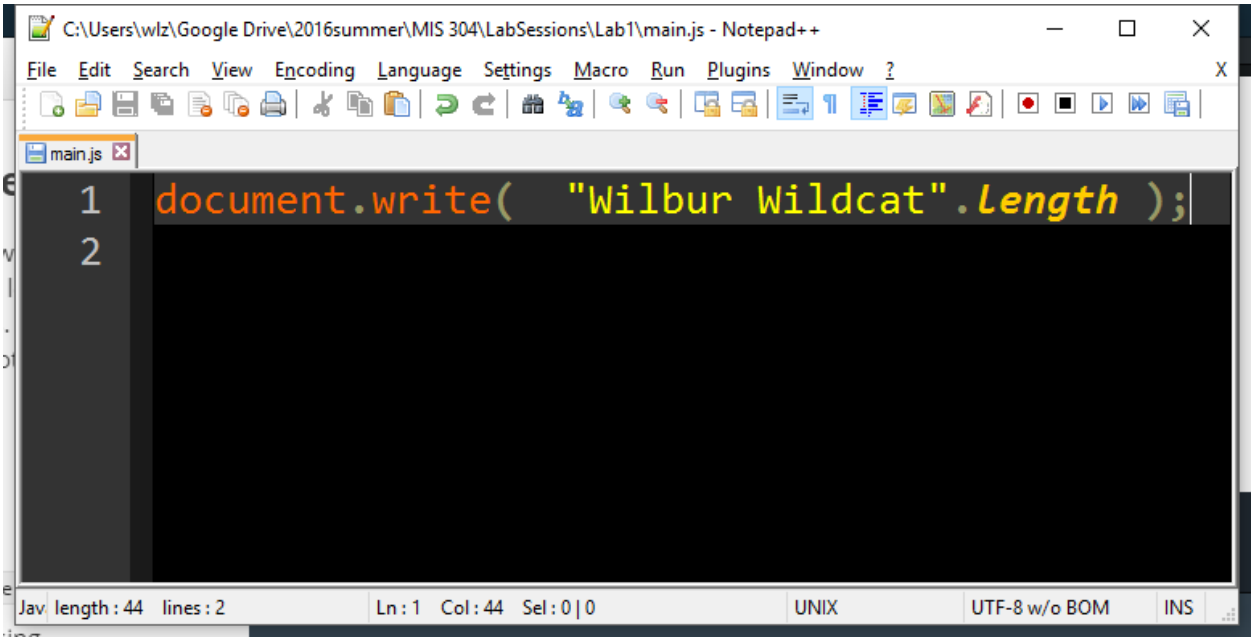


5. Go back to folder "Lab1". Right click on **index.html**, click "Open with…". Choose Google Chrome or Firefox. Do you see your name?

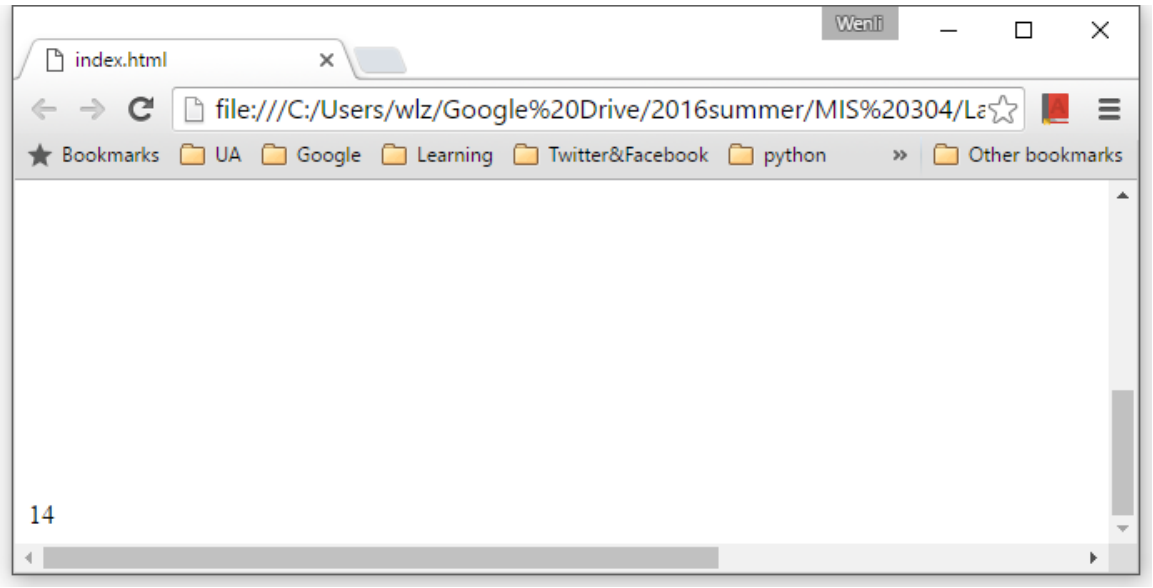


6. You just wrote a **string**.

A **string** can contain letters, numbers, spaces, and symbols. Strings are surrounded with quotes. These are all strings: "Wilbur Wildcat", "4". "What is your name?"
In our code, we're using document.write() simply to display the string of your name in the preview window. The important stuff is inside the parentheses, so let's just focus on that.
To discover the length of a string, write the string within quotes. Then write a period (full stop) and the word **length** like this: document.write("Wilbur Wildcat".length);
Don't forget save your file.



Reload **index.html**. What is the length of your name?

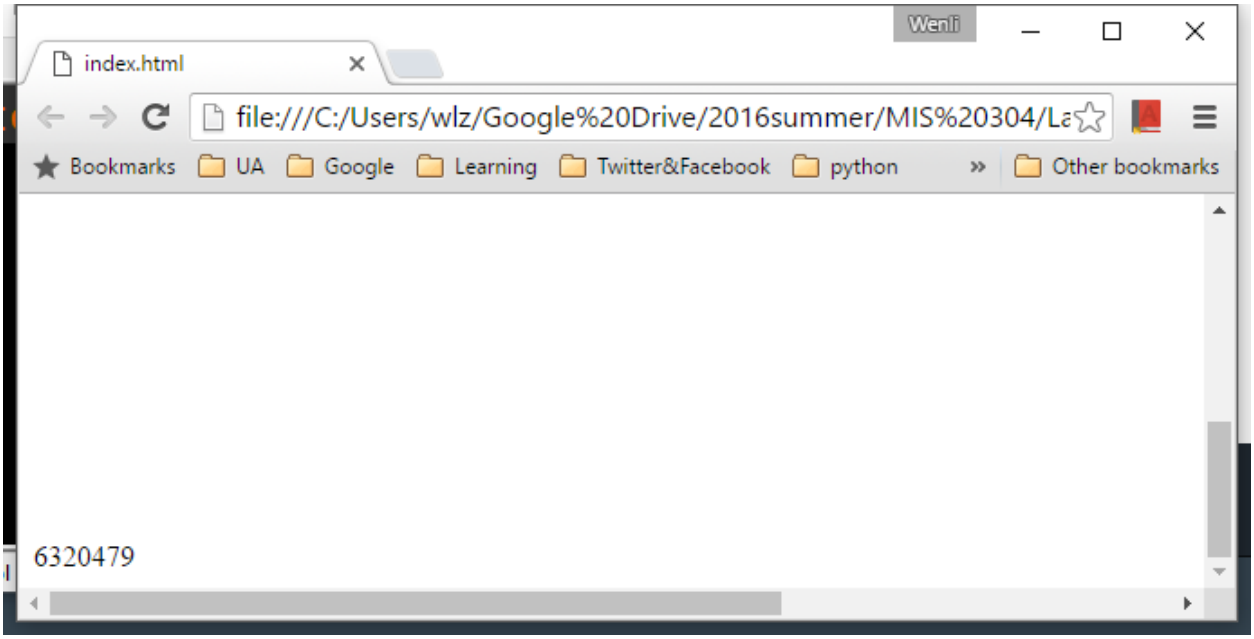


7. Now, let's do some math. You can add, subtract, multiply, and divide numbers in JavaScript, like this:

   Addition: 2 + 3 Subtraction: 6 – 3 Multiplication: 3 * 4 Division: 10 / 5

   Multiply 2 really big numbers! Write your expression between the parentheses, like this: document.write(1111 * 5689);

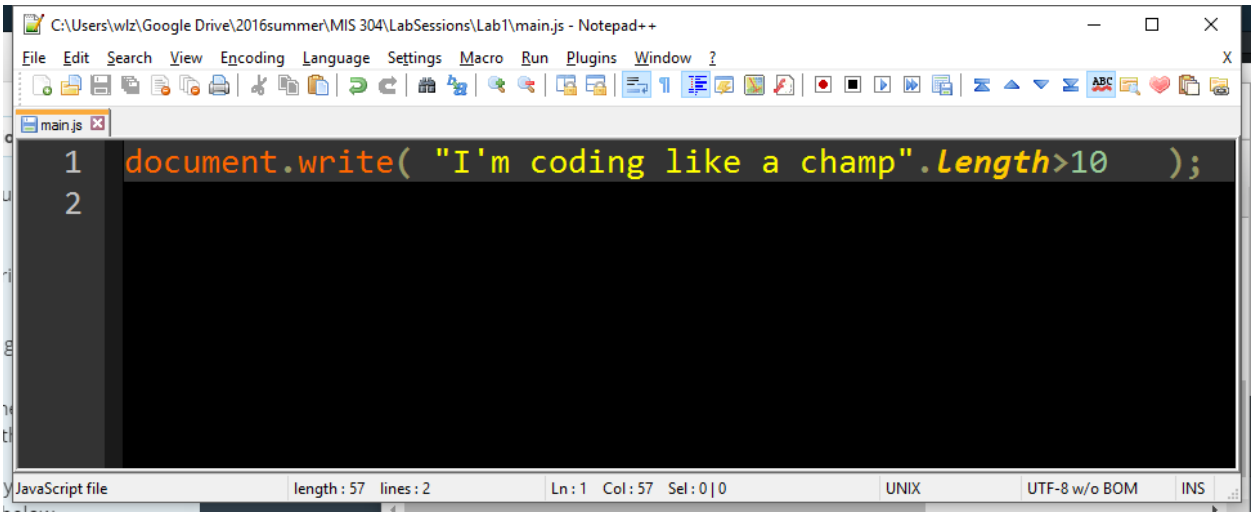
   What is the result?

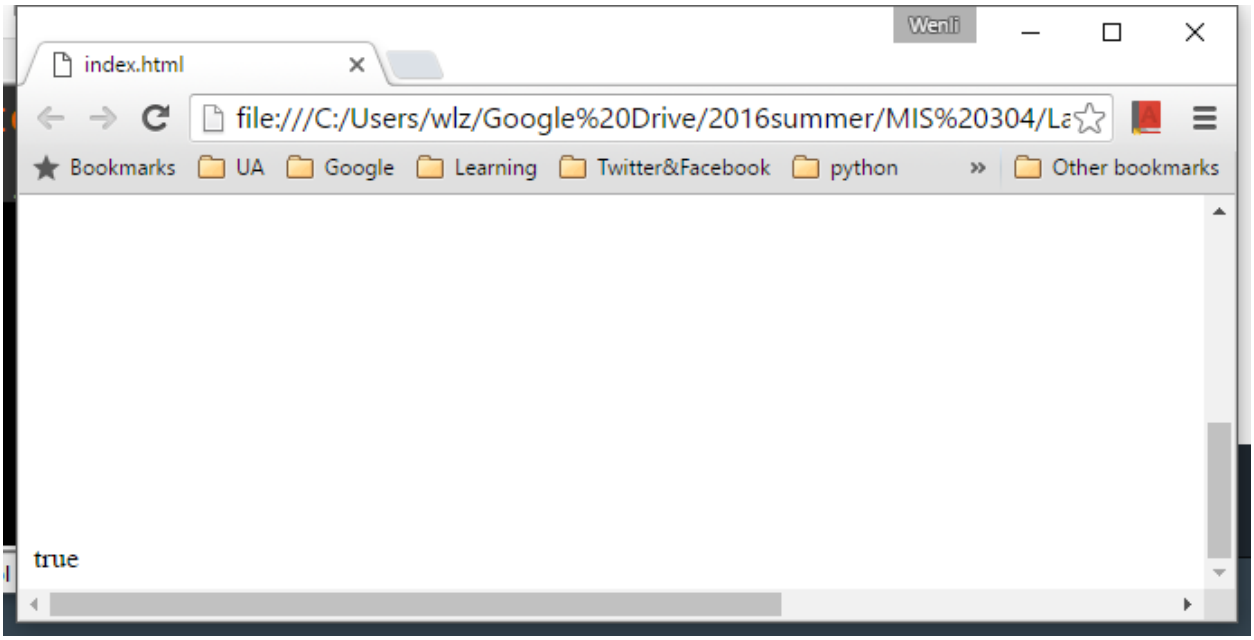


8. Next let's look at Boolean expressions. A Boolean expression is either true or false. For example, comparing two numbers returns a true or false result:

   23 > 10 is true 5 < 4 is false

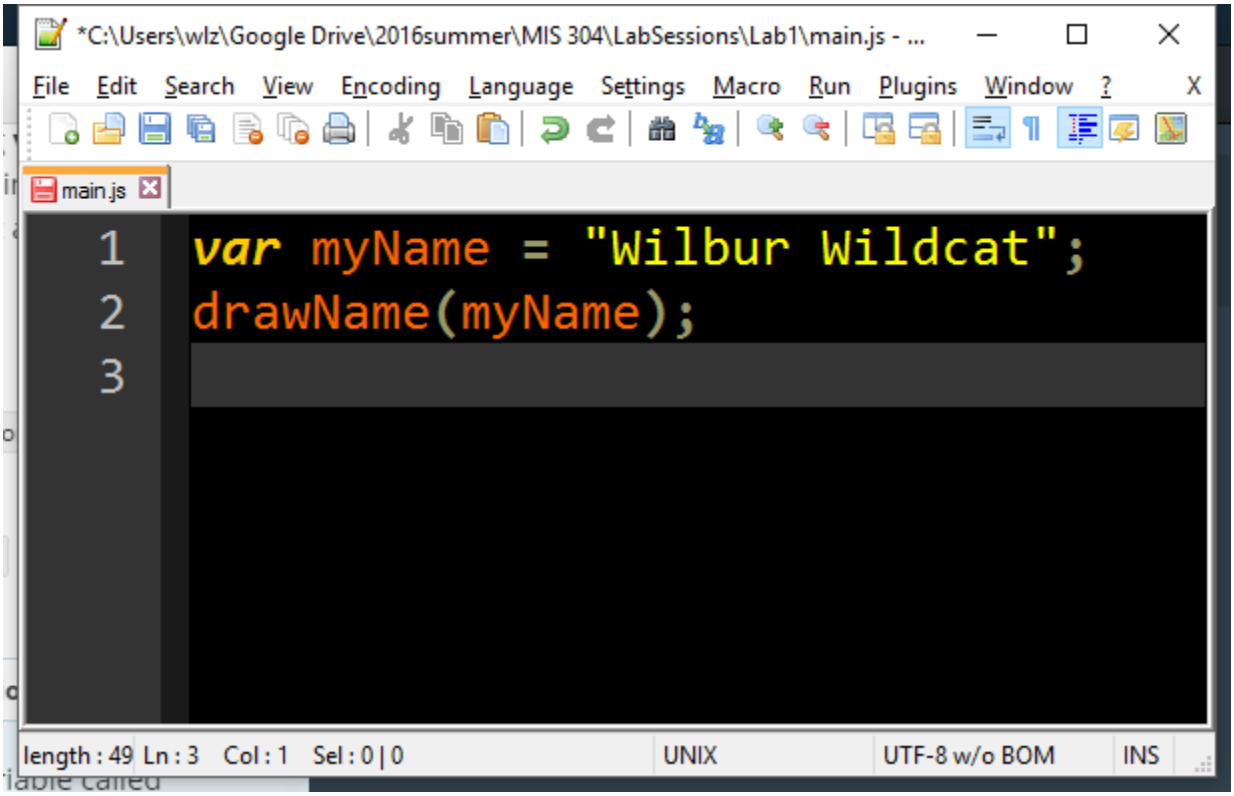
Let's compare two numbers to return a true result: First, write the string "I'm coding like a champ". Next, find the length of the string. Then, compare the string's length to see if it is greater than 10. What is the result?





9. So far, we've been typing in **strings**, **numbers**, and **booleans** into the editor. To do more complex coding, we need a way to "save" these values. We can do this using **variables**. A **variable** stores a string, number, or boolean, and gives it a specific, case-sensitive name. Examples: var myName = "Wilbur Wildcat "; var myAge = 5; var isEven = true

   Create a variable called myName and type in your name. Draw your name by using drawName(myName);

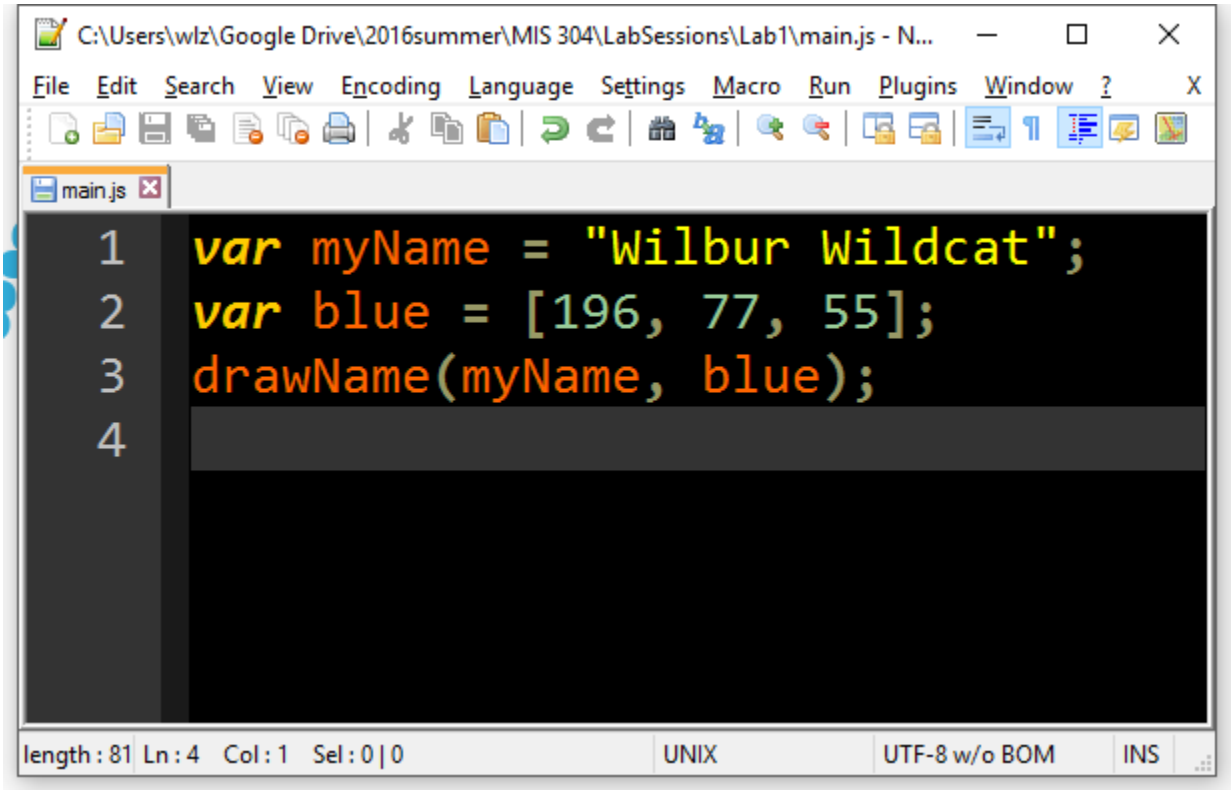


10. Your name is drawn as a collection of bubbles. How did this happen?

In line 1, you created a variable named myName in which you stored a string of your name. In line 2, a function named drawName() took your name string and drew it on the screen. What's a function? A function takes in an input, does something with it, and then returns an

output. In our code, the input was your name, and the output was the picture of your name as a bunch of bubbles.

Let's give the bubbles some color. Create a variable called blue and type in [196, 77, 55]. Add blue as another input to the drawName function, like this: drawName(myName, blue);. See what output you get.

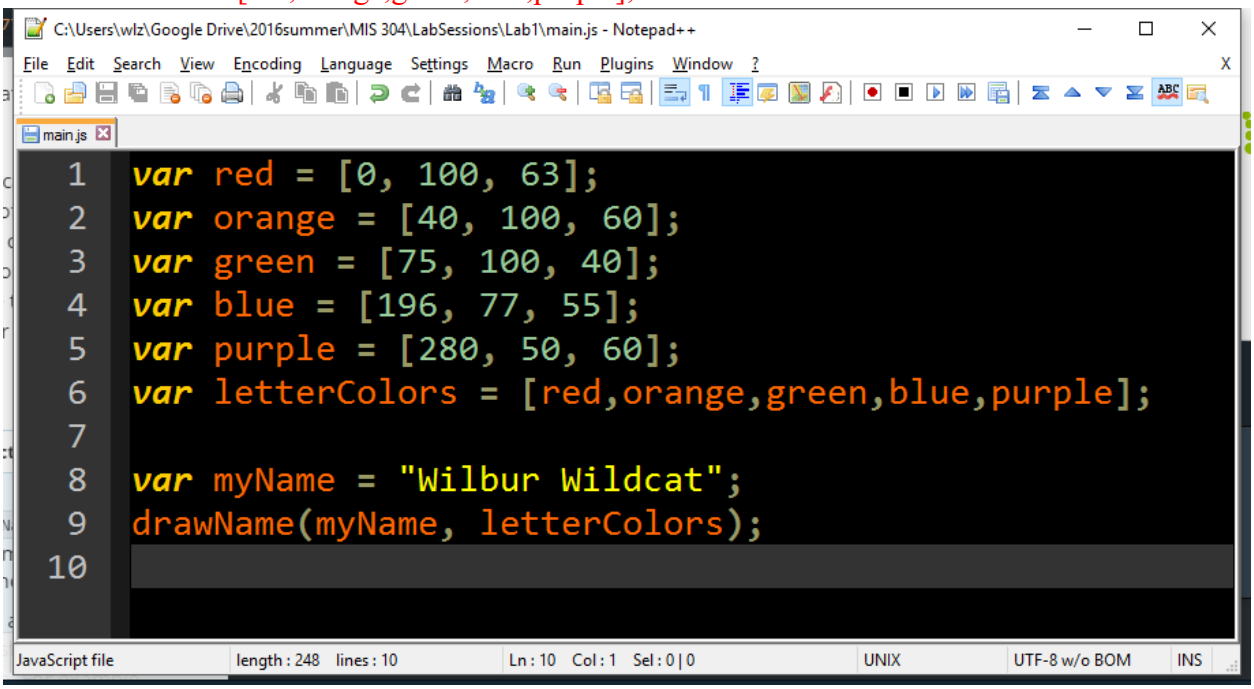


What is [196, 77, 55]? (Read: https://en.wikipedia.org/wiki/RGB_color_model)

11. Your name is now drawn as a collection of blue bubbles. But wouldn't it be cooler if we could use more than one color? Variables, like myName, can store **numbers** or **strings**. But so far, we've only been able to store one number or one string at a time. Good thing we have **arrays**. Arrays store lists of data. Anytime you see data surrounded by [], it is an array.

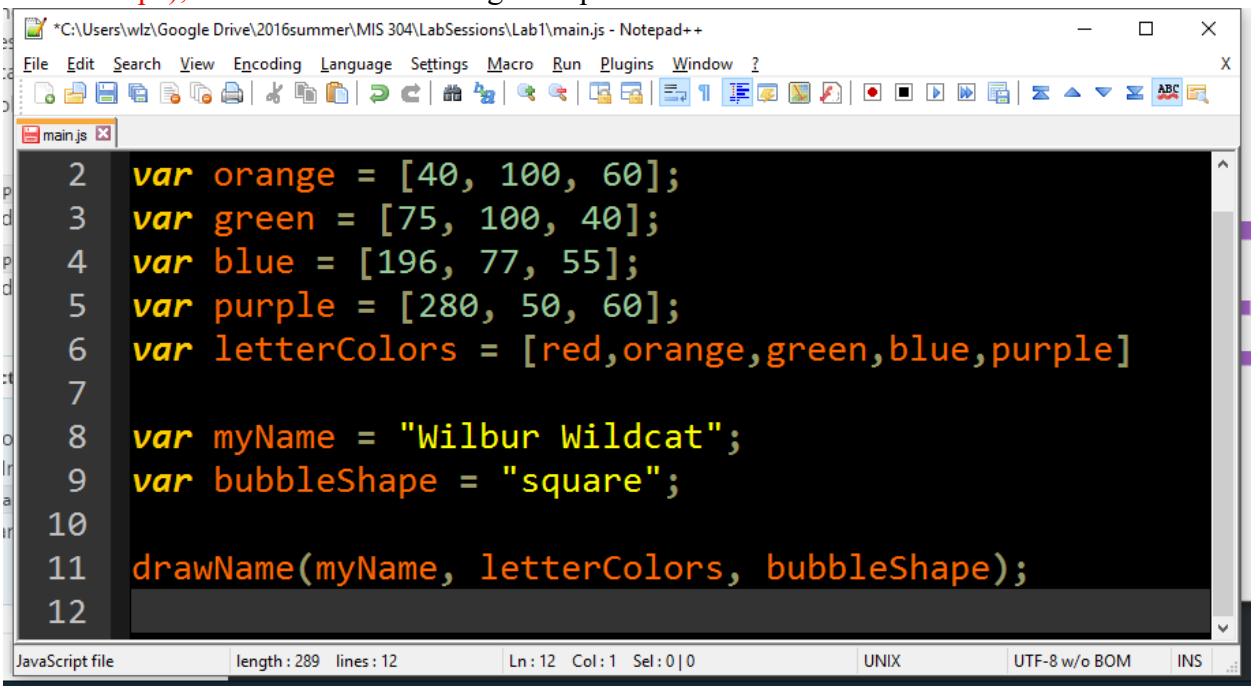
In fact, computers can understand colors as an array of numbers. Let us add five colors. They are simply variables storing arrays of 3 numbers. Let's use them now to make the bubbles in your name more colorful, use drawName(myName, letterColors); .

```
var red = [0, 100, 63];
var orange = [40, 100, 60];
var green = [75, 100, 40];
var blue = [196, 77, 55];
var purple = [280, 50, 60];
var letterColors = [red,orange,green,blue,purple];
```



12. What did we just do? We gave the **function** drawName() twoinputs. One was your name. The other was an array of colors. The output was a multi-colored picture of your name.

Now let's change the shape of the bubbles from circles to squares. There's a variable called bubbleShape that lets you control the shape of the bubbles.
When bubbleShape = "square", the bubbles are shaped as squares.
When bubbleShape = "circle", the bubbles are shaped as circles.
Now let's add var bubbleShape = "square";. Use drawName(myName, letterColors, bubbleShape);. The bubbles will change to squares.

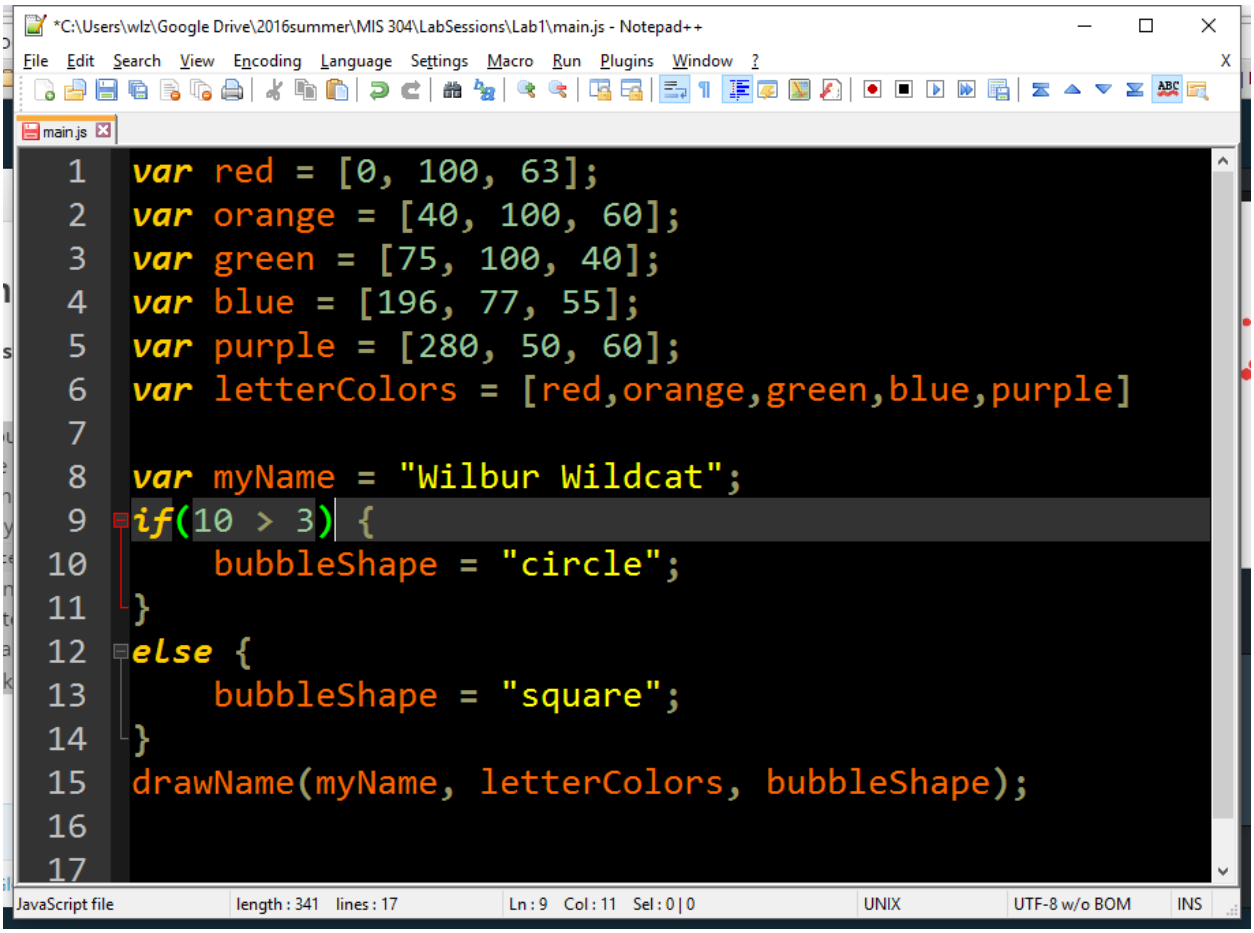


13. Next let's use booleans to decide whether a block of code should run. Take a look at this code:

```
if(10 < 3) {
   bubbleShape = "circle";
}
else {
   bubbleShape = "square";
}
```

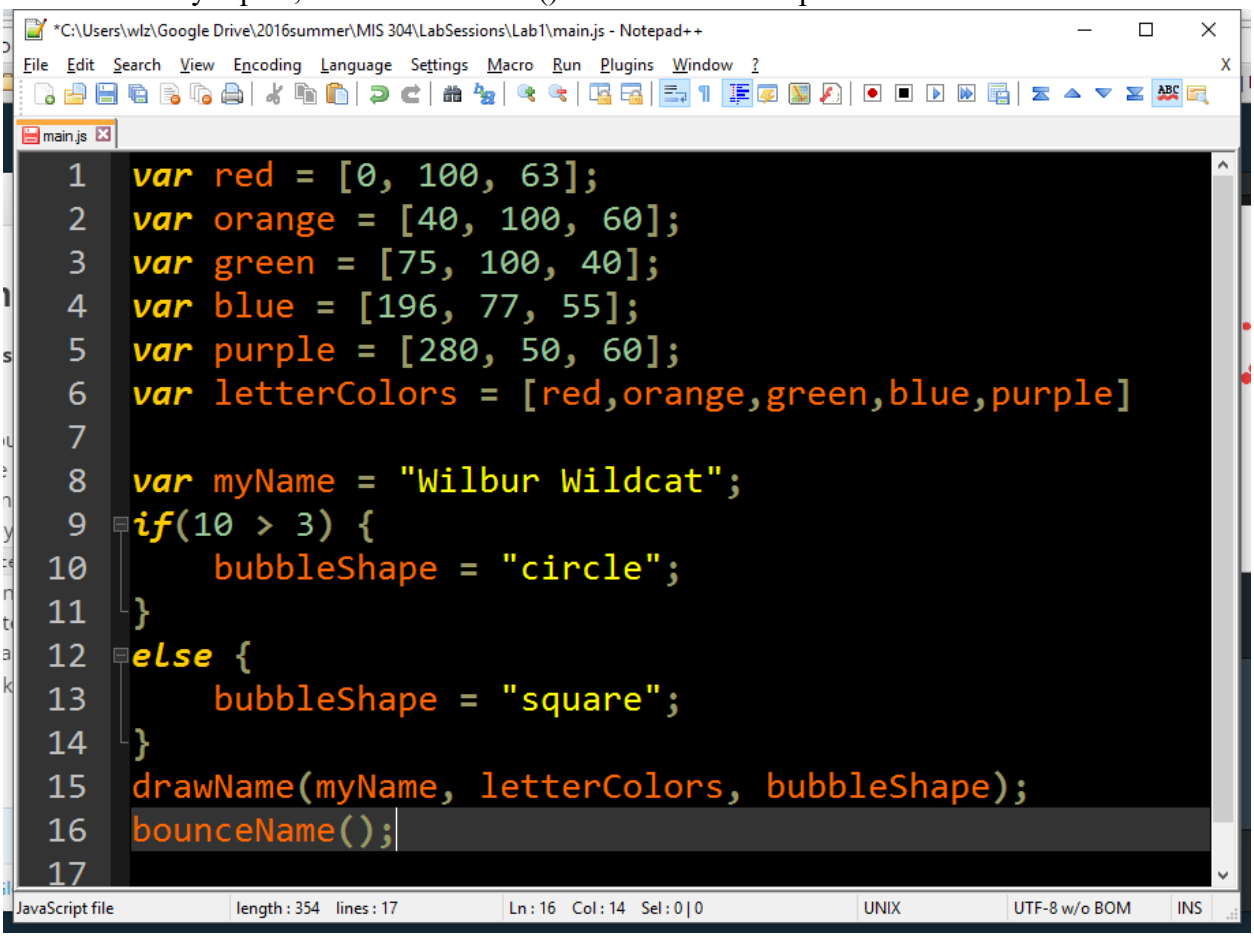
If the condition (in this case, 10 < 3) evaluates to true, then it runs the code inside the first pair of curly braces {}, which will make the bubbles circle-shaped. ***Else*** the condition is false (which it is, in this case), so it skips the code in the first block of curly braces entirely, and runs the code in the second block. Therefore, the bubbles will be square-shaped. This is called an if/else statement.

Delete bubbleShape = "square" of your code. Replace it with an if/else statement that makes the bubbles circle-shaped. Use the **if/else statement** in the example above to get started.



14. Let's make the bubbles in your name more interactive. It would be cool if you could mouse over your name and cause it to move in some way. We have a **function** called bounceName()

that shakes your name around when a mouse comes close to it. Note that bounceName() does not take any inputs, unlike drawName() which took two inputs.



15. Your name rattles around when you move your mouse near it. Now as cool as it is to see your name shake about, it would be even cooler to bounce the bubbles themselves. We have another function called bounceBubbles() that does this. Similar to bounceName(), this function does not take any inputs. Replace bounceName() with bounceBubbles().

*C:\Users\wlz\Google Drive\2016summer\MIS 304\LabSessions\Lab1\main.js - Notepad++

File Edit Search View Encoding Language Settings Macro Run Plugins Window ?

main.js

```
1  var red = [0, 100, 63];
2  var orange = [40, 100, 60];
3  var green = [75, 100, 40];
4  var blue = [196, 77, 55];
5  var purple = [280, 50, 60];
6  var letterColors = [red,orange,green,blue,purple]
7
8  var myName = "Wilbur Wildcat";
9  if(10 > 3) {
10     bubbleShape = "circle";
11 }
12 else {
13     bubbleShape = "square";
14 }
15 drawName(myName, letterColors, bubbleShape);
16 bounceBubbles();
17
```

JavaScript file | length : 357 lines : 17 | Ln : 16 Col : 17 Sel : 0 | 0 | UNIX | UTF-8 w/o BOM | INS

Deliverables

Take a screen shot of your final result and paste it on a word file yourNetID_L1.docx, submit in TurnItIn.