Attendance

Please login to Canvas and “Check-In”

Attendance is not a part of your grade for this class. The university has mandated that we take attendance for all classes, face-to-face, online and hybrid, to assist in contact tracing should an outbreak of Covid-19 occur.

FOX
MIS

TEMPLE UNIVERSITY

Information Systems in Organizations

An Introduction to Programming

FOX MIS

ROADMAP

START

Week 1:
Introduction & Systems Analysis
- Course Description
- Systems Thinking

Week 1:
Introduction to Process Mapping
- Systems & Processes
- Swim Lane Diagrams

- Max Labs 0- due
- Practice test - due

Week 2:
Digital Product Management & ERD

Week 2:
Introduction to Data Modeling
- Max Labs 1A/1B- due
- Max Labs 2A/2B due

Week 2:

Exam #1

10/30 – 11/1: Exam Availability

Week 3:
Information Systems
- ERP & CRM
- Data Analytics & SCM

- Lean IT #1 due

Week 4:
Platforms & Digital Business Models
- API's
- Cloud

Week 4:
Cybersecurity & AI
- Protection Protocols
- Artificial Intelligence

- Cybersecurity/AI assignment due
- Max Labs 3a/3b due

Week 4 :

Exam #2

11/13-11/5 Exam Availability

Week 5:
JavaScript Unit #1 & 2
- Hello World, Variables
- Input and Output
- Operator types
- Strings

Watch Lynda.com video – due
Code Academy due

Week 6:
JavaScript Unit #3&4
- Logical Operators
- Conditional Types
- Intro to Loops
- While and Do
- Writing the code
- Practice Coding Exam

Week 7:
HTML & CSS

- Coding Assignment -due
- Lean IT #2 due

Week 7:

Exam #3

12/8 – 12/ 9: Exam Availability

FINISH

An Introduction to Programming

Week 5

FOX MIS

Machines

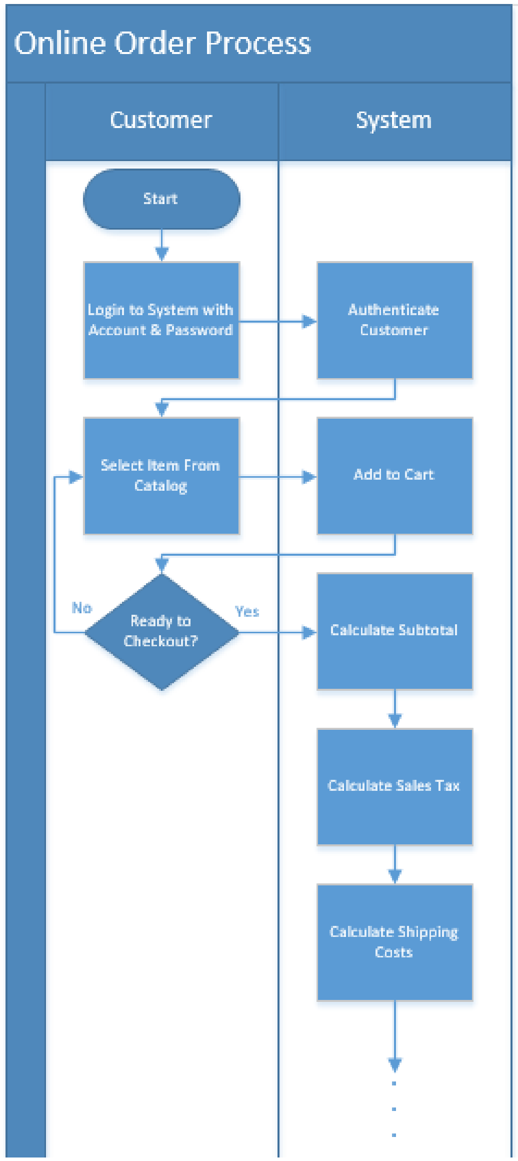
How many different things does a machine do?

Source: https://lh3.googleusercontent.com/_MsstYyQ4UORpFRdwNu6BN2_cL3vu_raiYq-RErSsdf_Y6v_lKZss8N-8qCO-Tnf3aalDw=s170
Source: https://lh3.googleusercontent.com/GB8PJSrB61OSat3rBAJO-wEjTGstYfB2N4ECL2LSYOahESElAVf7LWJMNrJ0undIP10o=s119
Source: https://lh3.googleusercontent.com/VVZxPvd2BpOPvECeBytaSZqFUfAg5UqPmeroft5ONc3bQVKYqr2s5IxoI4NvEP4unuJiFDI=s94
Source: https://lh3.googleusercontent.com/eZGdlrLtdzW9nEKR758LiagZ3_-wjxflYbpU-IfZeVdny34bXSgn4s0DQTsfS3t_O2eZ_g=s85
Source:
https://lh3.googleusercontent.com/OCVAphS1iC9nYvF2FeP_lINXwWQ69mCX8HjK2DJ7UU_xGnoHOgXd4YX8O4Zes09WIcIFxP8=s85
Source: https://lh3.googleusercontent.com/mCtpPS6Qz3guK0-cyCruwUwgbKYAJhWH1iTqGbCNjMOvSnGMvL1FeBGeDIBYboOC7P-gfQ=s102

Machines (cont.)

How many different things does a computer do?

How can this machine do so many different things?



Source: https://lh3.googleusercontent.com/59wm5Z8eXC40_WKib3Ghe3ORRuSITa17Xl5BN8QIikEh_cDnScrGnOouZnO4X-cBY6ad5w=s128

Source: https://lh3.googleusercontent.com/luQcPRIrni_WsHytr016U1SPu7FvyVlK2AfyBH3fXEjhoBPrmpzuTfgjvQV40UR63ZTq1jk=s134

Source: https://lh3.googleusercontent.com/Mkj8bl6o0XbrVqnFs40Y3AFVMibSJX1xra-Ll5rwbMCxGs67gpljSny2TUI4KASK1u8sD-M=s85

Source: https://lh3.googleusercontent.com/dMhdY1wyo91ER5uMkI1recx4-0w7fOjNUNmSB9ZCgVnIMqqqXNDkxiw9ckuWij_vtzFSEw=s128

Source: https://lh3.googleusercontent.com/qu1kdMmaM8SbvpyiDQEULPMPGfCrHneGT7WZ494FhgM8N4SHp-TOQvmuKBK2leESFBSPUjg=s151

Software
Programs = Software = Apps

But aren’t all programmers geeks?

In the digital economy, geeks are…

Problem Solvers

Designers

Creators

Enablers

Source: https://lh3.googleusercontent.com/1w8YU_tUBElGdnnz7bm6J11P0r1ULETREi4a66qF_kQCwwMF4VRdytLg_iu2sXcFSeru=s128

Source: https://lh3.googleusercontent.com/T1uuAU2Ov5B0oYU4KCifPlOaYu8cZCqbUBwF_4KXAHzvbU_VKp0ymBGetZ1pBUL0Q73oL4o=s159

Evolution of the Digital Product Manager…

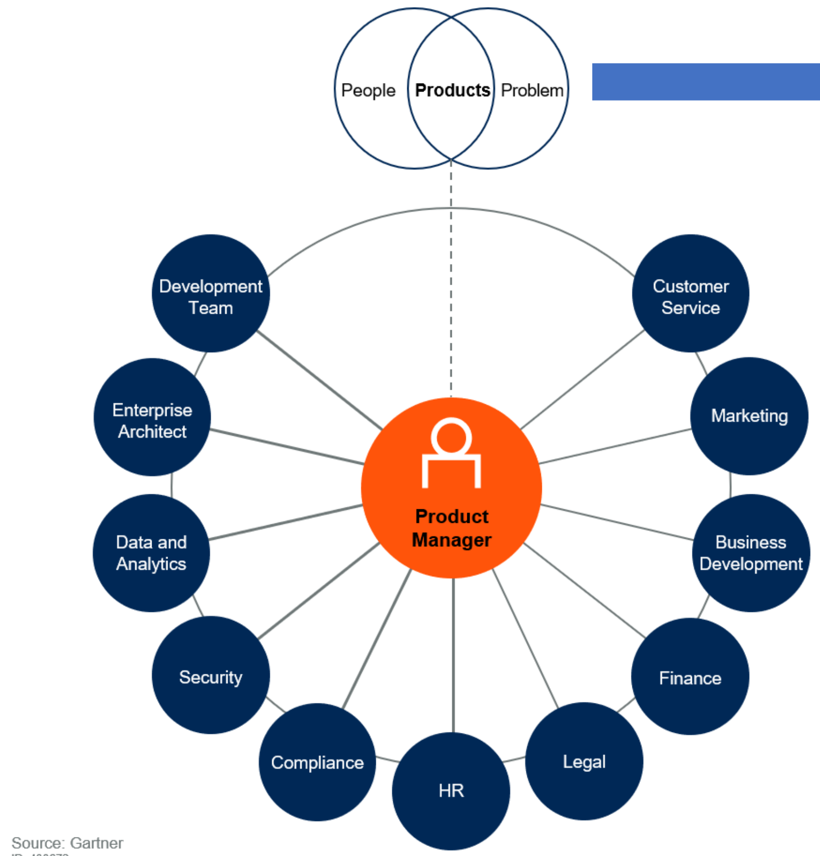
Problem Solvers

Designers

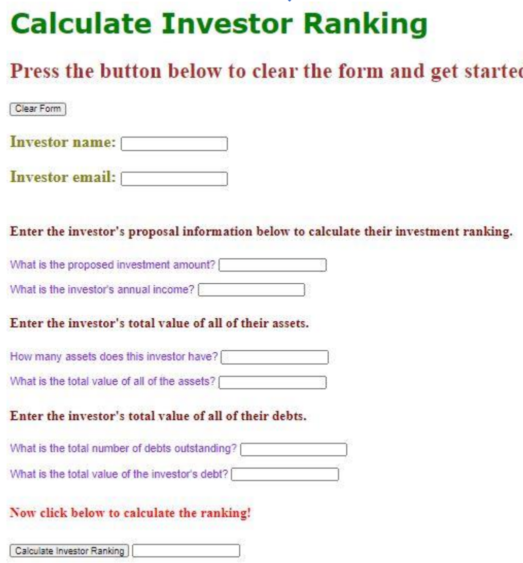
Creators

Enablers

The Product Manager's Work Circle



Source: Gartner
ID: 400672


Calculate Investor Ranking

Press the button below to clear the form and get started!

Clear Form

Investor name:

Investor email:

Enter the investor's proposal information below to calculate their investment ranking.

What is the proposed investment amount?

What is the investor's annual income?

Enter the investor's total value of all of their assets.

How many assets does this investor have?

What is the total value of all of the assets?

Enter the investor's total value of all of their debts.

What is the total number of debts outstanding?

What is the total value of the investor's debt?

Now click below to calculate the ranking!

Calculate Investor Ranking

What exactly is a program?

```
a.fn.scrollspy=d,this},a(window).on(
y),+function(a){"use strict";function b(b){return this.each(function(){var
e[b]()})}var c=function(b){this.element=a(b)};c.VERSION="3.3.7",c.TRANSITION_DURATION=150,c.prot
opdown-menu)"),d=b.data("target");if(d||(d=b.attr("href"),d=d&&d.replace(/.*(?=#[^\s]*$)/,"")),
st a"),f=a.Event("hide.bs.tab",{relatedTarget:b[0]}),g=a.Event("show.bs.tab",{relatedTarget:e[0]
faultPrevented()){var h=a(d);this.activate(b.closest("li"),c),this.activate(h,h.parent(),functio
rigger({type:"shown.bs.tab",relatedTarget:e[0]})})}}},c.prototype.activate=function(b,d,e){func
> .active").removeClass("active").end().find('[data-toggle="tab"]').attr("aria-expanded",!1),
ia-expanded",!0),h?(b[0].offsetWidth,b.addClass("in")):b.removeClass("fade"),b.parent(".dropdow
().find('[data-toggle="tab"]').attr("aria-expanded",!0),e&&e()}var g=d.find("> .active"),h=e&&
e")||!!d.find("> .fade").length);g.length&&h?g.one("bsTransitionEnd",f).emulateTransitionEnd
;var d=a.fn.tab;a.fn.tab=b,a.fn.tab.Constructor=c,a.fn.tab.noConflict=function(){return a.fn.t
"show")};a(document).on("click.bs.tab.data-api",'[data-toggle="tab"]',e).on("click.bs.tab.data
se strict";function b(b){return this.each(function(){var d=a(this),e=d.data("bs.affix"),f="ob
typeof b&&e[b]()})}var c=function(b,d){this.options=a.extend({},c.DEFAULTS,d),this.$target=a
",a.proxy(this.checkPosition,this)).on("click.bs.affix.data-api",a.proxy(this.checkPositionWi
null,this.pinnedOffset=null,this.checkPosition()};c.VERSION="3.3.7",c.RESET="affix affix-top
State=function(a,b,c,d){var e=this.$target.scrollTop(),f=this.$element.offset(),g=this.$targ
"bottom"==this.affixed)return null!=c?!(e+this.unpin<=f.top)&&"bottom":!(e+g<=a-d)&&"bottom"
!=c&&e<=c?"top":null!=d&&i+j>=a-d&&"bottom"},c.prototype.getPinnedOffset=function(){if(this
.RESET).addClass("affix");var a=this.$target.scrollTop(),b=this.$element.offset();return
WithEventLoop=function(){setTimeout(a.proxy(this.checkPosition,this),1)}
nt.height(),d=this.options.offset,e=d.top,f=d.botto
peof e&&(e=d.top(this.$element))
nt.css("top","")
```

Source: https://lh3.googleusercontent.com/7cDbXsffRS_f2HRcvXUtmTLZm9EcoMwHGlynMrb7G4bsGcCE_7o3HqToGjynKLosbP2EQ=s114

Source: https://www.wxxi.org/sites/wxxi/files/styles/x_large/public/201802/Algorithm.jpeg

How to learn how to program

- Program
- Program some more
- Program more after that
- Delete everything and program again!

WE'RE GOING TO HAVE TO ASK YOU TO COME IN THIS WEEKEND AND PRACTICE MORE CODING

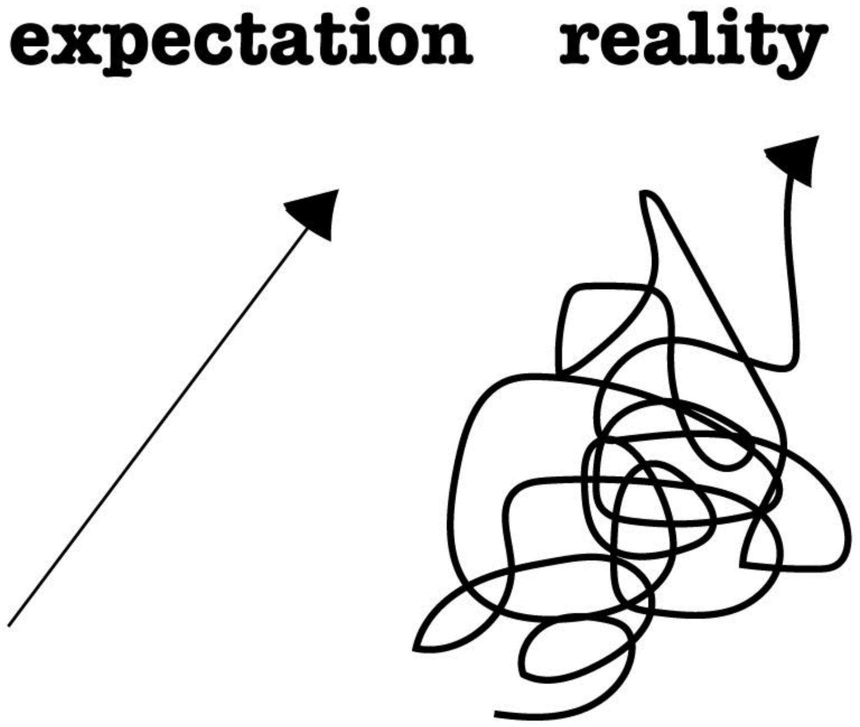
Source: Photofest: https://www.hollywoodreporter.com/review/office-space-review-1999-movie-1086336

“To err is human, but to really foul things up you need a computer.”

- Paul R. Ehrlich

FOX MIS

Managing Expectations



https://www.futuredesigncoaching.com/single-post/2017/12/12/When-Expectation-and-Reality-are-Misaligned

Hello World!

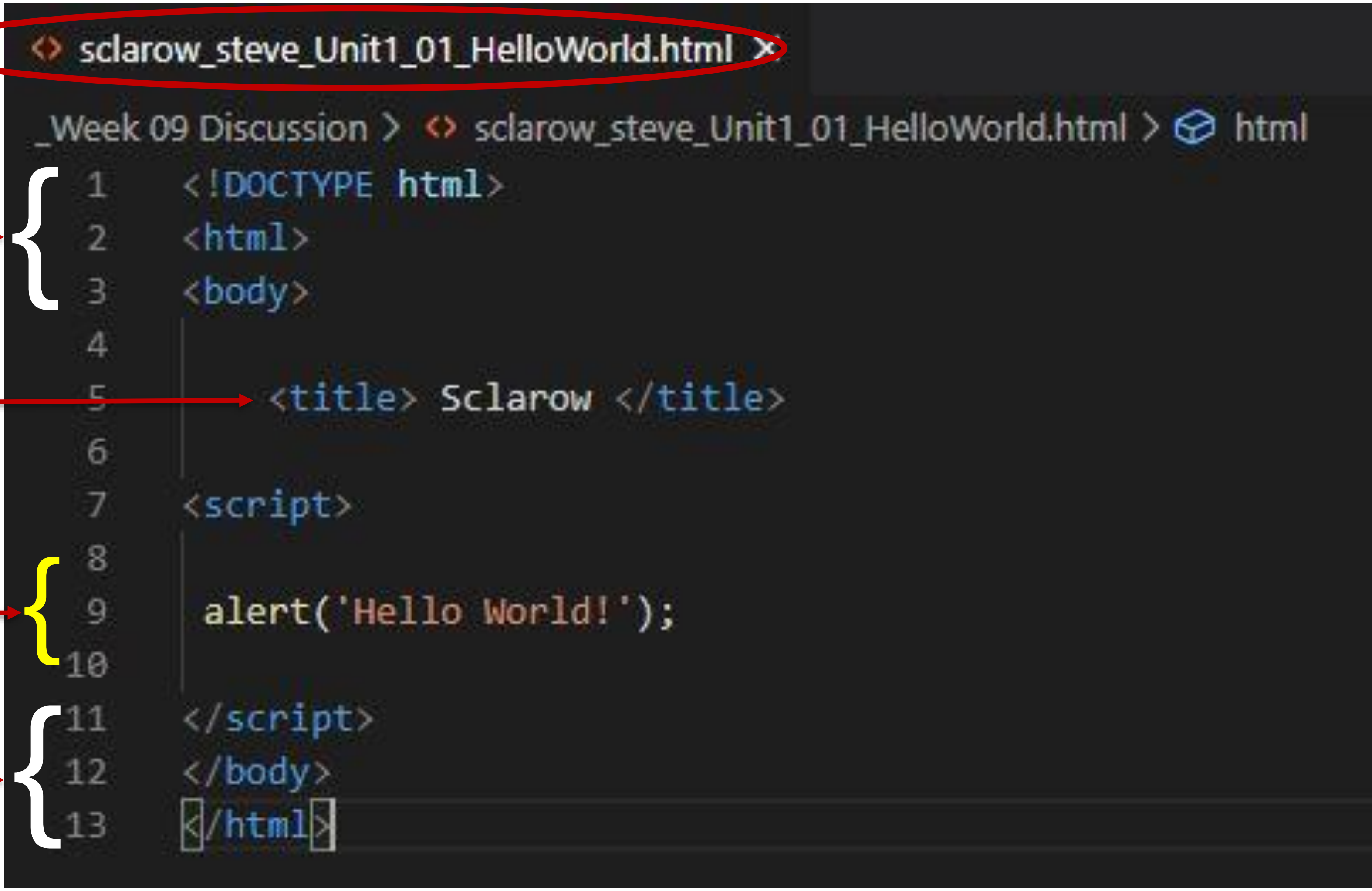
Keep track of where you store your code!

This is the HTML and never changes

Rename your title

This is the JavaScript code

This is the HTML and never changes


sclarow_steve_Unit1_01_HelloWorld.html
_Week 09 Discussion > sclarow_steve_Unit1_01_HelloWorld.html > html

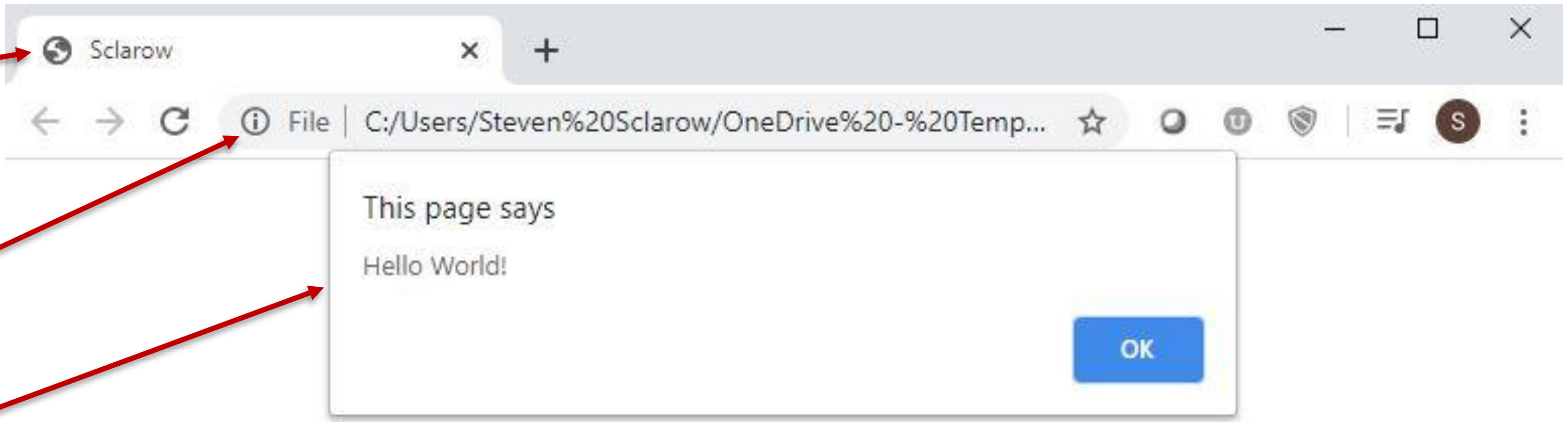
```
<!DOCTYPE html>
<html>
<body>

    <title> Sclarow </title>

<script>

 alert('Hello World!');

</script>
</body>
</html>
```

Hello World!

This is the Title

Keep track of where you store your code!

This is the output from Alert



This page says

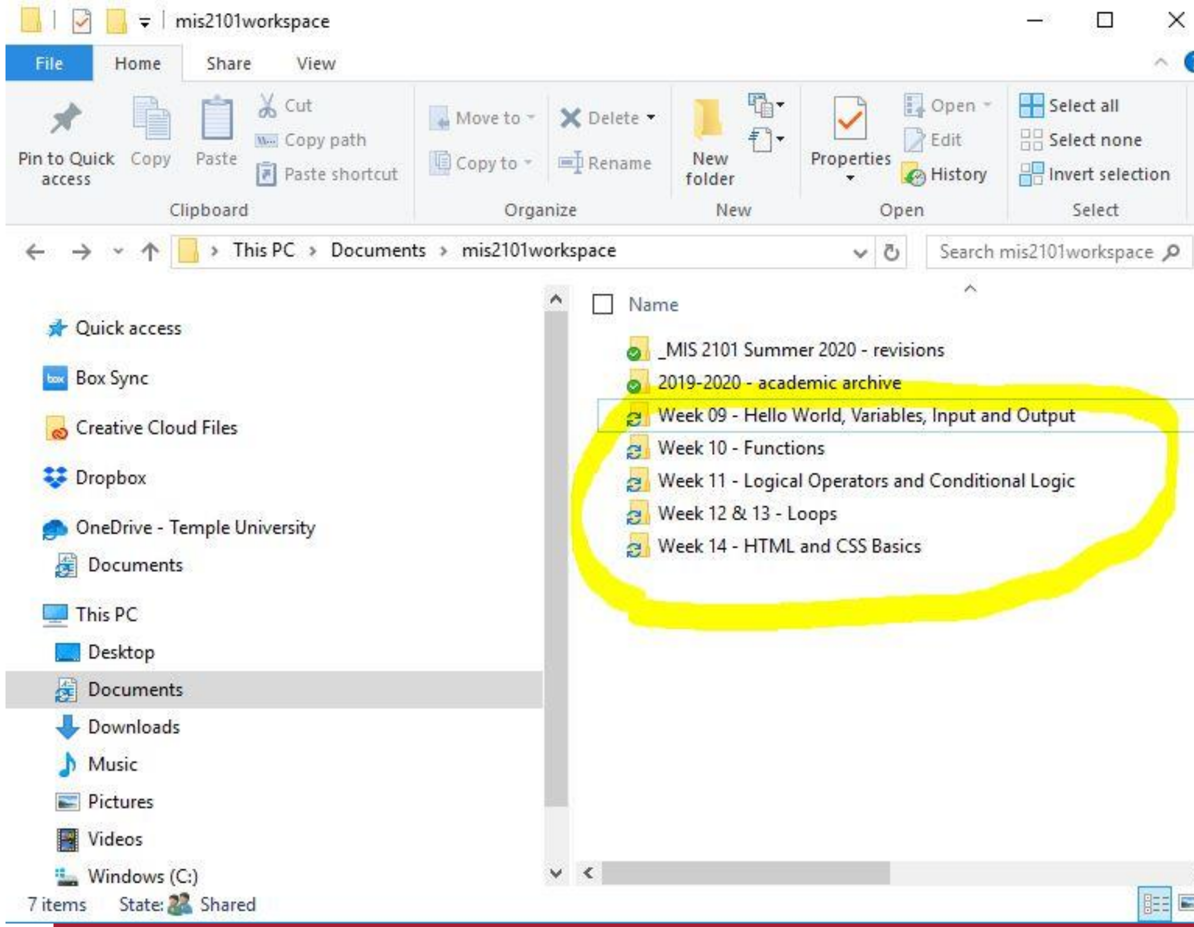
Hello World!

Coding Tools

- To code, we will need a text editor:
  - Installing-VS-Code-Windows
  - Installing-VS-Code-Mac-OS
- We need a browser to view our work.
  - Make CHROME your default browser

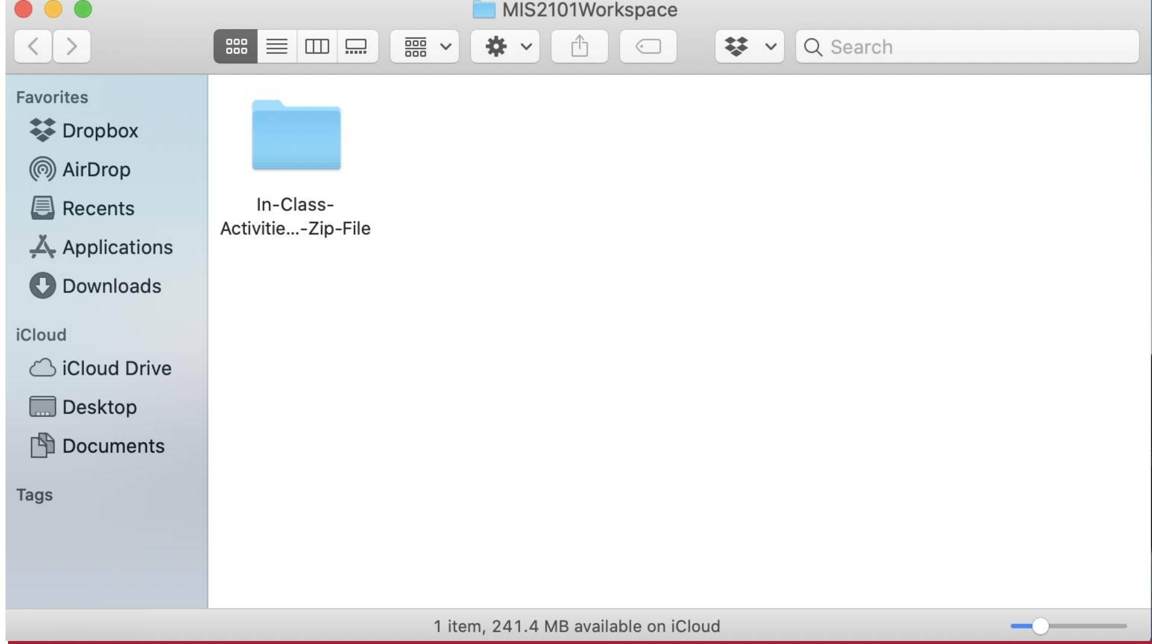
File Download: PC

- Create a folder on your hard drive
  - This is where your will save all of your coding files!
- Visit course site for the coding files
- Download each week's coding files into your new folder
  - You may need to "unzip" the files and extract them into your folder:
    - Mac help [(just google it!)]()
    - PC help [(just google it!)]()



File Download: Mac

- Create a folder on your hard drive
  - This is where your will save all of your coding files!
- Visit course site for the coding files
- Download each week's coding files into your new folder
  - You may need to "unzip" the files and extract them into your folder:
    - Mac help [(just google it!)]
    - PC help [(just google it!)]



Coding Files AKA Starter Files

When you download the coding files, they are embedded in a zip file.

A zip file is essentially a folder with files. The files in the folder have been compressed so that they don't take up much space

You can look at a file inside a zip file, you can for instance open a .html file in Google Chrome.

HOWEVER, you cannot save a file that is stored inside a zip file.

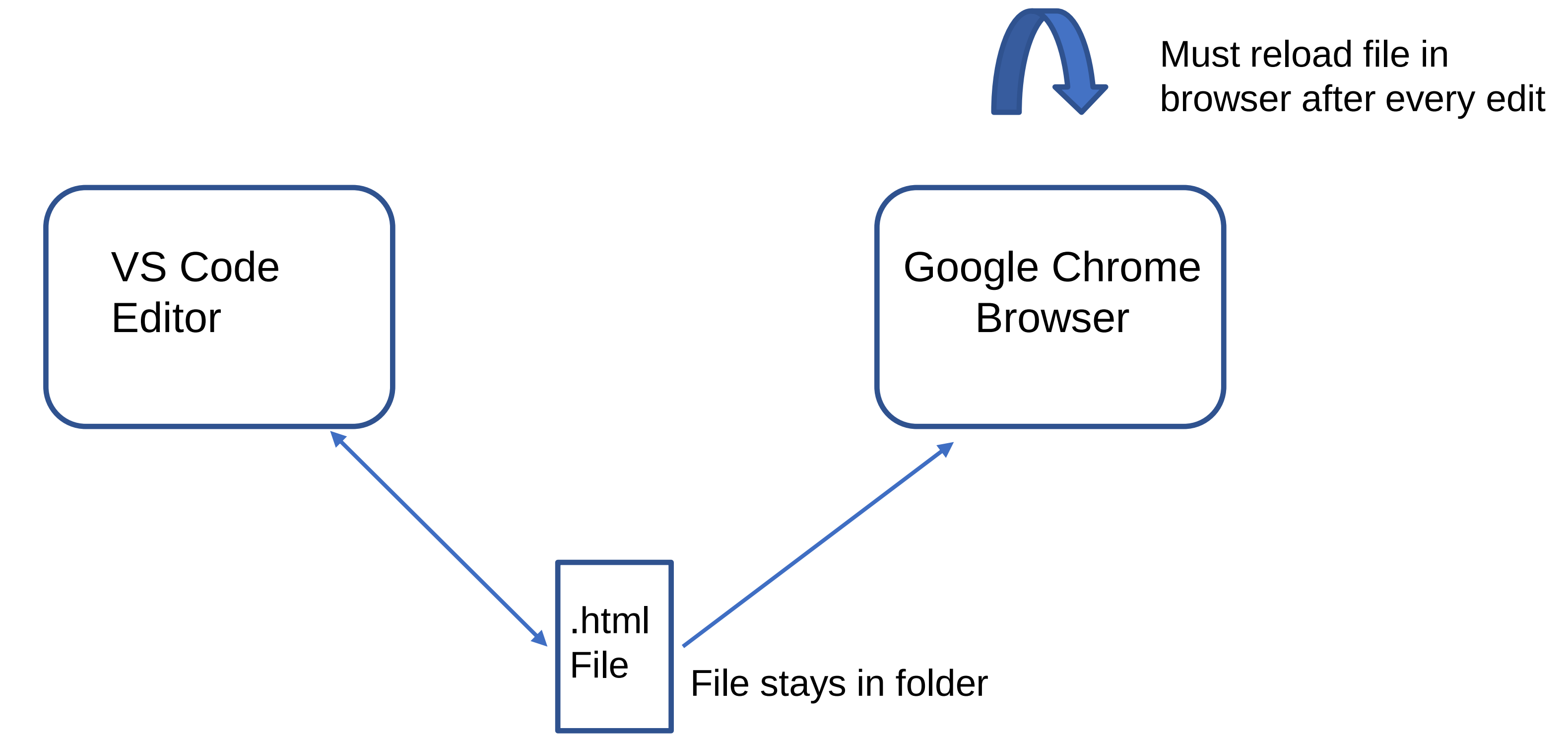
So, in order to edit the coding files, you will have to copy them from the zip file and place them in a folder on your computer:

Select all the files in the zip file, copy them, and paste them in a sensibly named folder. E.g. “Week 10 Hello World Coding Files”

Make sure that you know **exactly** where your files are. You do not want to submit a wrong version of your answer file, and you don't want to edit the wrong version of your program!

Working with the .html file

The html files that you will write Javascript in reside on your computer in a folder. This folder is also used as the workspace for VS Code.
While in the folder, the file is edited by VS Code, and read by Google Chrome. After every edit you have to reload it in Chrome.
You can do this by hitting the reload button, or by re-opening it while you are in VS Code.
Save your file after every edit! Ctrl-C (Windows) or Command-C (Mac).



File Naming Convention:

- lastname_firstname_Unit1_01_HelloWorld.html
  - Sample renaming: doyle_mart_Unit1_01_HelloWorld.html
- Properly naming your file is very important!
  - Improperly named files will not receive points/credit (no exceptions)
- Always check that you are saving your files to the correct location
- Always verify that you are saving an .html file type

Let's Code Hello World!

In-Class Activity

lastname_firstname_Unit1_01_HelloWorld.html

Values and Variables

Week 5

FOX
MIS

TIPS FROM MIS 2101 VIRTUAL HELPDESK

Programming is the “Reading, Writing and Arithmetic” of the Digital Age with Michelle Purnama

TEMPLE UNIVERSITY

In JavaScript, every piece of data that you provide, or use is considered to contain a value.

We saw this example earlier…

```
alert("hello, world!");
```

These words have a specific representation under the covers. They are considered **values**.

Values

Because you'll be working with values a whole lot, there are **two things** you need to simplify your life when working with them. You need to be able to:

1. Easily identify them

2. Reuse them without unnecessarily duplicating the value itself

Variables

The way to use variables is by using the **var** keyword followed by the name you want to give your variable. Here is us declaring a variable:

```
var myText
```

Right now, your variable has simply been **declared**. It doesn't contain a value. We can fix that by **initializing** our variable to a value like...

Variables with Values

Our variable **myText** has now been initialized to the value of **hello, world!**

```
var myText = “hello, world!”;
```

```
<script>
  var myText = "hello, world!";
  alert(myText);
</script>
```

The “assignment” operator

“alert()” is used to display information

What gets displayed is **hello, world!**

```
<script>
  var myText = “hi everybody!”;
  alert(myText);
</script>
```

What gets displayed now?

```
<script>
  var myText = prompt("What text would you like to display? ");
  alert(myText);
</script>
```

"prompt()" lets you get input from the user

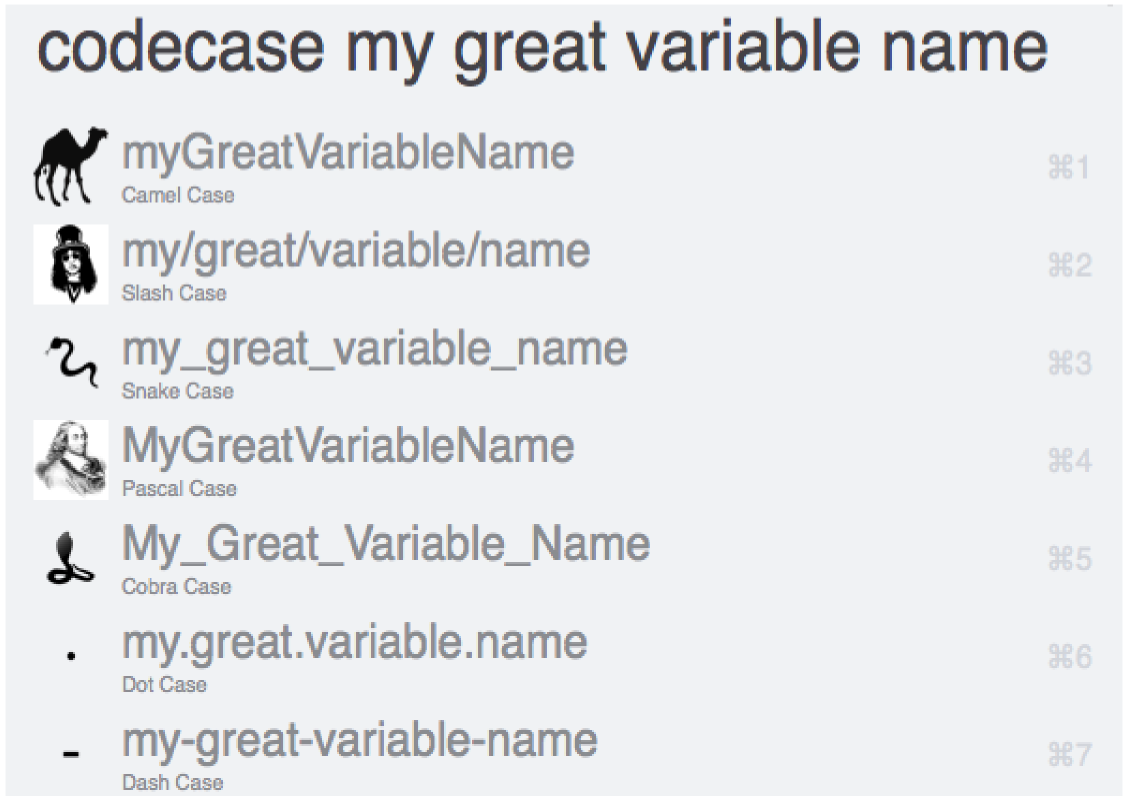
What gets displayed now?

Throughout your code, wherever you referenced the **myText** variable, you will now see the new text appear.

For larger applications, this convenience with having just one location where you can make a change that gets reflected everywhere is a major time saver.

You have a lot of freedom in naming your variables however you see fit. Ignoring what names you should give things based on philosophical / cultural / stylistic preferences, from a technical point of view, JavaScript is **very lenient** on what characters can go into a variable name.

Note: JavaScript is case sensitive. If the variable is myText, it has to be referenced as myText not MYtEXT.


codecase my great variable name

myGreatVariableName	Camel Case	⌘1
my/great/variable/name	Slash Case	⌘2
my_great_variable_name	Snake Case	⌘3
MyGreatVariableName	Pascal Case	⌘4
My_Great_Variable_Name	Cobra Case	⌘5
my.great.variable.name	Dot Case	⌘6
my-great-variable-name	Dash Case	⌘7

Source:
https://content.invisioncic.com/r229491/monthly_2018_02/codecase.png.9abc0521d2c0925906f5d0c309ed3f7b.png

Basically, keep the following things in mind when naming your variable:

1. Your variables can be as short one character, or they can be as long as you want - think thousands and thousands...and thousands of characters.

2. Your variables can start with a letter, underscore, or the $ character. They can't start with a number.

3. Outside of the first character, your variables can be made up of any combination of letters, underscores, numbers, and $ characters. You can also mix and match lowercase and uppercase to your heart's content.

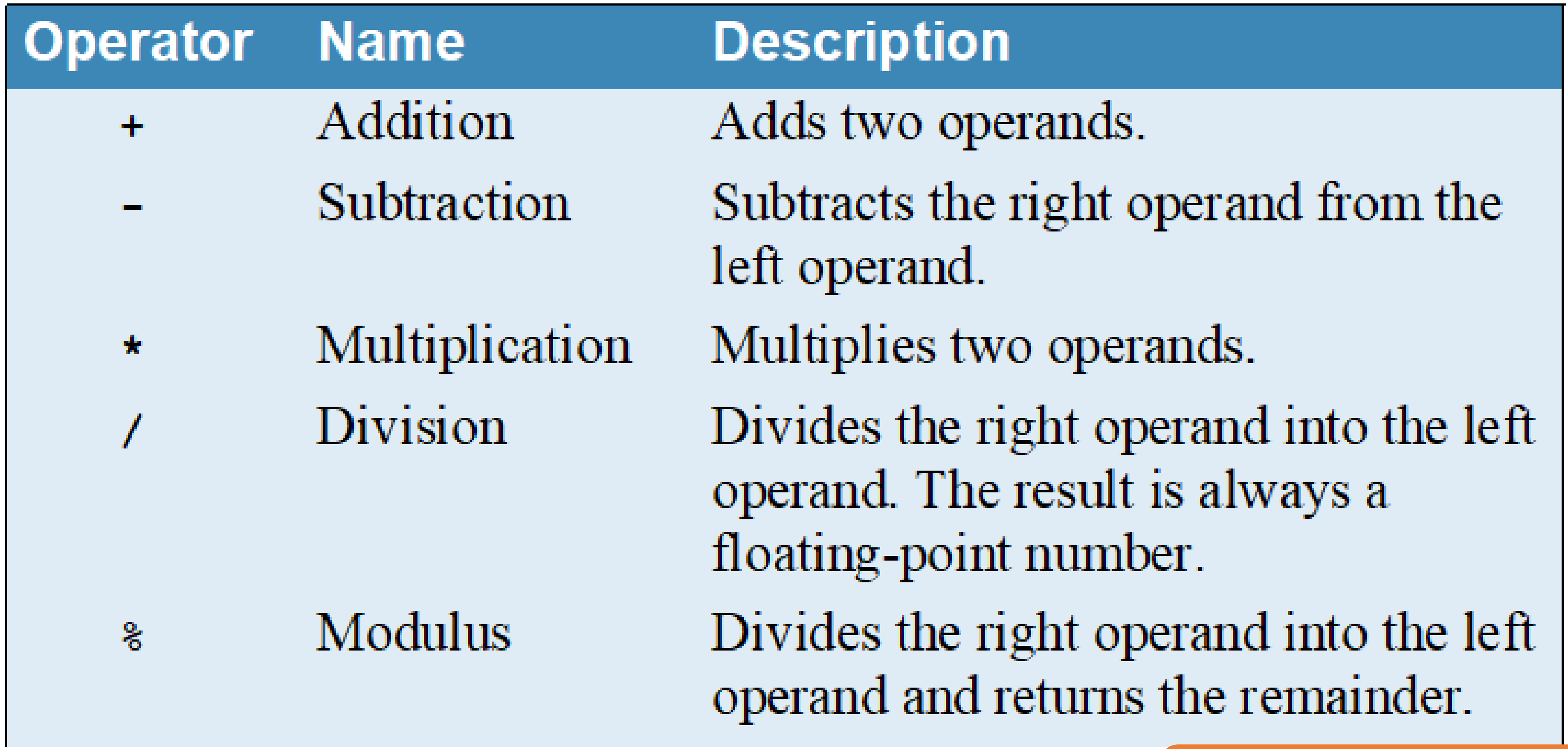
4. Spaces are not allowed.

```
var myText;
var $;
var r8;
var _counter;
var $field;
var thisIsALongVariableName_butItCouldBeLonger;
var __$abc;
var OldSchoolNamingScheme;
```

Which of the following is not a valid variable name?

1. myBest_variable
2. myBest$Variable
3. myBest variable
4. _otherVariable
5. $Variable
6. $_Variable
7. my2ndVariable
8. [✓ 2ndVariable]

Arithmetic operators



Operator	Name	Description
+	Addition	Adds two operands.
-	Subtraction	Subtracts the right operand from the left operand.
*	Multiplication	Multiplies two operands.
/	Division	Divides the right operand into the left operand. The result is always a floating-point number.
%	Modulus	Divides the right operand into the left operand and returns the remainder.

WARNING! This is ***not*** a complete list.

```
subtotal = 200;
taxPercent = .05;
taxAmount = subtotal * taxPercent;                  // 10
total = subtotal + taxAmount;                       // 210
```

Code that calculates the perimeter of a rectangle

```
width = 4.25;
length = 8.5;
perimeter = (2 * width) + (2 * length)
                                          // (8.5 + 17) = 25.5
```

Concatenation operator

Operator	Description
+	Concatenates two values.
+=	Adds the result of the expression to the end of the variable.

Just a fancy name for putting two strings together.

Concatenation is a very common task!

How to concatenate string variables with the + operator

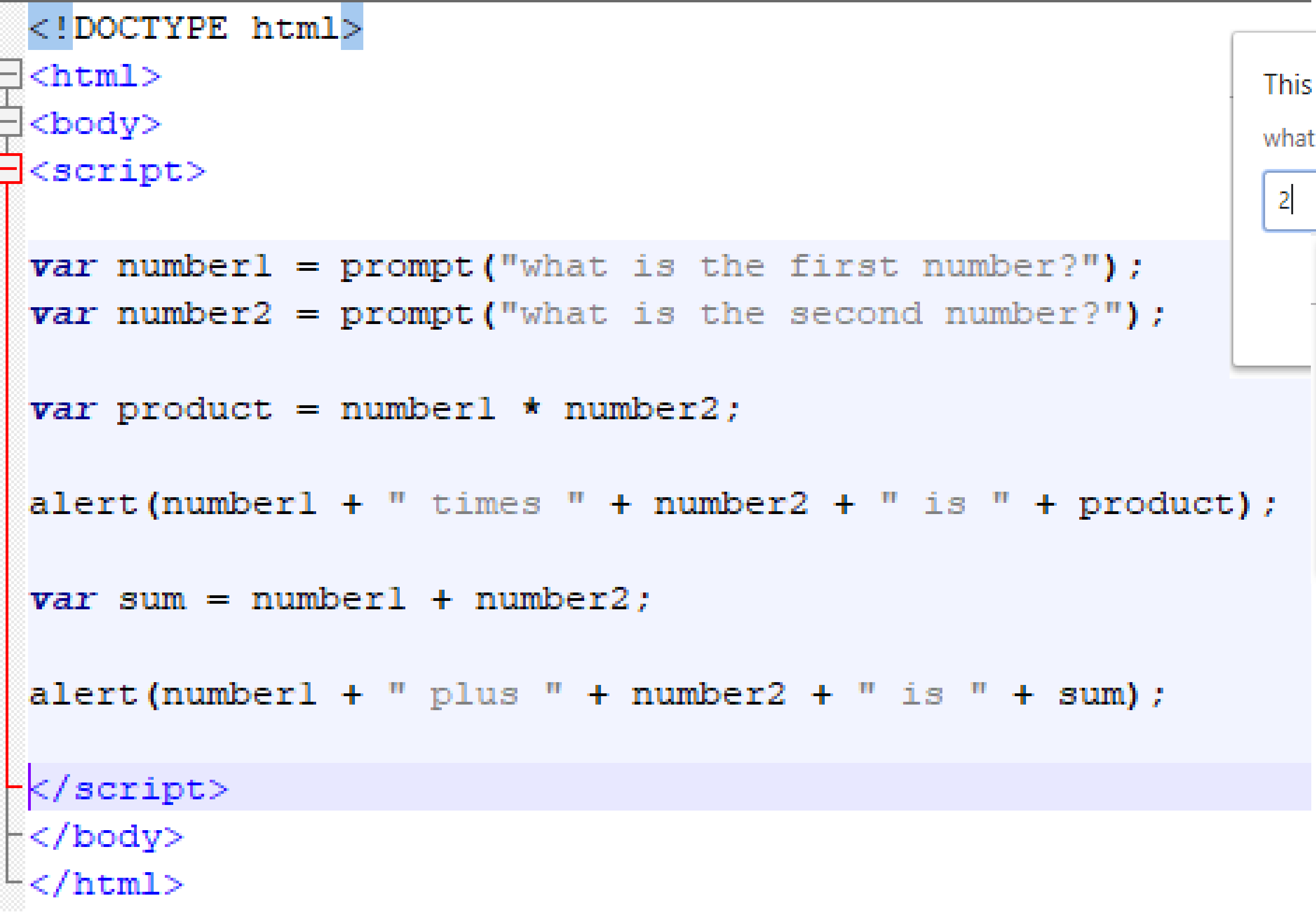
```
var firstName = "Grace", lastName = "Hopper";
var fullName = lastName + ", " + firstName;
                          // fullName is "Hopper, Grace"
```

How to concatenate string variables with the += operator

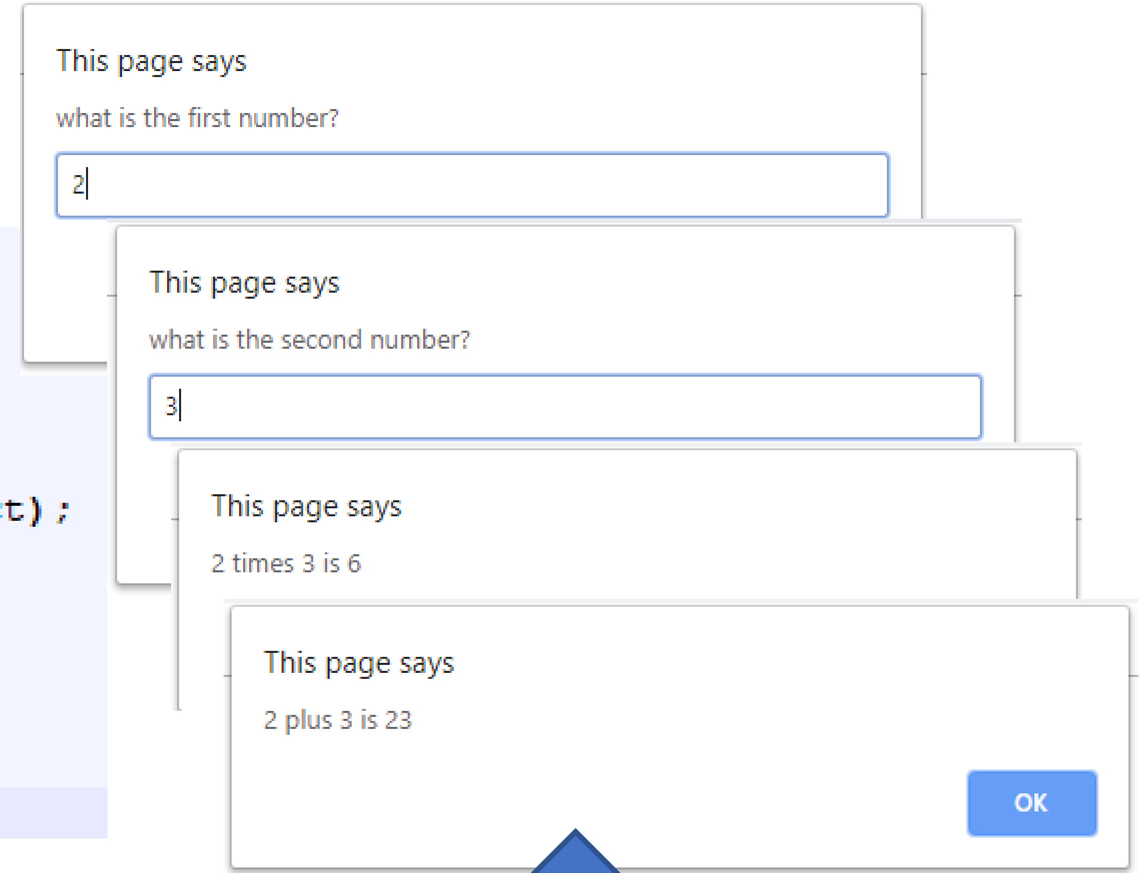
```
var firstName = "Grace", lastName = "Hopper";
var fullName = lastName;   // fullName is "Hopper"
fullName += ", ";          // fullName is "Hopper, "
fullName += firstName;     // fullName is "Hopper, Grace"
```

Prompt(), addition and concatenation

- Prompt always returns a string
- If the string looks like a number, JavaScript will convert the string to a number to do arithmetic
- JavaScript doesn't always do what you expect when using the "+" operator because it is used to perform addition when dealing with numbers and concatenation when dealing with strings
- If JavaScript isn't converting strings to numbers as you expect, use parseInt() for whole number or parseFloat() for decimal to convert them

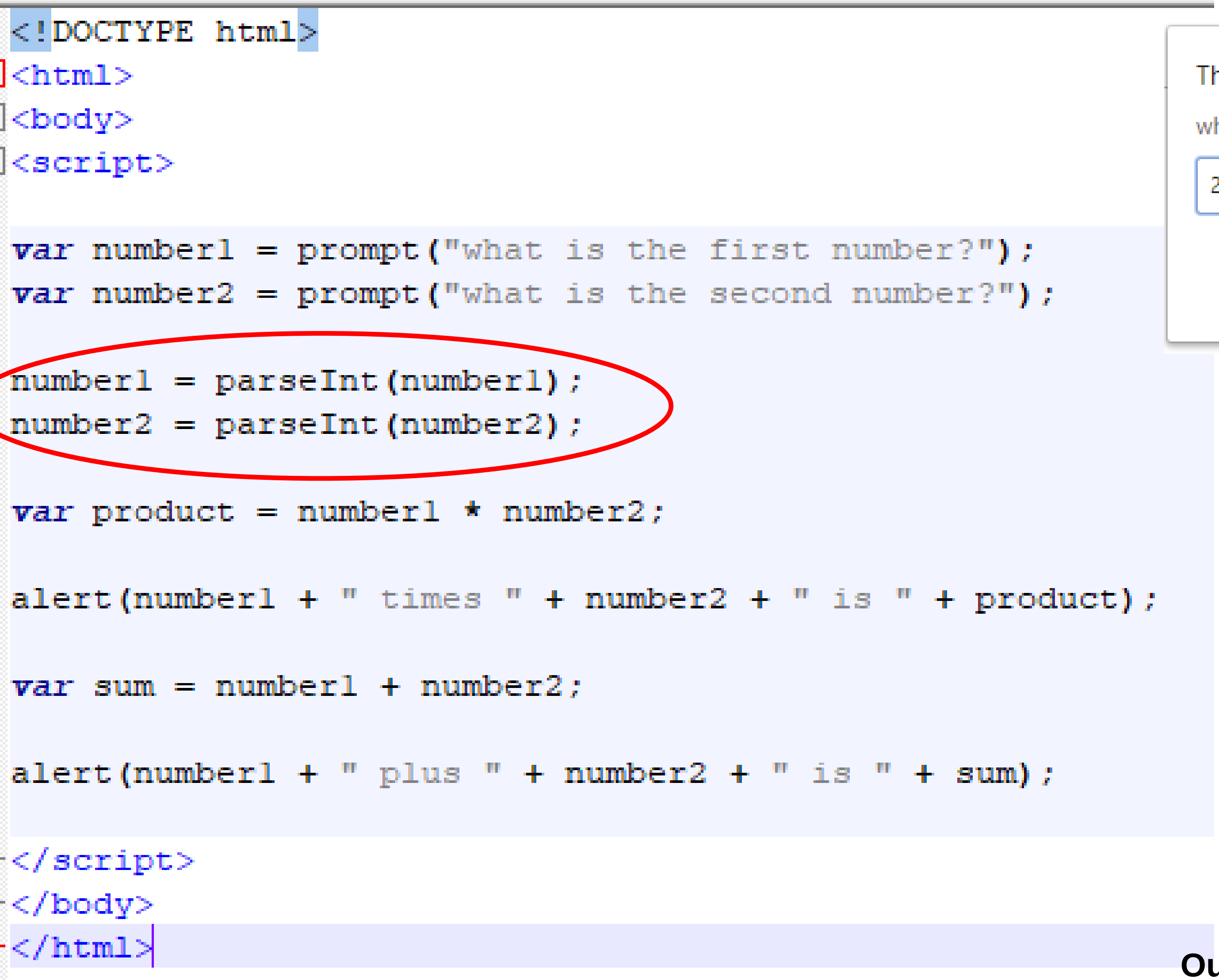


```
<!DOCTYPE html>
<html>
<body>
<script>

var number1 = prompt("what is the first number?");
var number2 = prompt("what is the second number?");

var product = number1 * number2;

alert(number1 + " times " + number2 + " is " + product);

var sum = number1 + number2;

alert(number1 + " plus " + number2 + " is " + sum);

</script>
</body>
</html>
```

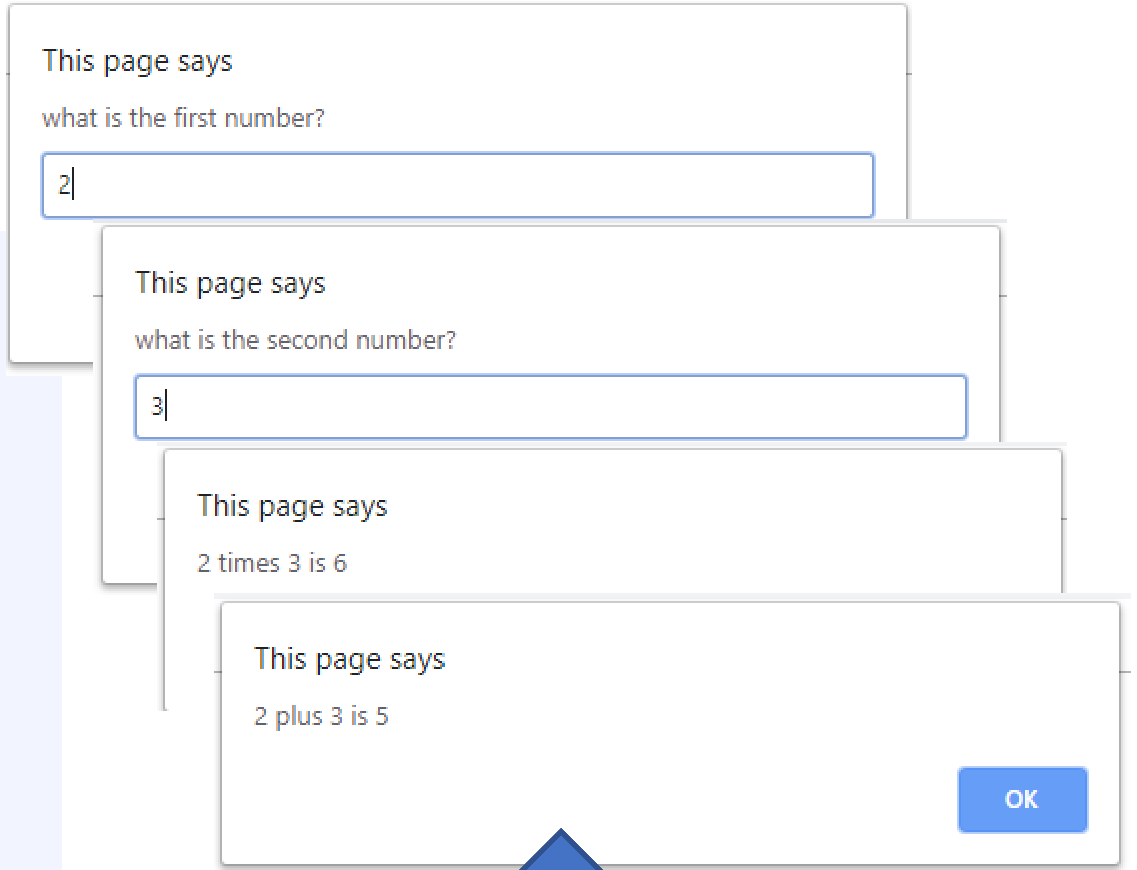


```
This page says
what is the first number?
2

This page says
what is the second number?
3

This page says
2 times 3 is 6

This page says
2 plus 3 is 23
```

Output if JavaScript doesn't convert the number to a number

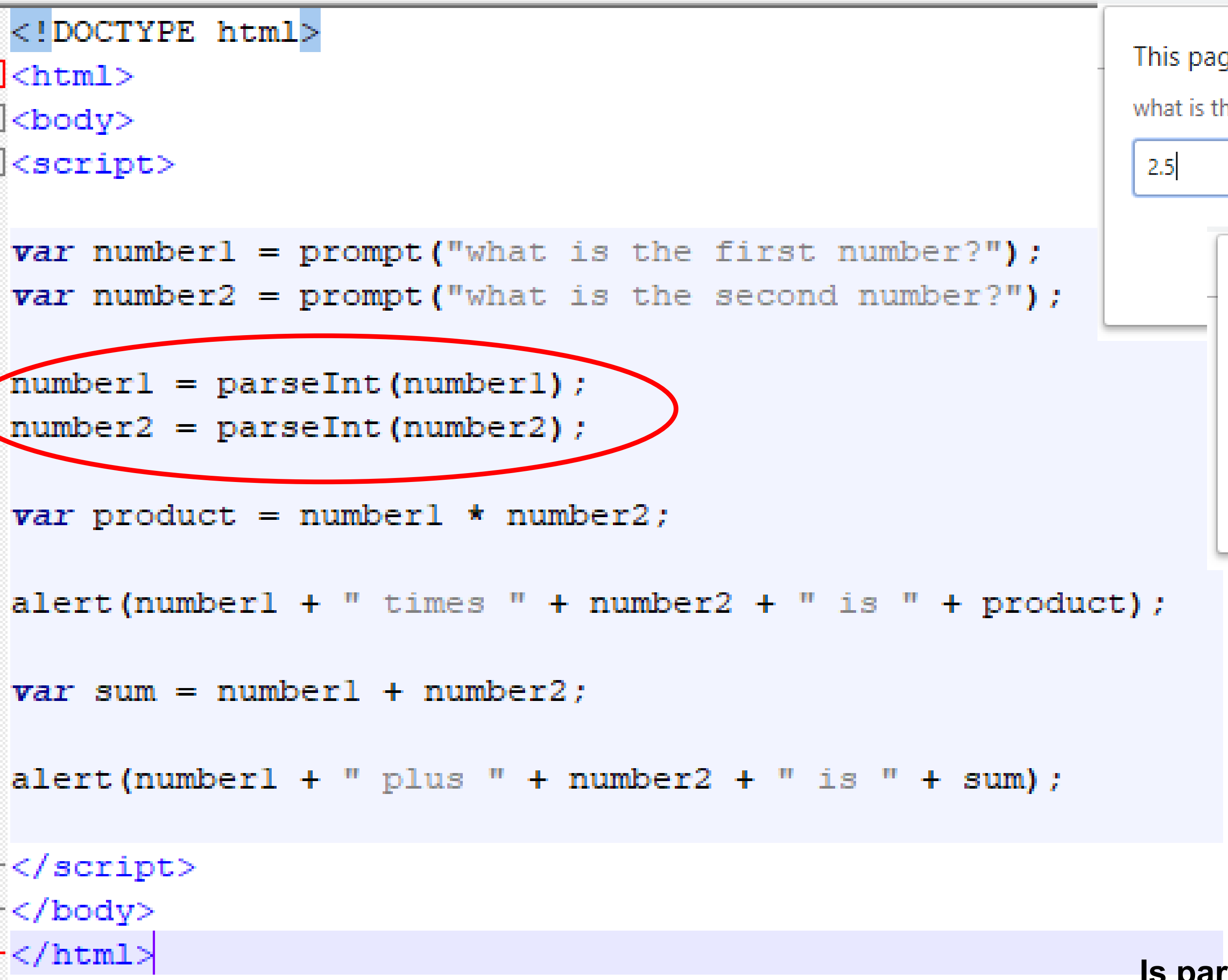


```
<!DOCTYPE html>
<html>
<body>
<script>

var number1 = prompt("what is the first number?");
var number2 = prompt("what is the second number?");

number1 = parseInt(number1);
number2 = parseInt(number2);

var product = number1 * number2;

alert(number1 + " times " + number2 + " is " + product);

var sum = number1 + number2;

alert(number1 + " plus " + number2 + " is " + sum);

</script>
</body>
</html>
```

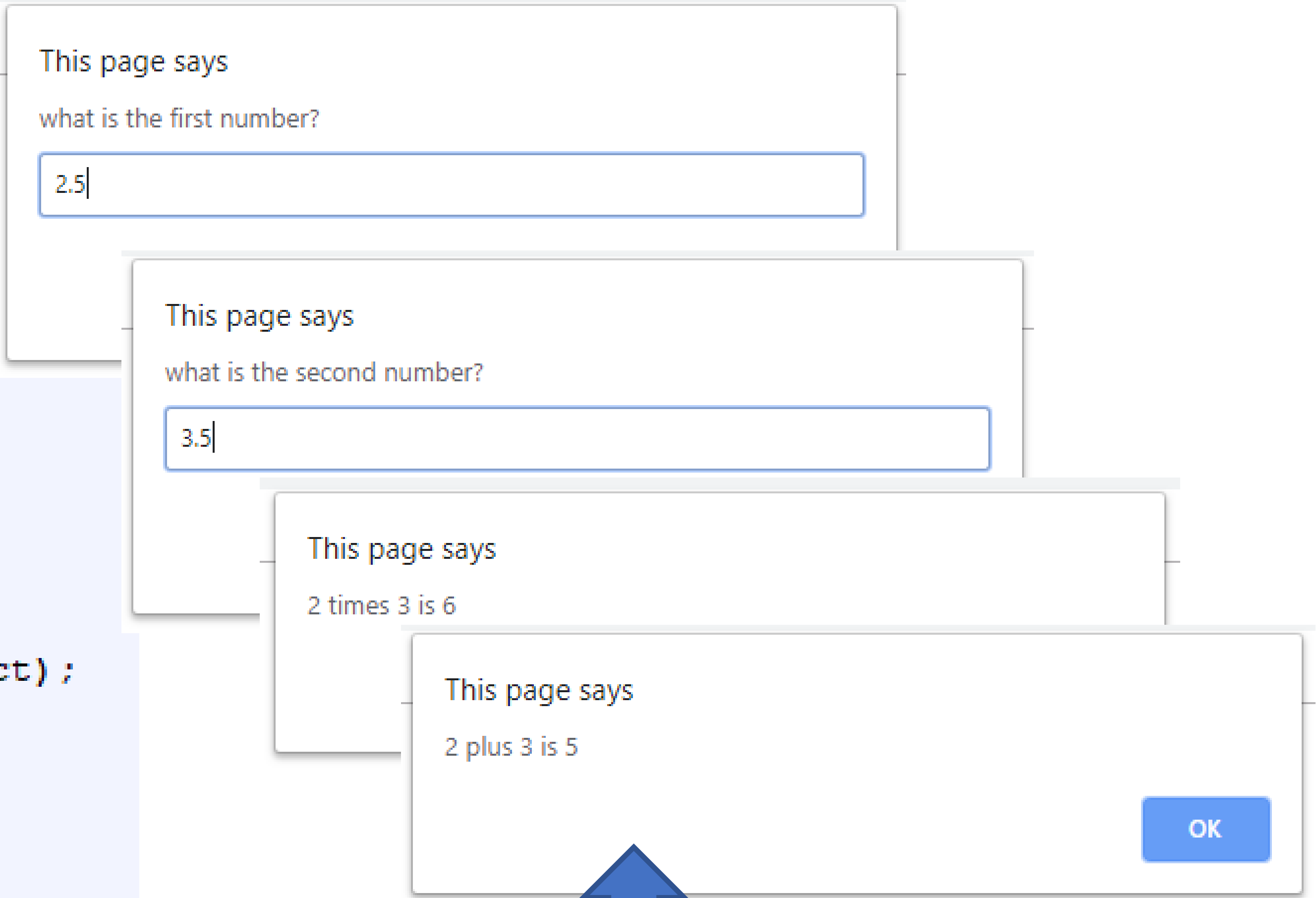


```
This page says
what is the first number?
2

This page says
what is the second number?
3

This page says
2 times 3 is 6

This page says
2 plus 3 is 5
```

Output once we use parseInt to convert the string to a number



```
<!DOCTYPE html>
<html>
<body>
<script>

var number1 = prompt("what is the first number?");
var number2 = prompt("what is the second number?");

number1 = parseInt(number1);
number2 = parseInt(number2);

var product = number1 * number2;

alert(number1 + " times " + number2 + " is " + product);

var sum = number1 + number2;

alert(number1 + " plus " + number2 + " is " + sum);

</script>
</body>
</html>
```



This page says

what is the first number?

2.5

This page says

what is the second number?

3.5

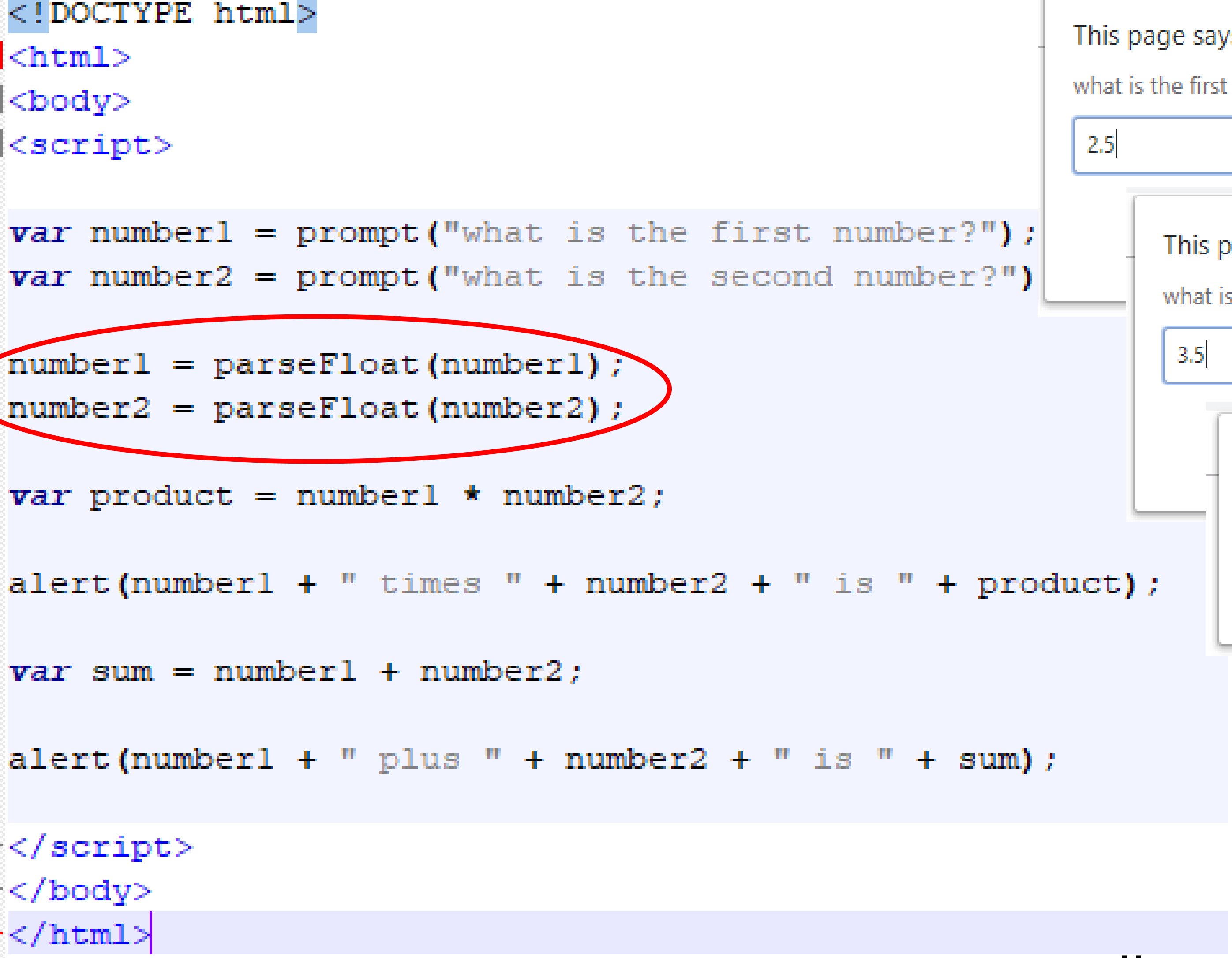
This page says

2 times 3 is 6

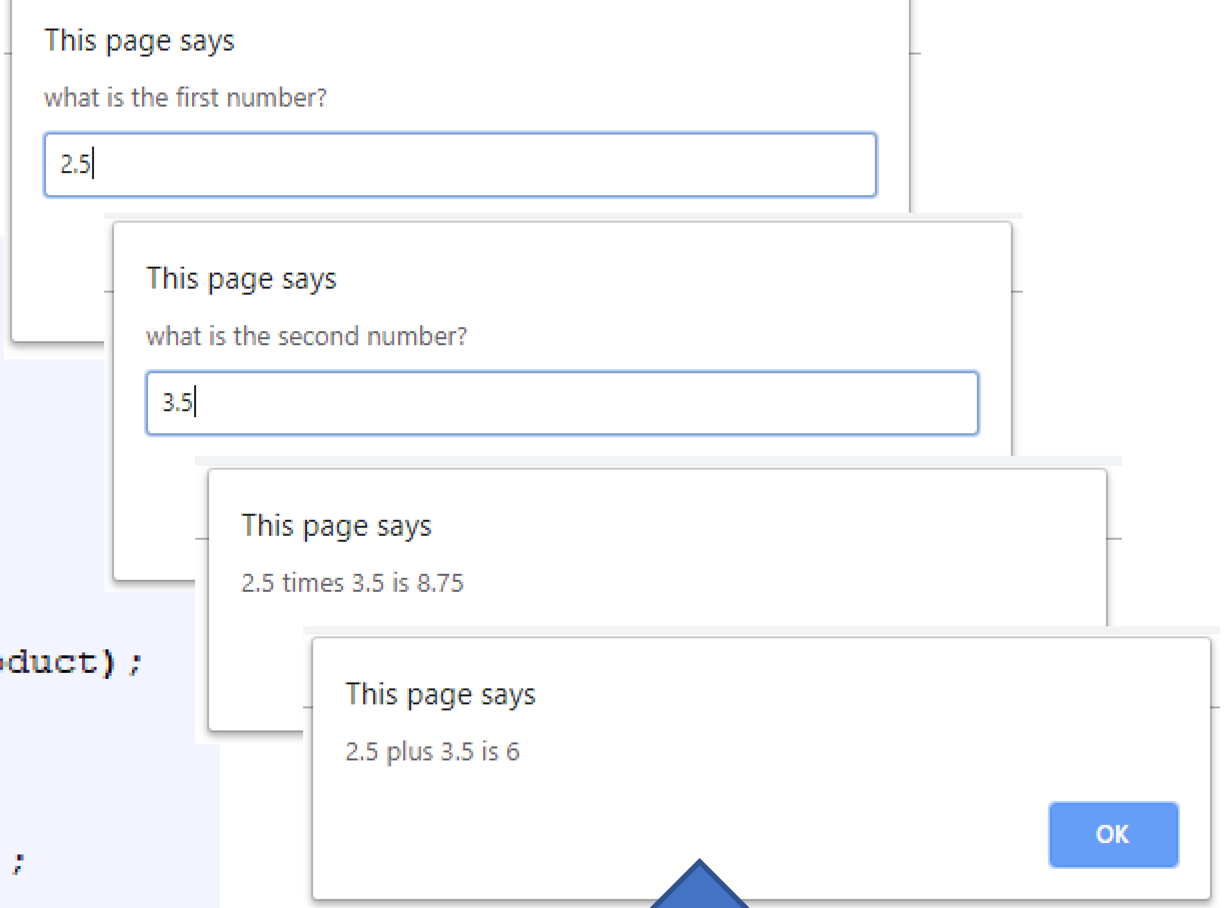
This page says

2 plus 3 is 5

Is parseInt what we want if we're dealing with floating point numbers (a.k.a. numbers with a decimal point)?



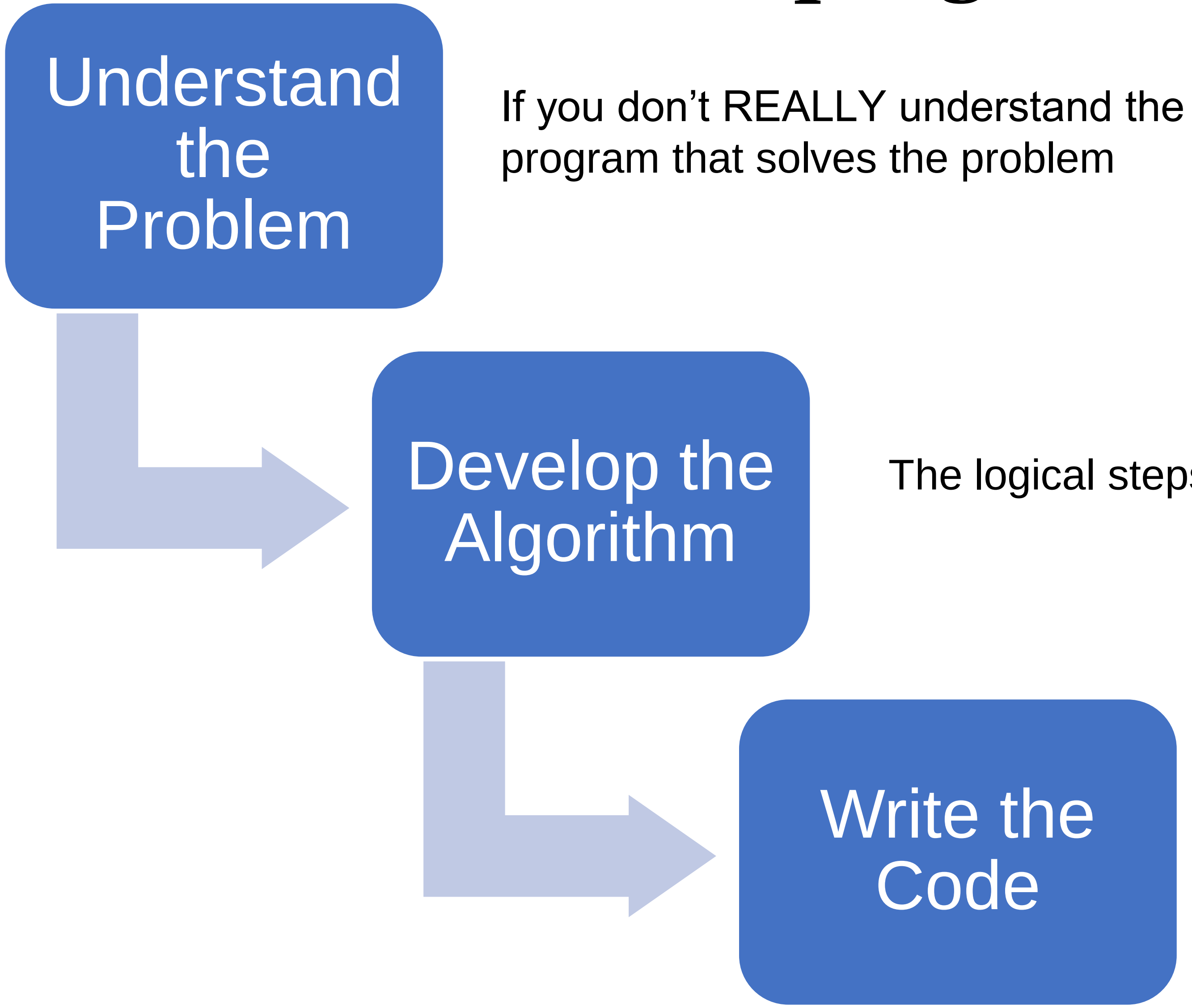
```
<!DOCTYPE html>
<html>
<body>
<script>

var number1 = prompt("what is the first number?");
var number2 = prompt("what is the second number?")

number1 = parseFloat(number1);
number2 = parseFloat(number2);

var product = number1 * number2;

alert(number1 + " times " + number2 + " is " + product);

var sum = number1 + number2;

alert(number1 + " plus " + number2 + " is " + sum);

</script>
</body>
</html>
```



```
This page says
what is the first number?
2.5

This page says
what is the second number?
3.5

This page says
2.5 times 3.5 is 8.75

This page says
2.5 plus 3.5 is 6
```

Use parseFloat when we're dealing with floating point numbers (a.k.a. numbers with a decimal point)?

How to write a program in 3 easy steps!



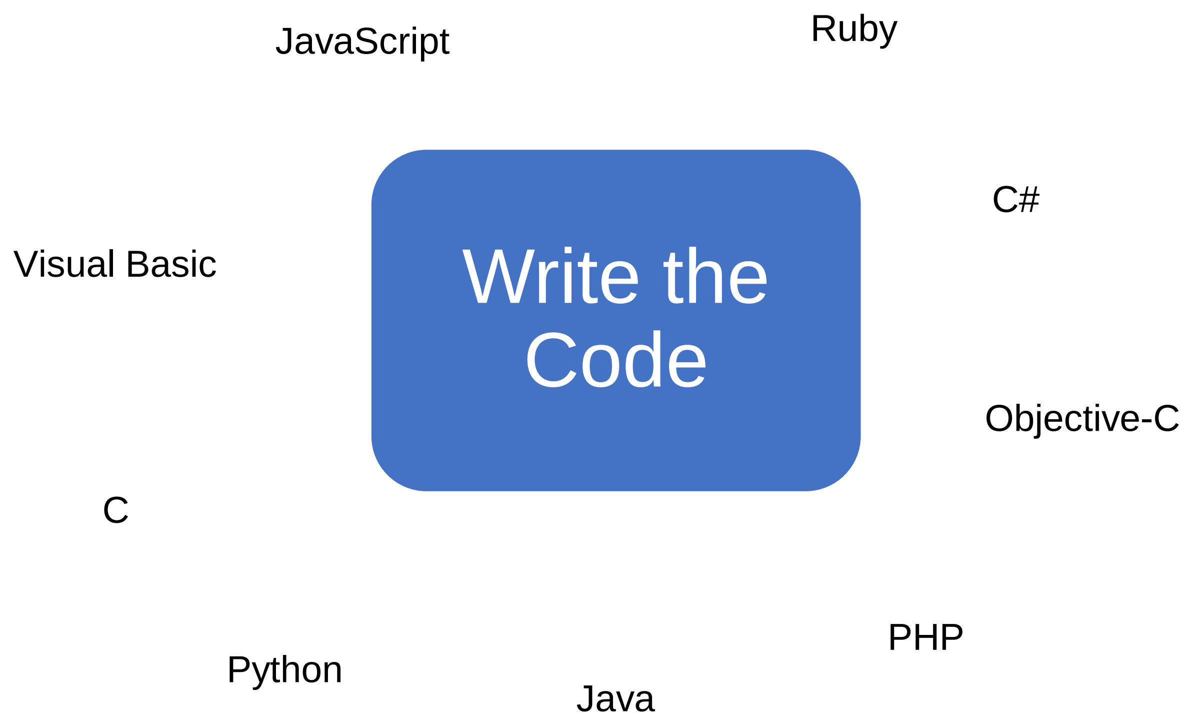
If you don’t REALLY understand the problem, you’ll never in a million years write a program that solves the problem

The logical steps that solves this particular problem

The automated process

The only part that is language specific...

- Once you have the algorithm (the hard part!), translating the algorithm to a particular programming language is fairly easy. If writing the code seems difficult, your problem is usually a bad algorithm!
- You can use lots of different languages. Some languages do some things better than others but they all do the same basic things.
- **In this class we will be using JavaScript, the de-facto standard for applications that run in a browser.**



If Swim Lane diagrams model processes and

If a program is just an automated process

Can we use a similar technique to model automated processes?

Pair programming is an agile software development technique in which two programmers work together at one workstation. One, the *driver*, writes code while the other, the *observer* or *navigator*, reviews each line of code as it is typed in. The two programmers switch roles frequently.

-Wikipedia

“Challenges”! = Practice this week!

Challenges

- HelloWorld (already done)
- Address
- GuessANumber
- Profits
- LandCalculations
- TotalPurchases

Diamond Peer Teacher
Jack Granieri

Address Coding Walkthrough

FOX
MIS

The code: Address

```
<html>
<body>
<script>

var name = prompt("What is your name? ");
var address = prompt("What is your street address? ");
var city = prompt("What city do you live in? ");
var state = prompt("What state do you live in? ");
var zip = prompt("What is your zipcode? ");
var telephone = prompt("What is your telephone number? ");
var major = prompt("What is your college major? ");

alert('Hello ' + name);
alert('from ' + address + ' ' + city + ', ' + state + ' ' + zip);
alert('who can be reached at ' + telephone);
alert('and is studying ' + major);

</script>
</body>
</html>
```

1. Make the tags: <html> for the html page, <body> for the content in the page, <script> for the code in the page.

2. Here we get the name from the user and store it in variable called name.
- “var” helps us make variables,
- “=“ assigns the value to the variable
- “prompt” gets the data from the user.

3. “alert”, displays a box on the browser with the information. Now display all the details

4. Close the tags

Diamond Peer Teacher
Sean Boyer

[Profits Coding Walkthrough]

FOX
MIS

The code: Profits

```
<html>
<body>
<script>


var totalSales = prompt("What are your projected total sales?
");
var profits = totalSales * .23;

alert('the expected profits on ' + totalSales + ' in total sales is ' +
profits);


</script>
</body>
</html>
```

1. Make the tags: <html> for the html page, <body> for the content in the page, <script> for the code in the page.

2. Here we get the total sales from the user and store it in variable called totalsales.
- “var” helps us make variables
- “=“ assigns the value to the variable
- “prompt” gets the data from the user.

3. “alert”, displays a box on the browser with the information. Now display all the details

4. Close the tags

Diamond Peer Teacher Patrick Jurgelewicz

Guess A Number Coding Walkthrough

FOX
MIS

The code: GuessANumber

```
<html>
<body>
<script>

var randomNumber = Math.floor(Math.random() * 100) + 1;

alert('The random number generated is ' + randomNumber);

</script>
</body>
</html>
```

1. Make the tags: <html> for the html page, <body> for the content in the page, <script> for the code in the page.

2. Make a variable for random number. Then assign the random number equation to it.
   - “var” helps in creating a variable
   - “=“ helps in assigning the value to a variable.

3. Display the random number

4. Close the tags

FOX MIS

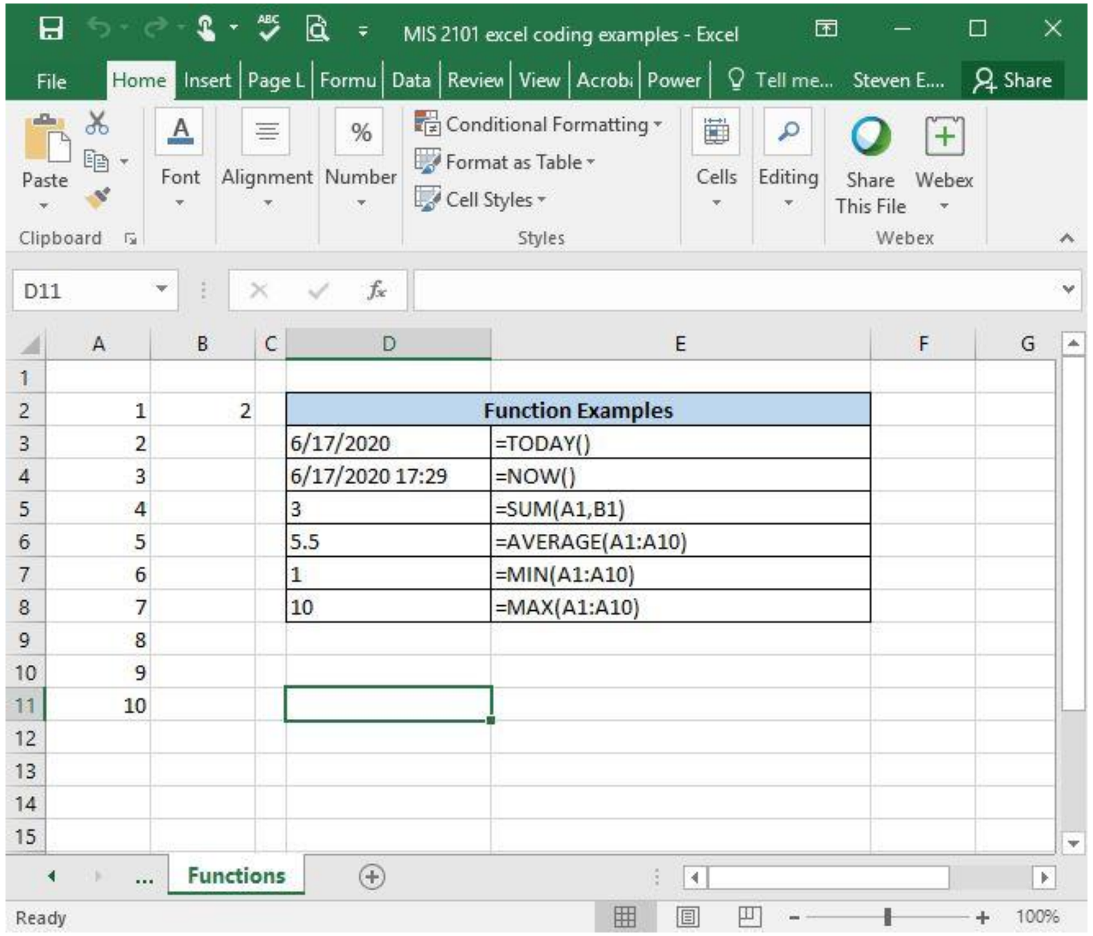
Functions

Week 5

FOX
MIS

What is a function in Excel?

- =TODAY()
- =NOW()
- =SUM(A1,B1)
- =AVERAGE(A1:A10)
- =MIN(A1:A10)
- =MAX(A1:A10)



	A	B	C	D	E
1					
2	1	2		**Function Examples**	
3	2			6/17/2020	=TODAY()
4	3			6/17/2020 17:29	=NOW()
5	4			3	=SUM(A1,B1)
6	5			5.5	=AVERAGE(A1:A10)
7	6			1	=MIN(A1:A10)
8	7			10	=MAX(A1:A10)
9	8				
10	9				
11	10				

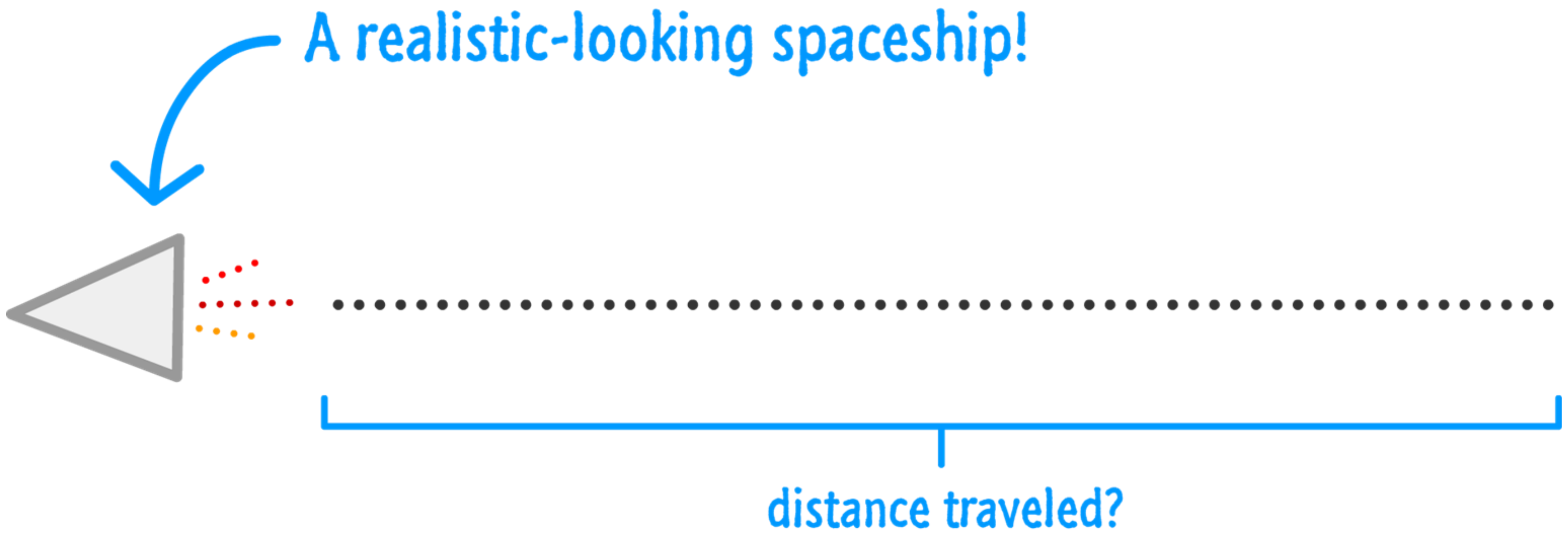
What is a function in Excel? (cont.)

- All functions
  - Have a name
  - Are passed zero or more pieces of information
  - Return a value

Functions allow our code to be more maintainable and reusable!

What is a function in JavaScript?

- A way to organize your code to make it easier to create, maintain and reuse
- All functions
  - Have a name
  - Are passed zero or more pieces of information
  - Return a value (usually)
- Main program just does basic input/output. All of the real work is packaged up and performed in functions



Source: JavaScript Absolute Beginner's Guide by Kirupa Chinnathambi distance = speed x time

Source: JavaScript Absolute Beginner’s Guide by Kirupa Chinnathambi

Meanwhile in JS Land

That diagram can be turned into the following:

```
var speed = 10;
var time = 5;
alert(speed * time);
```

Let’s say we have to calculate the distance multiple times.

Our code might look as follows.

```
var speed = 10;
var time = 5;
alert(speed * time);

var speed1 = 85;
var time1 = 1.5;
alert(speed1 * time1);

var speed2 = 12;
var time2 = 9;
alert(speed2 * time2);

var speed3 = 42;
var time3 = 21;
alert(speed3 * time3);
```

You should avoid unnecessarily repeating code. It makes your life more complicated.

This is where functions come in...

Meet the Function

Using functions, the code we saw earlier can look like this:

```
function showDistance(speed, time) {
 alert(speed * time);
}


showDistance(10, 5);
showDistance(85, 1.5);
showDistance(12, 9);
showDistance(42, 21);
```

What exactly is a function?

At a very basic level, a function is nothing more than a wrapper for some code. It does two things well:

1. Groups statements together

2. Makes your code reusable

You will rarely write or use code that doesn't involve functions!

```
var speed = 10;
var time = 5;
alert(speed * time);

var speed1 = 85;
var time1 = 1.5;
alert(speed1 * time1);

var speed2 = 12;
var time2 = 9;
alert(speed2 * time2);

var speed3 = 42;
var time3 = 21;
alert(speed3 * time3);
```

vs.

```
function showDistance(speed,
time) {
 alert(speed * time);
}


showDistance(10, 5);
showDistance(85, 1.5);
showDistance(12, 9);
showDistance(42, 21);
```

A Simple Function

```
function sayHello() {
 alert("hello!");
}
```

You have the **function** keyword, followed by your function name, some weird parentheses and brackets, and the code your function will run when called.

Calling a Function

```
function sayHello() {
 alert("hello!");
}
sayHello();
```

The function call is typically the name the function you want to **call** (aka **invoke**) followed again by the parentheses.

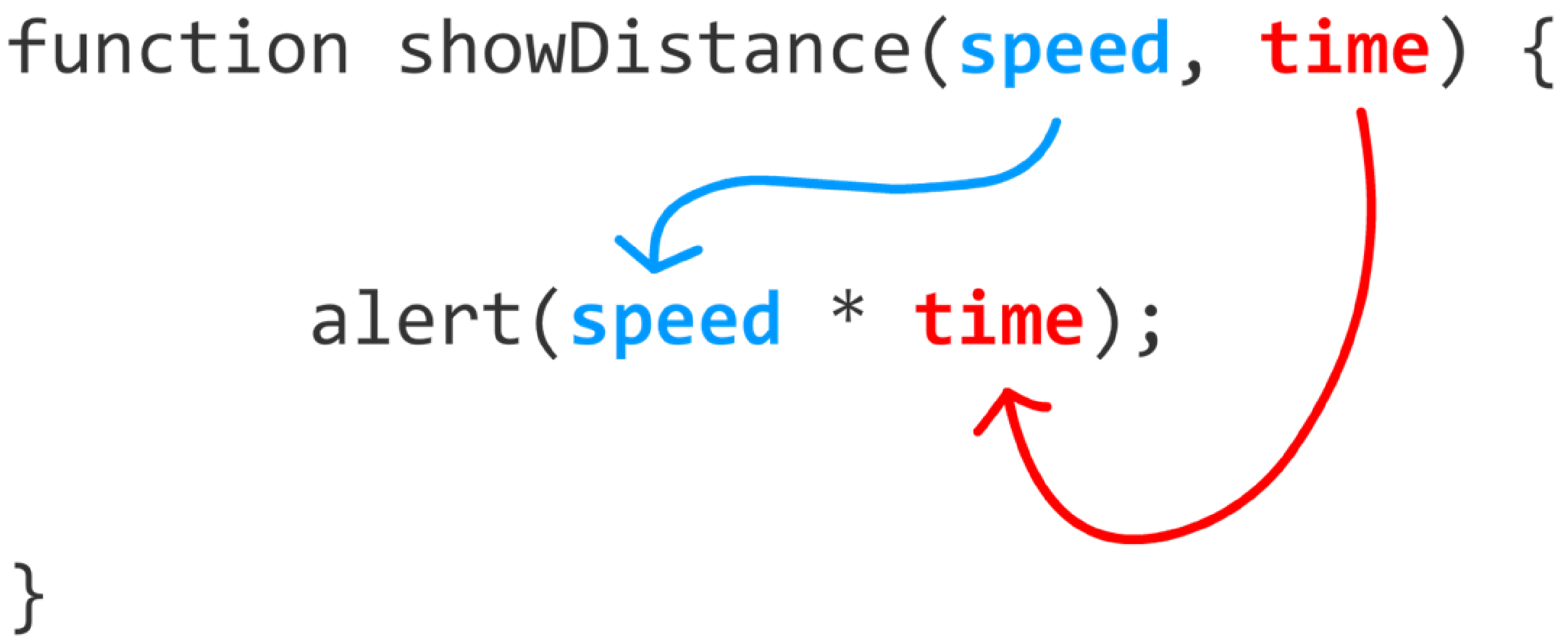
What exactly a function does can be customized. It doesn’t have to be boring and predictable like what have seen so far. One way is by providing what are known as **arguments** where your function call contains some data that you pass into the function.

Hello, again!

The **showDistance** function takes two arguments: **speed**, **time**:

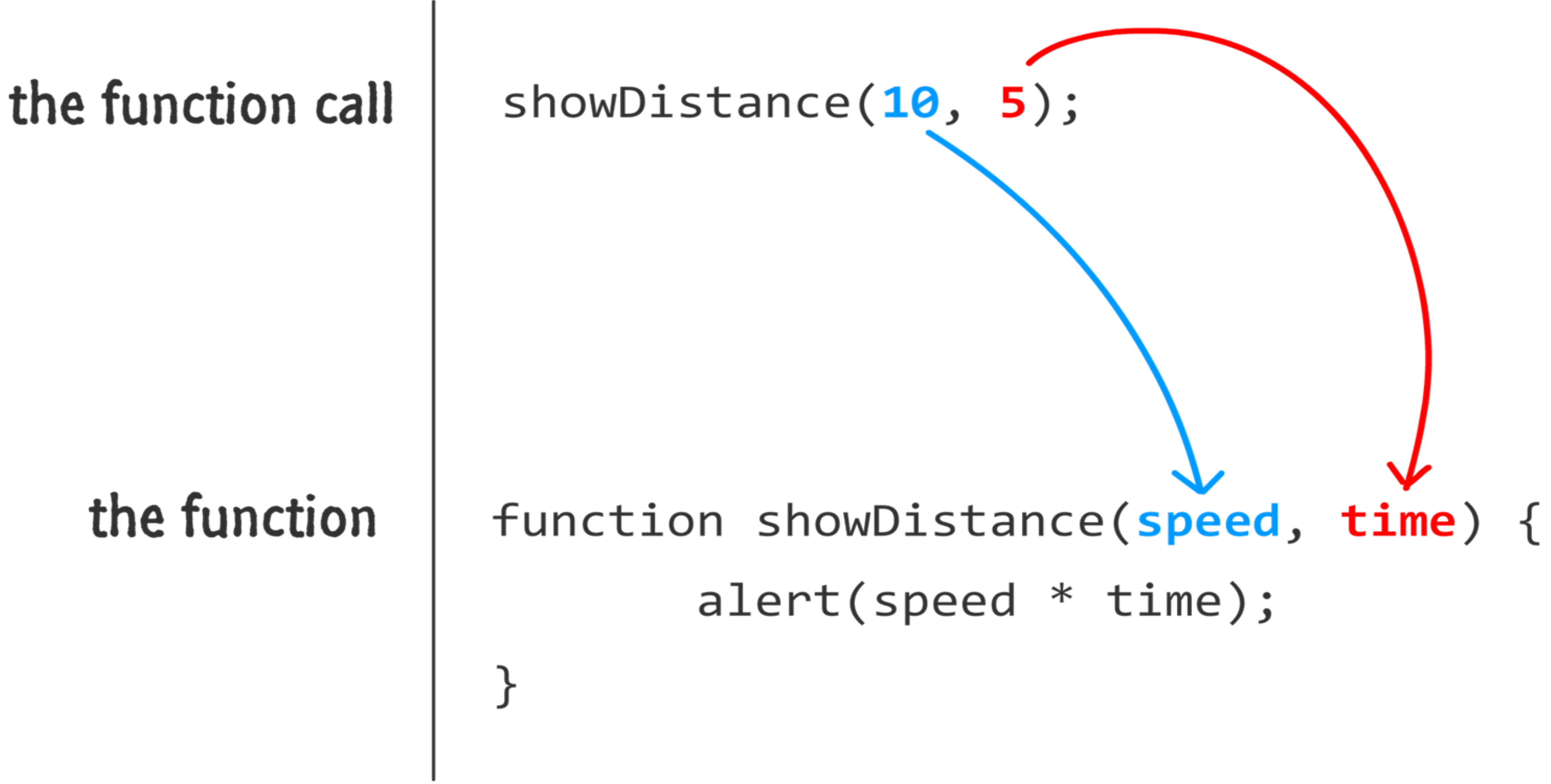
```
function showDistance(speed, time) {
 alert(speed * time);
}


showDistance(10, 5);
showDistance(85, 1.5);
showDistance(12, 9);
showDistance(42, 21);
```

the function



```
function showDistance(speed, time) {

        alert(speed * time);

}
```

Source: JavaScript Absolute Beginner’s Guide by Kirupa Chinnathambi



the function call

```
showDistance(10, 5);
```

the function

```
function showDistance(speed, time) {
        alert(speed * time);
}
```

Source: JavaScript Absolute Beginner's Guide by Kirupa Chinnathambi

Returning Data

```
function getDistance(speed, time) {
  var distance = speed * time;
  return distance;
}

var myDistance = getDistance(10, 5);
alert(myDistance);
```

The **return** keyword allows you to send data back to whatever called your function in the first place.

Once your **function** hits the **return** keyword, it stops everything it is doing at that point, returns whatever value you specified to the caller, and exits the function only. It does not exit the program!!!!

No code in your function after **return** will run.

```
function getDistance(speed, time) {
  var distance = speed * time;
  return distance;

  if (speed < 0) {
      distance *= -1;
  }
}
```

“Challenges”! = Practice this week!

Challenges

- **TotalDistance**
- **SalesTax**
- **MPG**
- **TipTaxTotal**
- **C2F**
- **IngredientAdjuster**

Diamond Peer Teacher Jack Granieri

Total Distance Coding Walkthrough

FOX MIS

The code: Total Distance

```
<html>
<body>
<script>

function distanceTraveled(drivingSpeed, timeDriving) {

        return drivingSpeed * timeDriving;

}

var speed = prompt('How fast are you traveling in miles per hour? ');
var time = prompt('How long are you driving at that speed in hours? ')

alert('While driving at ' + speed + ' miles/hour, you will have traveled ' +
distanceTraveled(speed, time) + ' miles in ' + time + ' hours.');


</script>
</body>
</html>
```

1. Make the tags: <html> for the html page, <body> for the content in the page, <script> for the code in the page.

2. Here we call the function. It is like an equation with variables in it. The word "function" helps in creating a function. The variables inside the parenthesis are the values a function takes. This function named distanceTraveled returns the distance which is the product of speed and time.

3. Here we get the speed in miles/hour and time in hours taken to drive.

- "var" helps us make variables
- "=" assigns the value to the variable
- "prompt" gets the data from the user.

4. "alert", displays a box on the browser with the information. Notice how the function is called here, with the parameters. Now display all the details

5. Close the tags

FOX MIS

Diamond Peer Teacher Patrick Jurgelewicz

Sales Tax Coding Walkthrough

FOX MIS

The code: Sales Tax

```
<html>
<body>
<script>

function orderTotal(subtotal) {

        var stateSalesTax = subtotal * .05;
        var countySalesTax = subtotal * .025;

        return subtotal + stateSalesTax + countySalesTax;

}

var amountOfPurchase = parseFloat(prompt('What is the amount of the
purchase? '));

alert('The total for the order is $' + orderTotal(amountOfPurchase) + '
include state and county sales tax.');

</script>
</body>
</html>
```

1.Make the tags: <html> for the html page, <body> for the content in the page, <script> for the code in the page.

2. Here we call the function. It is like an equation with variables in it. The word "function" helps in creating a function. The variables inside the parenthesis are the values a function takes. This function named orderTotal is called with subtotal as parameter and:
stateSalesTax which is calculate with respect to subtotal
countySalesTax which is also calculated with respect to subtotal

3. The function returns the equation by adding up all the variables

4. Here we get the amount in dollars from the user and store it in variable called amountOfPurchase.

- "var" helps us make variables
- "=" assigns the value to the variable
- "prompt" gets the data from the user.
- "parseFloat" converts the data into decimals

5. "alert", displays a box on the browser with the information. Notice how the function is called here, with the parameters. Now display all the details

6. Close the tags

FOX MIS

Homework Introduction
(peek under the hood)

- **Review Riley's Ranking Calculator:**
  - **Let's look at the functions**

Have a Safe and Happy
Thanksgiving!

More to Come

Prepare with Readings & Videos before our next class!!!