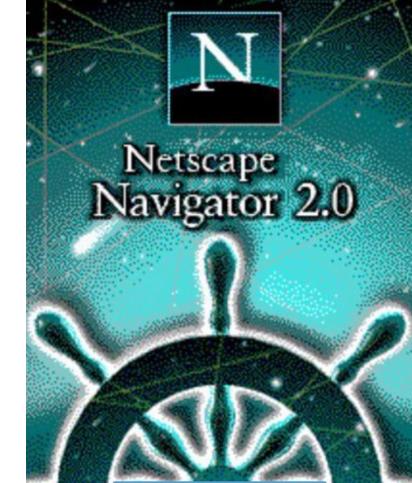


COMP 1842

Week1 – Part 2 - JS intro

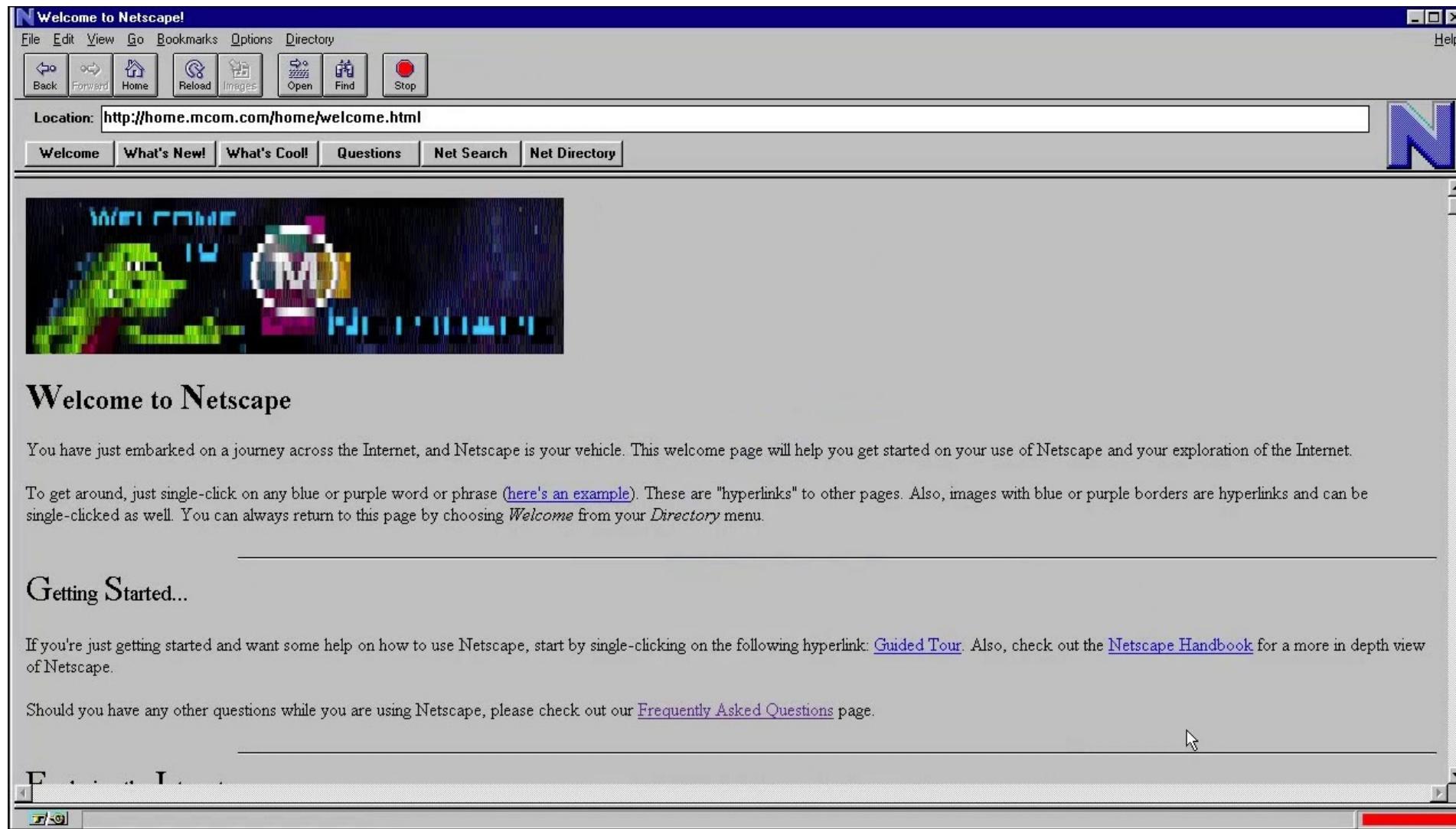
Matt Prichard

What is JavaScript



- JavaScript is a platform-independent, event-driven, interpreted programming language.
- JavaScript's official name is "ECMAScript". The standard is developed and maintained by the ECMA organisation.
- ECMAScript versions have been abbreviated to ES1, ES2, ES3, ES5, and ES6.
- Since 2016 new versions are named by year (ECMAScript 2016 / 2017 / 2018).
- The language was invented by Brendan Eich at Netscape (with Navigator 2.0), and has appeared in all Netscape (defunct since Mar 2008) and Microsoft browsers since 1996.

Time warp



What is JavaScript cont...

- JavaScript was designed to add interactivity and to provide client-side programming to HTML pages
- JavaScript is a scripting language or a lightweight programming language .
- Executable JavaScript may be included into an HTML page as either...
 - External - a separate file containing JavaScript
 - Embedded - as the content of a <script> element
 - Inline - as the value of an HTML event attribute
- Everyone can use JavaScript without purchasing a license

If you only remember one thing from today...

JavaScript
is not Java



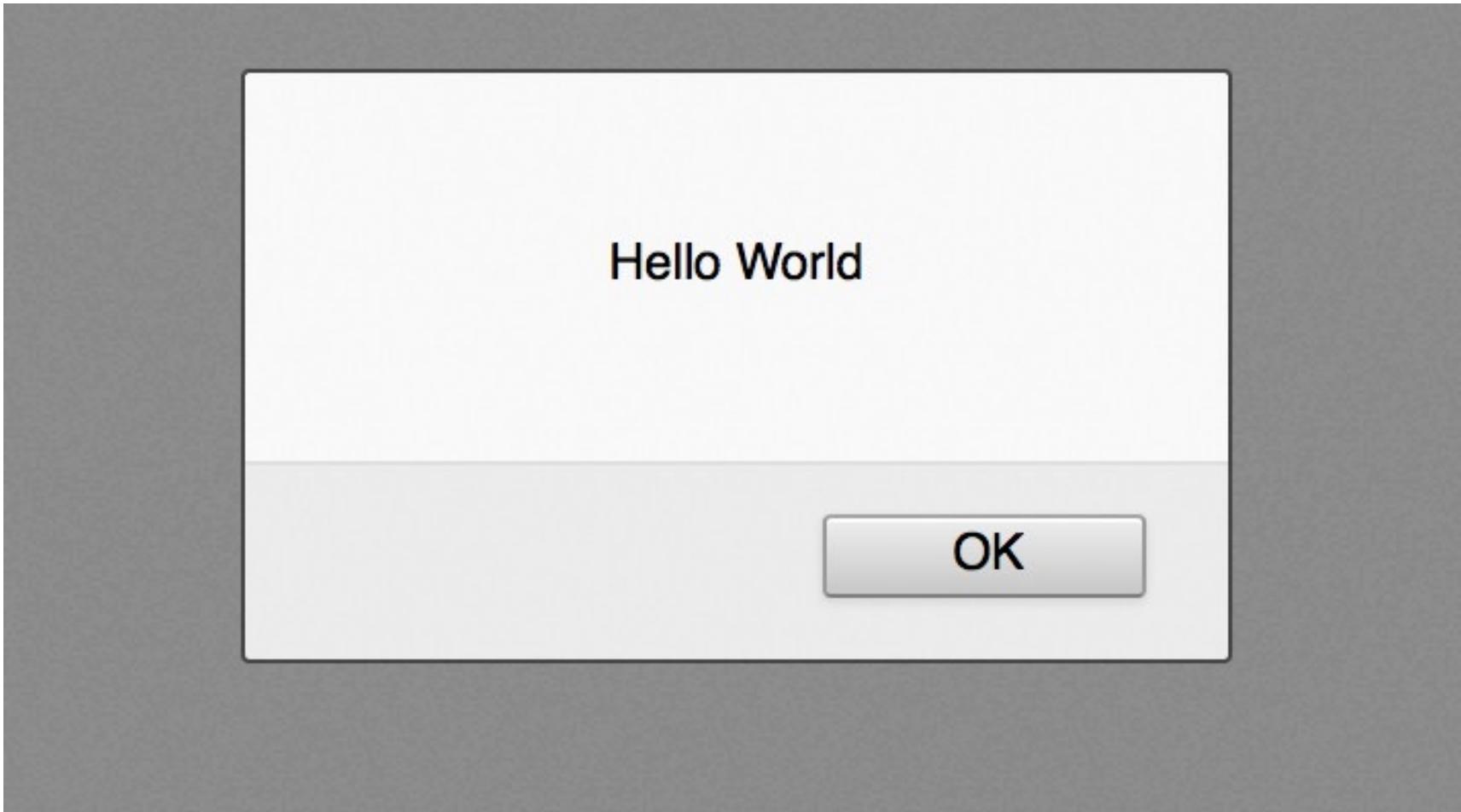
Introduction to JavaScript

demonstrations

```
1  <!DOCTYPE html>
2  <html lang="en">
3  <head>
4      <meta charset="UTF-8">
5      <title>Document</title>
6  </head>
7  <body>
8
9  <script>
10     alert("Hello World");
11 </script>
12
13 </body>
14 </html>
```

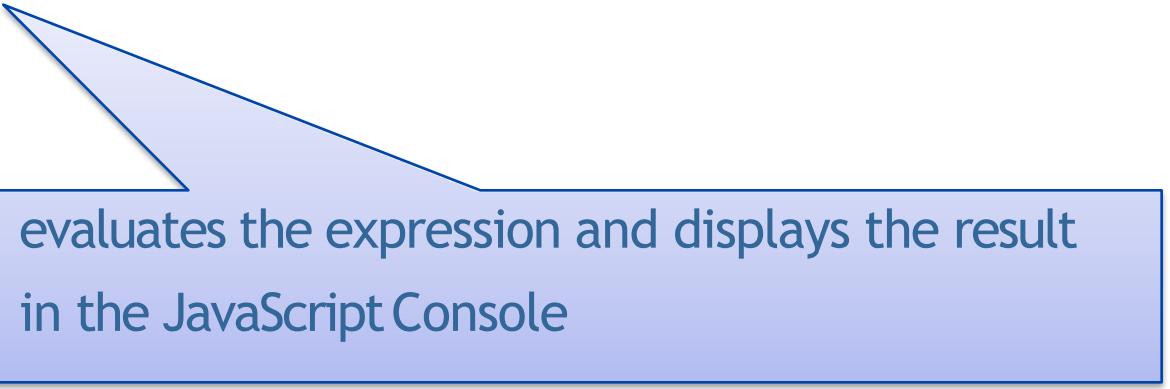
To insert a JavaScript into an HTML page, use the `<script>` tag. The `<script>` and `</script>` tells where the JavaScript starts and ends. The lines between the `<script>` and `</script>` contain the JavaScript.

Alert box in FireFox

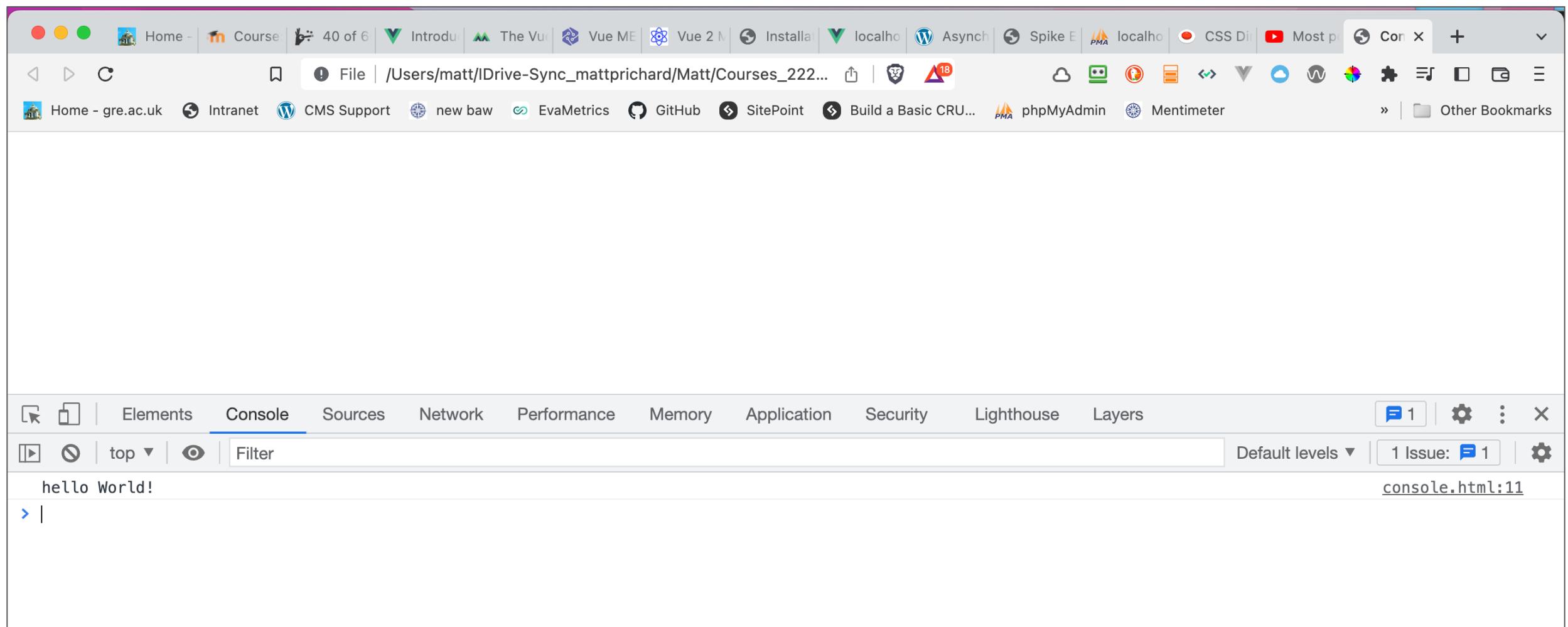


```
1  <!DOCTYPE html>
2  <html lang="en">
3  <head>
4  |     <meta charset="UTF-8">
5  |     <title>Document</title>
6  </head>
7  <body>
8
9  <script>
10 |     console.log("Hello World");
11 </script>
12
13
14 </body>
15 </html>
```

```
console.log(expression);
```



evaluates the expression and displays the result
in the JavaScript Console



innerHTML

- Another way to get output to the screen is using innerHTML
- This allows us to output to any HTML element we know the ‘id’ of.
- We create an HTML element and give it an ID.

```
12  
13  <p id="text"></p>  
14
```

innerHTML cont.

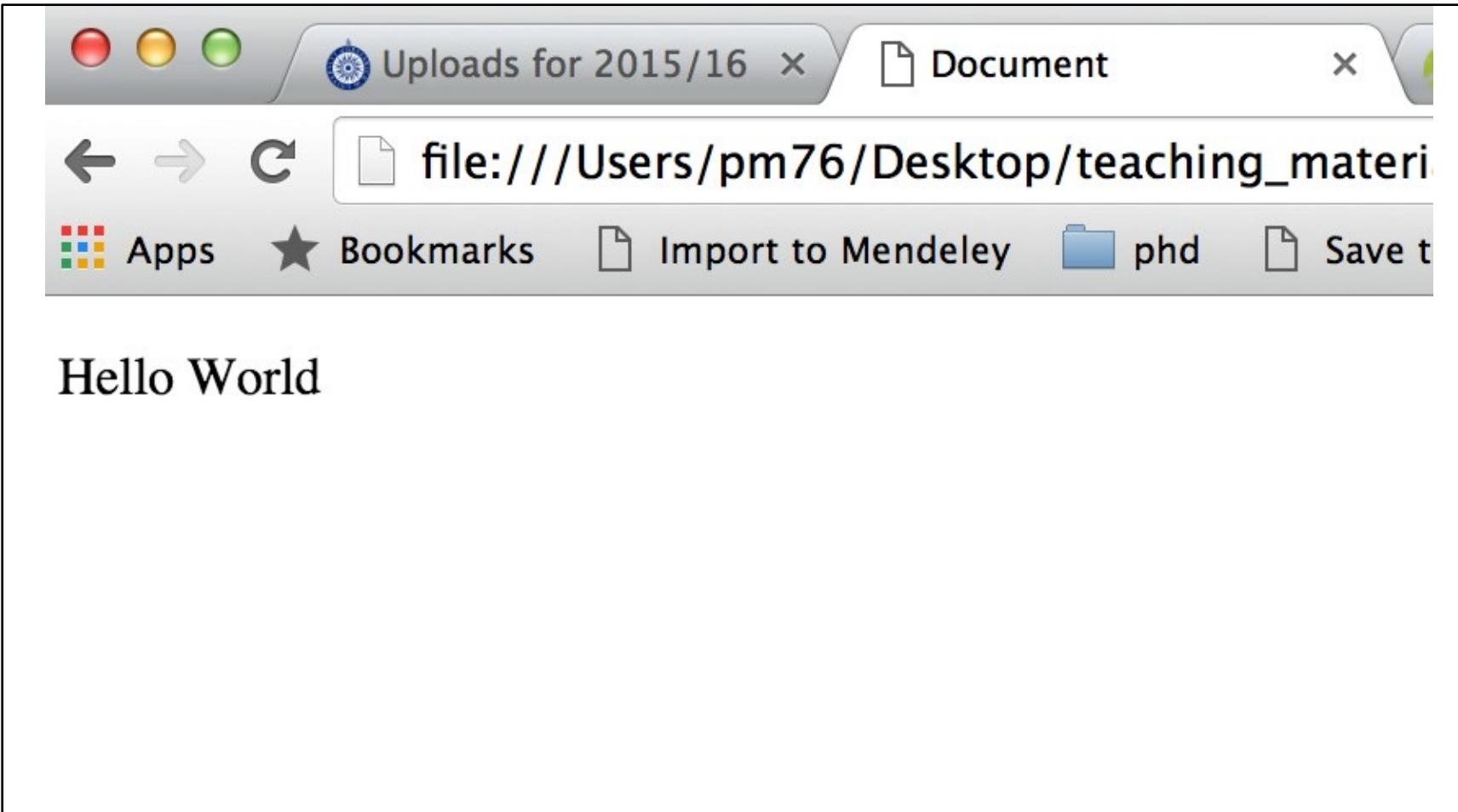
The getElementById() method accesses the first element with the specified id.

```
document.getElementById('text').innerHTML = "matt";
```

Then access the inner HTML property of that element and assign it a value

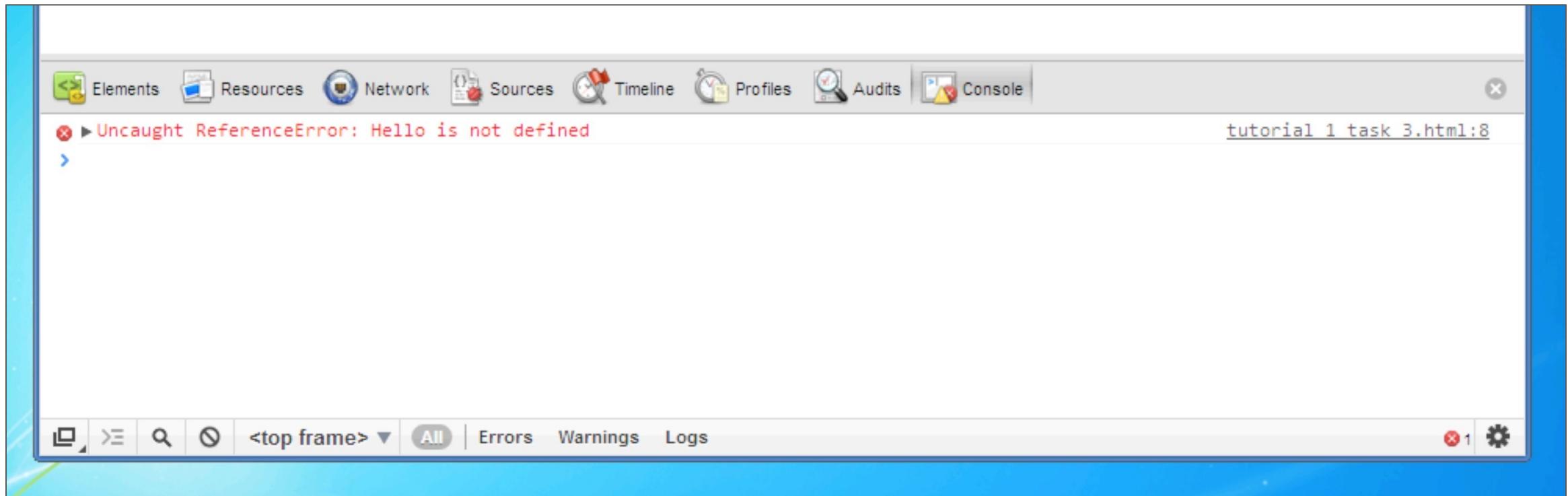
```
1  <!DOCTYPE html>
2  <html lang="en">
3  <head>
4      <meta charset="UTF-8">
5      <title>COMP1779 – JavaScript Tutorial 1</title>
6  </head>
7  <body>
8  <p id="text"></p>
9  <script>
10 |     document.getElementById("text").innerHTML = "Hello World";
11 </script>
12 </body>
13 </html>
```

Inner html in a browser



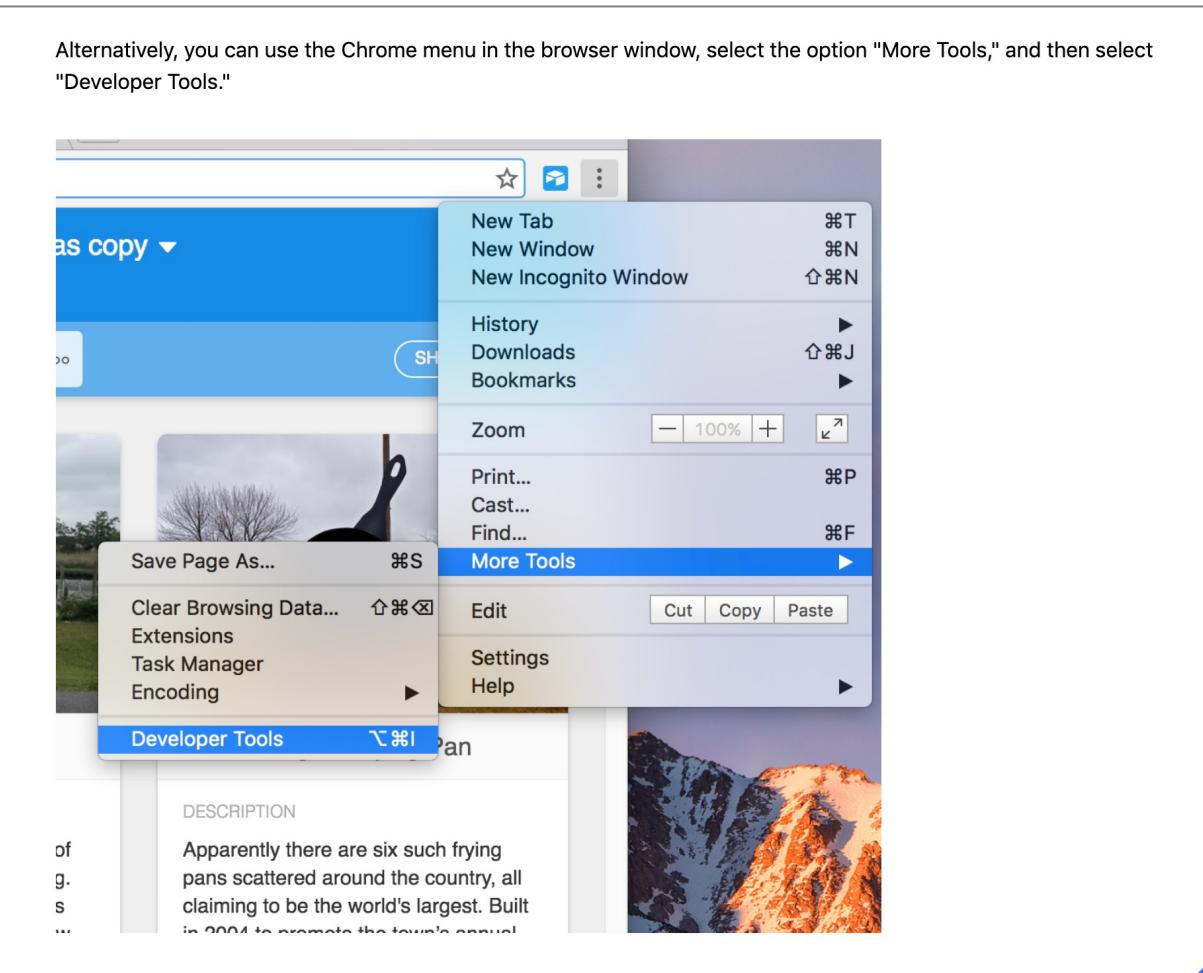
Debugging

Access the Console in your browser



Chrome

To open the developer console window on Chrome, use the keyboard shortcut **Cmd Shift J** (on Windows) or **Cmd Option J** (on Mac).



<https://support.airtable.com/hc/en-us/articles/232313848-How-to-open-the-developer-console>

Variables

Tells JavaScript this
is a variable

Variable name (Some are reserved).
Variable names in lowercase

```
var message = "Hello";  
console.log(message);
```

Outputs the contents of
the variable *message*

Puts the word *Hello* into
the variable *message*

Numbers don't have quotes

Puts the number 29 into
the variable age

```
var age = 29;  
console.log(age);
```

JavaScript has one number type

```
var age = 29;
```

It can hold integers

```
var awidth = 21.65;
```

It can hold floats or decimals

```
var cash = 15e5;
```

It can hold large number with scientific notation

```
var numberOne = 1;  
var numberTwo = 2.00;  
var numberThree="3";  
var numberFour ="4";  
console.log(numberOne + numberTwo);  
console.log(numberThree + numberFour);
```

3

34

“JavaScript has dynamic data types, This means the same variable can be used as different types.”

Example

```
var x;           // Now x is undefined
var x = 5;       // Now x is a Number
var x = "John"; // Now x is a String
```

The Boolean data type can only represent two values: true or false.

```
var correct = true;  
var wrong = false;
```

http://www.w3schools.com/jsref/jsref_obj_boolean.asp



Naming Variables



letters and numbers only
but cannot start with a number

Also allowed

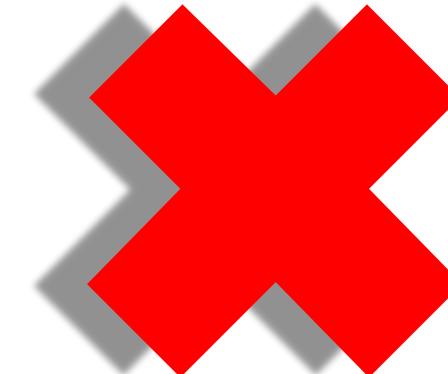
\$ (dollar)

_ (underscore)

```
var employee  
var employee001  
var _employee  
var $employee
```



```
var 1employee  
var employee 01  
var #employee  
var employee.name
```



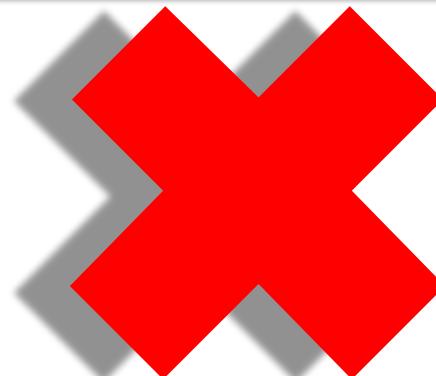
Pick variable names that
mean something in the
context of your program !

```
var employeeID  
var employeeName  
var employee_id  
var employee_name
```



lowercase first letter then following words uppercase letter or
use an underscore to separate words

```
var avariable  
var rover  
var gfhdsfgj
```



Variable names - summary

1. Must begin with a letter, \$ or _ Can't start with a number.
2. Can contain letters, numbers, \$ and _ Can't contain a hyphen (-) or full stop (.)
3. Cannot use reserved words as they 'do something' in the language – full list:

http://www.w3schools.com/js/js_reserved.asp

4. **Variable names are case sensitive** 'dog' and Dog' are different.
5. Use names that describe the information the variable stores.
6. If using more than one word use a Capital letter for each word after the first ([camelCase](#) – camelCaps) or use underscores (not hyphens) to join words.

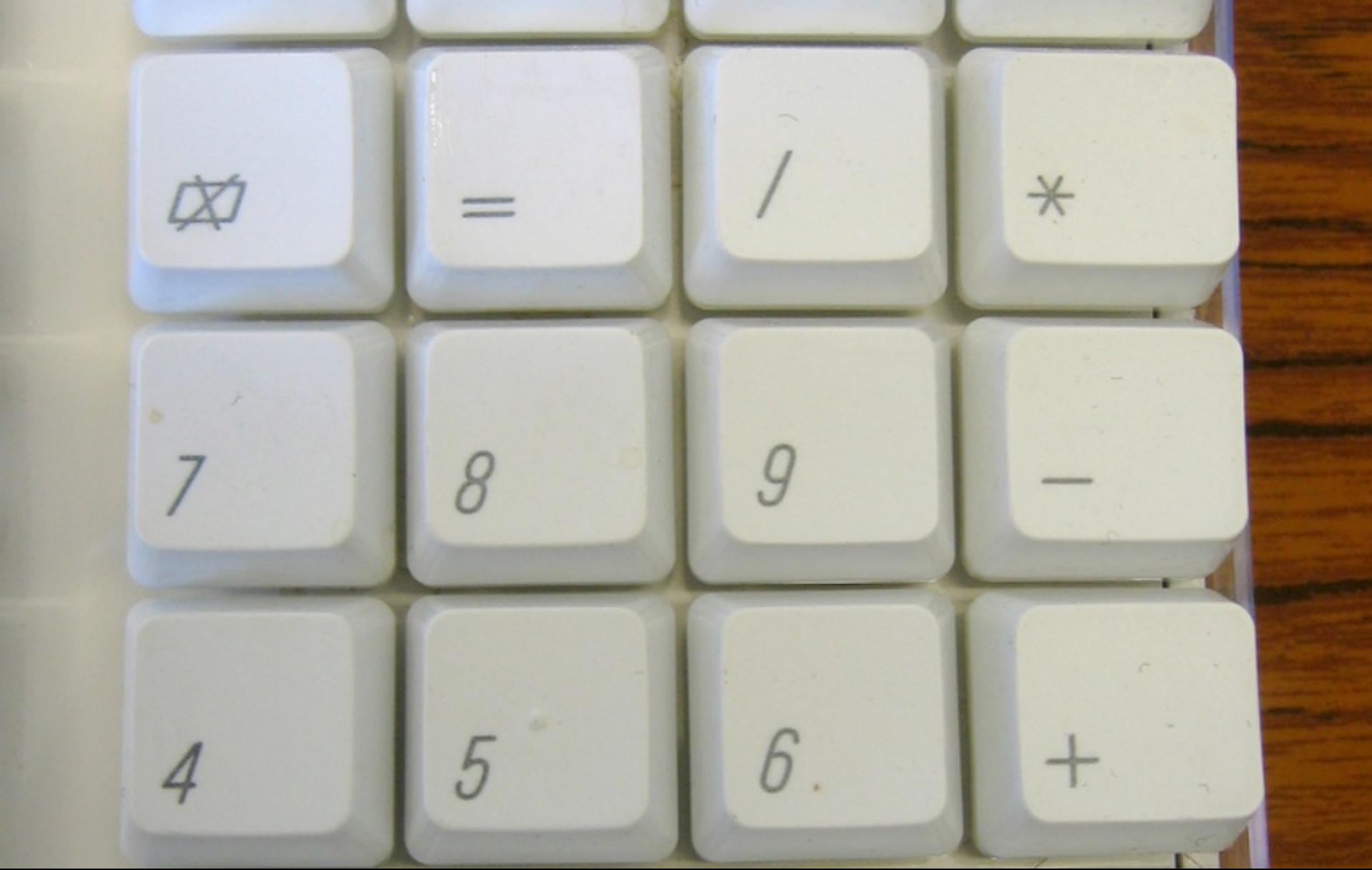
Concatenation

“Concatenation (from Latin concatenare, to link together) is taking two or more separately located things and placing them side-by-side next to each other so that they can now be treated as one thing.

In computer programming two or more character strings can be concatenated for the purpose of saving space or so that they can be addressed as a single item.”

In JavaScript you use the **+** symbol

```
“string1” + “string2” + variable3 + “ ”
```



Operators

JavaScript Arithmetic Operators

Arithmetic operators are used to perform arithmetic between variables and/or values.

Same as PHP mostly

Operator	Description	Example	Result
+	Addition	x= y+2	x=7
-	Subtraction	x= y-2	x=3
*	Multiplication	x= y*2	x=10
/	Division	x= y/2	x=2.5
%	Modulus (division remainder)	x= y%2	x=1
++	Increment	X= y++	x=6
--	Decrement	X= y--	x=4

JavaScript Assignment Operators

Assignment operators are used to assign values to JavaScript variables.

Same as PHP mostly

Operator	Example	Same As	Result
=	x=y		x=5
+=	x+=y	x=x+y	x=15
-=	x-=y	x=x-y	x=5
=	x=y	x=x*y	x=50
/=	x/=y	x=x/y	x=2
%=	x%=y	x=x%y	x=0

Comparison Operators

Comparison operators are used in logical statements to determine equality or difference between variables or values.

Same as PHP mostly

Operator	Description	Example	
<code>==</code>	is equal to	<code>x==8</code>	false
<code>===</code>	is exactly equal to (value and type)	<code>x === 5</code> <code>x === "5"</code>	true false
<code>!=</code>	is not equal	<code>x != 8</code>	true
<code>></code>	is greater than	<code>x > 8</code>	false
<code><</code>	is less than	<code>x < 8</code>	true
<code>>=</code>	is greater than or equal to	<code>x >= 8</code>	false
<code><=</code>	is less than or equal to	<code>x <= 8</code>	true

Logical Operators

Logical operators are used to determine the logic between variables or values.

Same as PHP mostly

Operator	Description	Example	
&&	and	$(x < 10 \&\& y > 1)$	true
	or	$(x==5 \mid\mid y==5)$	false
!	not	$!(x==y)$	true

Comments

//

.....



```
<script>
    var i = 2;    var j = 4;
    // multiply two numbers
    var answer = i * j;
    // output answer
    console.log(answer);
</script>
</body>
</html>
```

