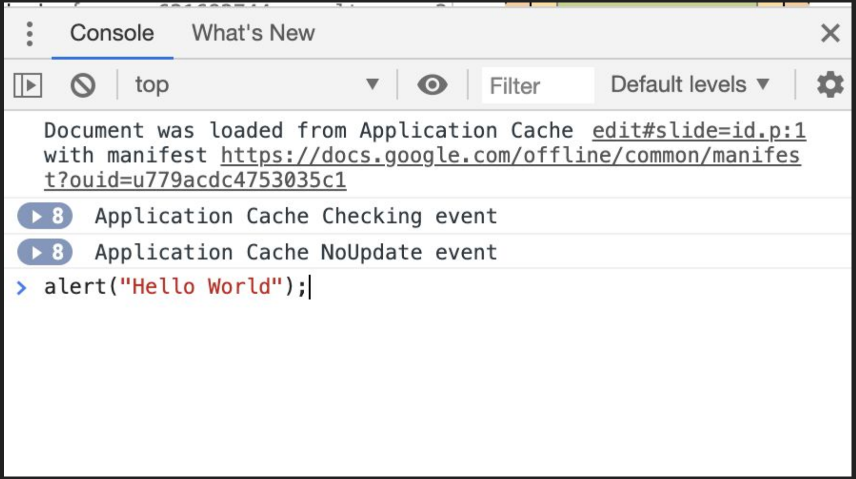
Intro to JavaScript

Why Study JavaScript?

- JavaScript is one of the **3 languages** all web developers **must** learn:
  - **HTML** to define the content of web pages
  - **CSS** to specify the layout of web pages.
  - **JavaScript** to program the behavior of web pages.
- Web pages are not the only place where JavaScript is used. Many desktop and server programs use JavaScript. Node.js is the best known. Some databases, like MongoDB and CouchDB, also use JavaScript as their programming language.

JavaScript Console

- The JavaScript Console is inside the developers tools.
- When on a web page, press F12 to open developer tools. Click console.
- You can write JS code inside there to interact with a web page.
- For example, typing alert(“Hello World!”); inside of the JS console will bring up an alert on the web page.



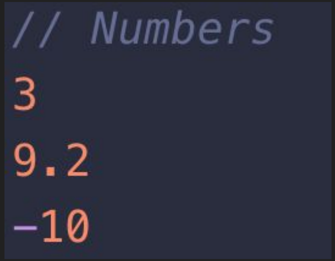
```
Document was loaded from Application Cache  edit#slide=id.p:1
with manifest https://docs.google.com/offline/common/manifes
t?ouid=u779acdc4753035c1
8  Application Cache Checking event
8  Application Cache NoUpdate event
> alert("Hello World");
```

JavaScript Primitive Data Types

- One of the big ideas at the core of every programming language is the languages ability to differentiate between different categories of data.
- For instance, a language could differentiate between a number and a word or it could differentiate between a whole number and a fractional number, or a positive number and a negative number.
- They vary from language to language.
- In Javascript, there are five primitive data types
  - Numbers
  - Strings
  - Booleans
  - Null
  - Undefined

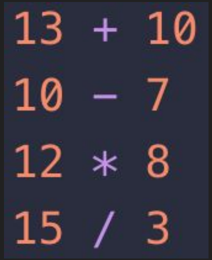
Numbers

- JavaScript doesn’t care if a number is whole, fractional, or negative. They’re treated all as just number.



```
// Numbers
3
9.2
-10
```

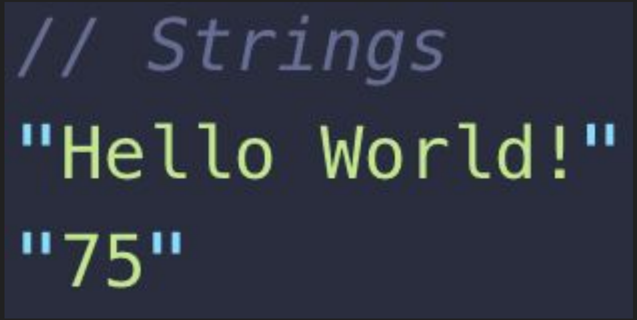
- You can also do math with numbers by using addition (+), subtraction (-), multiplication (*), and division (/).



```
13 + 10
10 - 7
12 * 8
15 / 3
```

Strings

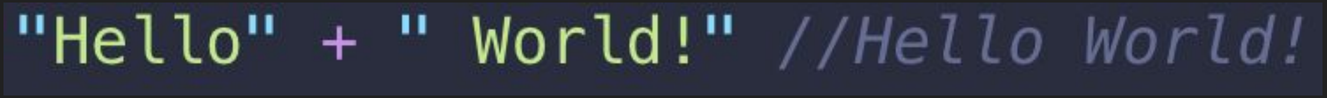
- Strings are basically text.
- The important thing is that they’re inside of quotes.
- You can have more than one word inside of a String.
- Numbers inside of quotes are considered Strings.



```
// Strings
"Hello World!"
"75"
```

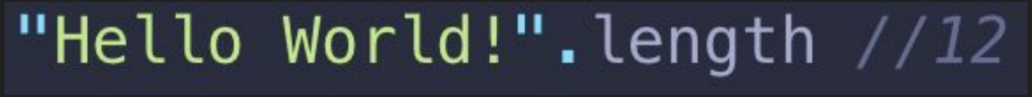
Strings cont.

- Can use Single or Double quotes
- Can concatenate strings by using the + sign.



```
"Hello" + " World!" //Hello World!
```

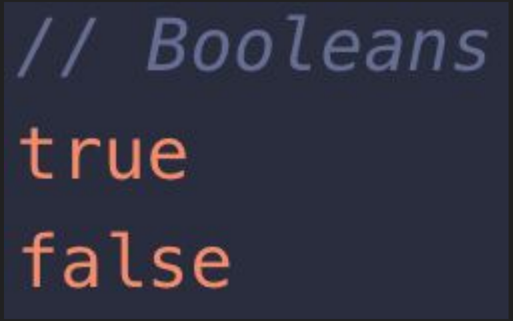
- Strings have a length property to find the length of the string.



```
"Hello World!".length //12
```

Booleans

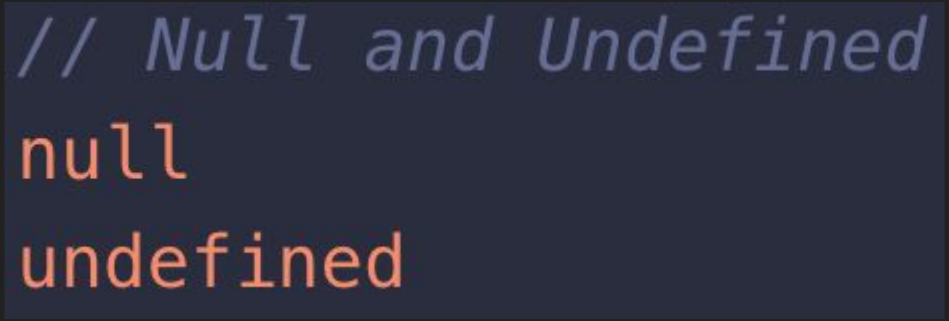
- Booleans only have two options: True or False
- There’s no quotes or numbers. Just the word True or the word False and that’s it.
- These will make more sense later in the course.



```
// Booleans
true
false
```

Null & Undefined

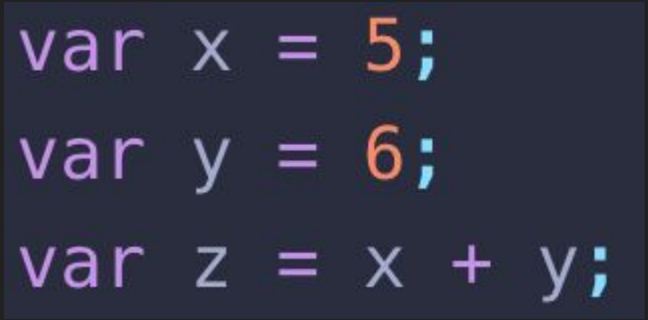
- Null and Undefined are actually values.
- For example, if you try using a variable that has not been set then you’ll receive the value of undefined.
- These will also make more sense later.



```
// Null and Undefined
null
undefined
```

JavaScript Variables

- JavaScript variables are containers for storing data value.
- In this example, x, y, and z are variables.

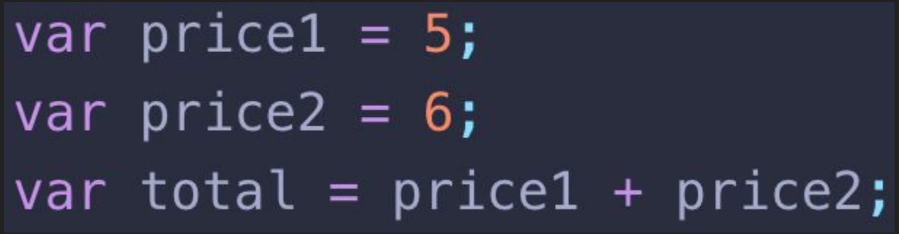


```
var x = 5;
var y = 6;
var z = x + y;
```

- From this example: x stores the value 5, y stores the value 6, and z stores the value 11.

JavaScript Variables cont.

- In this example, price1, price2, and total are variables.



```
var price1 = 5;
var price2 = 6;
var total = price1 + price2;
```

- In programming, just like in algebra, we use variables (like price1) to hold values.
- In programming, just like in algebra, we use variables in expressions

total = price1 + price2;

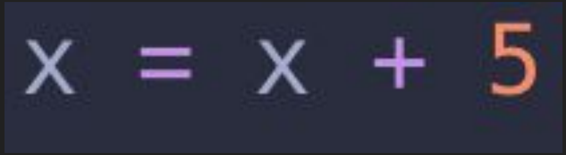
- From this example, you can calculate the total to be 11.

JavaScript Variables - Identifiers

- All JavaScript **variables** must be **identified** with **unique names**.
- These unique names are called **identifiers.**
- Identifiers can be short names (like x and y) or more descriptive names (age, sum, totalVolume).
- The general rules for constructing names for variables (unique identifiers) are:
    - Names can contain letters, digits, underscores, and dollar signs.
    - Names must begin with a letter.
    - Names can also begin with $ and _
    - Names are case sensitive (y and Y are different variables)
    - Reserved words (like JavaScript keywords) cannot be used as names.

JavaScript Variables - The Assignment Operator

- In JavaScript, the equal sign (=) is an “assignment” operator, not an “equal to” operator.
- This is different from algebra. The following does not make sense in algebra:

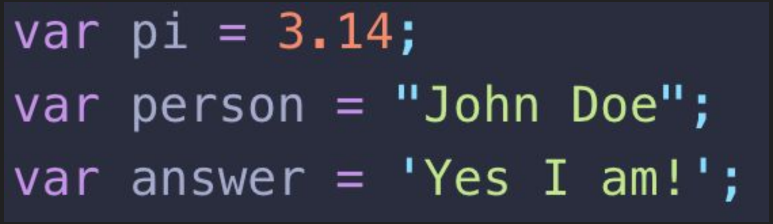


```
x = x + 5
```

- In JavaScript, it makes perfect sense: it assigns the value of x + 5 to x.
- The “equal to” operator is written like == in JavaScript.

JavaScript Variables - Data Types

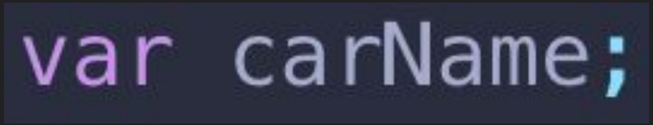
- JavaScript variables can hold numbers like 100 and text values like “John Doe”.
- In programming, text values are called text strings.
- JavaScript can handle many types of data, but for now, just think of numbers and strings.
- Strings are written inside double or single quotes. Numbers are written without quotes.
- If you put a number in quotes, it will be treated as a text string.



```
var pi = 3.14;
var person = "John Doe";
var answer = 'Yes I am!';
```

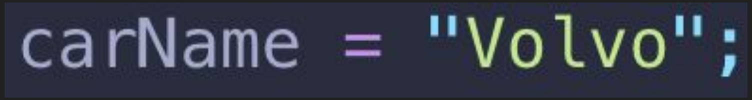
JavaScript Variables - Declaring Variables

- Creating a variable in JavaScript is called “declaring” a variable.
- You declare a JavaScript variable with the var keyword:



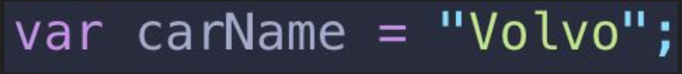
```
var carName;
```

- After the declaration, the variable has no value (technically it has the value of undefined)
- To assign a value to the variable, use the equal sign:



```
carName = "Volvo";
```

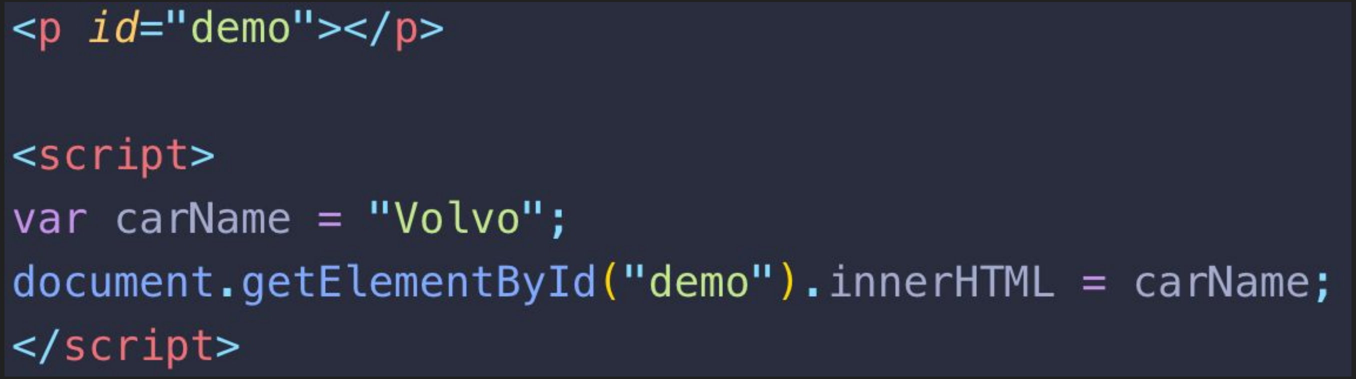
- You can also assign a value to the variable when you declare it:



```
var carName = "Volvo";
```

JavaScript Variables - Declaring Variables cont.

- In the example below, we create a variable called carName and assign the value “Volvo” to it.
- Then we “output” the value inside an HTML paragraph with id=”demo”

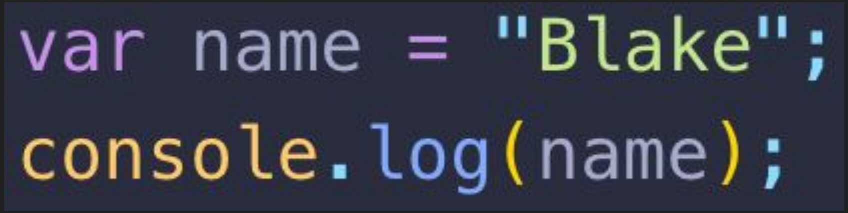


```
<p id="demo"></p>

<script>
var carName = "Volvo";
document.getElementById("demo").innerHTML = carName;
</script>
```

JavaScript Variables: Calling the Variable

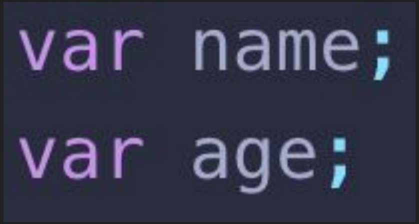
- Recall the stored value by calling the variable name.



```
var name = "Blake";
console.log(name);
```

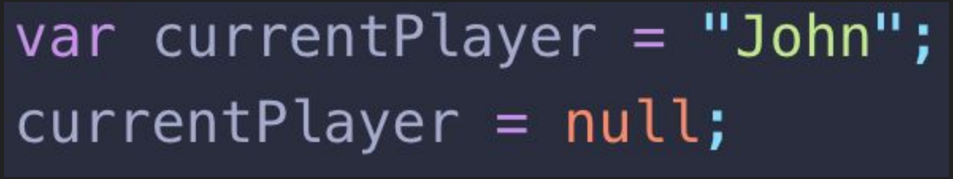
JavaScript Null & Undefined

- Variables that are declared but not initialized are undefined.
- The following variables are undefined.



```
var name;
var age;
```

- Null is “explicitly nothing”



```
var currentPlayer = "John";
currentPlayer = null;
```

Useful Built-In Methods for Next Assignments

- The following built-in methods will help you for the next few assignments:

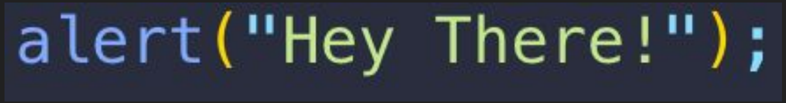
`console.log`

`alert`

`prompt`

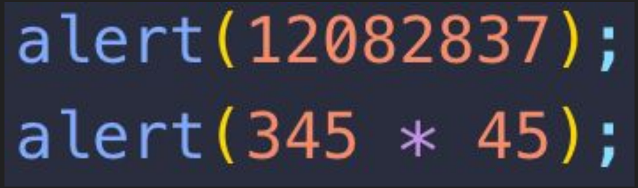
Built-in Methods - Alert

- When used, it will pop open a message box to alert the user.
- The alert syntax looks like this:



```
alert("Hey There!");
```

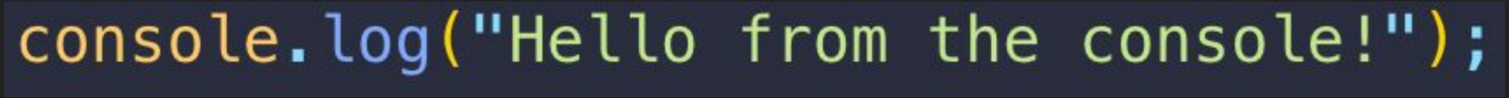
- You can have the alert say anything you'd like.
- You can also use numbers and math.



```
alert(12082837);
alert(345 * 45);
```

Built-in Methods - Console.log

- Similar to alert except less intrusive.
- Instead of displaying inside of a message box, it will be displayed inside of the javascript console.
- If someone doesn't have the console open then they will never see it print.
- It looks like the following.

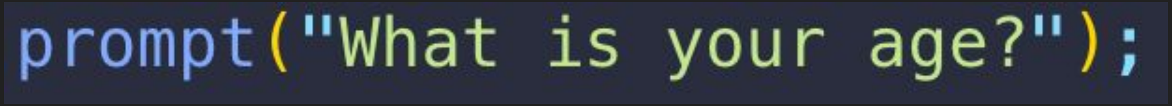


```
console.log("Hello from the console!");
```

- This will print "Hello from the console!" inside our JavaScript console.
- Usually used for our own purposes. You would never expect your user to actually go to the console.

Built-in Methods: Prompt

- Allows us to get input from a user.
- It looks like this:

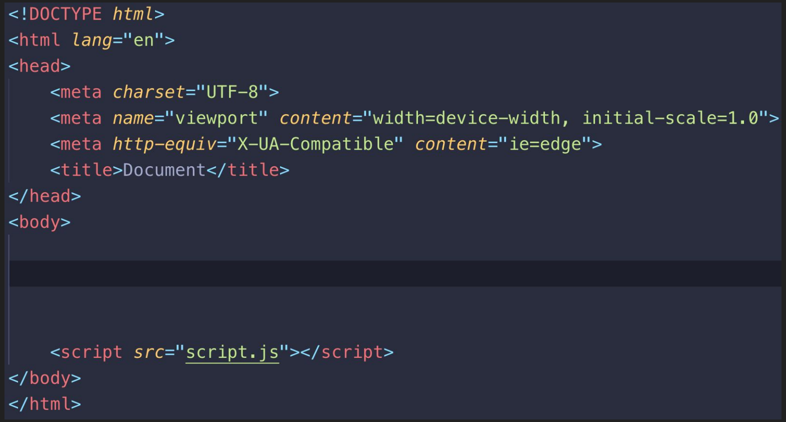


```
prompt("What is your age?");
```

- The second part of prompt is that you can store it in a variable to use at a later time.
- You would not use this in a real production web site but will be useful for the next few exercises.

Writing JavaScript in a Separate File

- Open a new file in your code editor.
- Save it as a .js file inside the same directory as your HTML file. It’s common for it to be named script.js or index.js
- Inside of your HTML file, right before the closing <body> tag., create a <script> tag with the src attribute.
- The value of the src attribute should be the name of the JS file you just created.



```html
<!DOCTYPE html>
<html lang="en">
<head>
    <meta charset="UTF-8">
    <meta name="viewport" content="width=device-width, initial-scale=1.0">
    <meta http-equiv="X-UA-Compatible" content="ie=edge">
    <title>Document</title>
</head>
<body>




    <script src="script.js"></script>
</body>
</html>
```

Writing JavaScript in a separate file cont.

- The index.js file is now where you’ll be writing your JS code.

Assignment 6.1: JS Stalker Exercise

Variables, Strings, Prompt, and Console.log

- Ask for the user’s first name.
- Ask for the user’s last name.
- Ask for the user’s age.
- Print out the user’s full name in a sentence.
- Print out the user’s age in a sentence.

Click here for example.

Assignment 6.2: Age Calculator

- Enter your age and find out how many days you’ve been alive.

Click here for example.