

JavaScript I

INFO 253A: Front End Web Architecture

Kay Ashaolu

JavaScript!

- A fully featured programming language
- Used to be only for the browser, used along with HTML and CSS
- Now used on web servers and for Application Program Interfaces (APIs)
- Great example of composability!

Why learn JavaScript first?

- Truthfully, front end development is heavily dependent on JavaScript
- If a site doesn't need JavaScript, chances are that there's some low code tool that can produce the site without needing to know code
- Demonstrates the software engineering skills needed now to become a frontend engineer

JavaScript: Powering your pages

- JavaScript brings a website to life
- No need for a server to execute by the browser
- Pop-ups, drop down navigation
- Form validation, animation
- Application development

JavaScript: Powering your Servers

- JavaScript is used to power the servers that launch websites! (NodeJS)
- No longer considered a "play" language

JavaScript: Powering your APIs

- API: Application Program Interface
 - **We will not cover this in this class**
- JavaScript is not just reserved for building web pages
- JavaScript can power pure back end functionality

We will be using JavaScript

- To build our website
- To power interactivity between the user and the website
- To communicate to external servers using APIs

So let's learn some JavaScript

- We are going to use our browsers to run our JavaScript
- This will be a little weird at first, but remember that the browser is the "executor" of the code that you write on the frontend
- Once we learn HTML/CSS, it will be more apparent how all of these technologies work together

What this is not

- This is **not** an exhaustive review of the JavaScript programming language
- One goal is to get you familiar with the features of the language
- Another goal is to give you the opportunity to ask questions
- The better you know pure JavaScript, the more you'll be able to build rich web applications
- Again, even though we are learning React in this course, the better you know the JavaScript language, the better you will understand React

Before we start: code editor

- You will need an editor that can save files in plain text
- Better if you find something that can highlight code to make it much more readable
- My tool of choice (currently): **Visual Studio Code**

How to run JavaScript

- Create an HTML file called "index.html"
- Add the following text in the file:

```
1 <html>
2     <head>
3         <title>Learning JavaScript</title>
4     </head>
5     <body>
6         <script src="script.js"></script>
7     </body>
8 </html>
```

What did we just do?

- You just created an HTML file that is blank, but tells the browser to execute "script.js", located in the same location as this html file
- Remember the HTML file is the first content to get to the browser to be parsed to build the DOM.
- Next, because the "script" tag is present, it signals to the browser to retrieve "script.js", so another request is made to retrieve that
- Once it is received, "script.js" is executed line by line

Variables? Data Types?

- A variable is an empty space holder for a future value
- There are a few types of values that JavaScript uses
 - Numbers and Booleans (true or false)
 - Strings or a sequence of characters
 - Functions
 - Arrays (lists of variables) and Objects

Variables

- Declare with let to scope correctly
- Weak, dynamic typing

```
1 let a = 3;  
2 let b = 5;  
3 let c = a + b;
```

Strings

```
1 let a = "hello";  
2 let b = "world!";  
3 let c = a + b;
```

Arrays (Lists)

```
1 let list = [2,3, "KAY"];  
2 alert(list[2]);
```


Objects

```
1 let titles = {
2   info253: 'Web Arch',
3   info256: 'Applied NLP'
4 };
5 alert(titles.info253);
6
7 let schools = {
8   berkeley: {info253: 'Web Arch'},
9   stanford: {cs101: 'Intro CS'}
10 };
11 alert(schools.stanford.cs101);
```

But Note

- We will go over all of these types through writing functions
- You don't have to do this, but you **should** do this
- It is extremely important to be comfortable with writing and using functions

Functions

So what is a function

- Think of a function as a black box, where you put things in (inputs) and out comes a result (output)
- It's a black box because you don't necessarily need to know how it's converting the inputs
- The parameters are the inputs, the return statement defines the output

Function Syntax

```
1 let add = function(a, b) {return a + b;}
2 let c = add(2, 7)
3 /* c is now 9 */
4
5 let arithmetic = {add: add, subtract: function(a, b) {return a -
6 let d = arithmetic.subtract(11, 4);
7 let e = arithmetic.add(0, 3);
8 /* d is now 7; e is now 3; */
```

Our First Function

```
1 function helloWorld(first_name, last_name) {  
2     let message = "Hello World " + first_name + " " + last_name;  
3     return message;  
4 }  
5  
6 let first = prompt("Enter your first name");  
7 let last = prompt("Enter your last name");  
8 let output = helloWorld(first, last);  
9 alert(output);
```

What does this function do?

What did that function do?

- Declared a function that we put in two values or inputs: (first name and last name)
- The function declared a variable (message) that contains the string "Hello World [your first name] [your last name]"
- After it's done, it will produce one value or output: (the message) and return it
- We ask the user for their first and last name, and print out the result from the function

More to come

- Conditionals
- For and While loops
- Arrays
- Objects

For the non programmers here stay with me, learn this and it will empower you throughout this course and beyond

Questions?