

Intro to JavaScript and jQuery

James Xue

COMS W4170

Thurs. Sept. 20, 2018

1

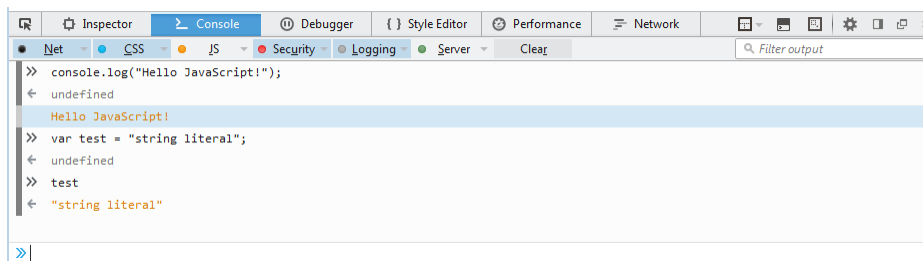
Agenda

- JavaScript
- jQuery
- Build a weather app

2

JavaScript

- Programming language to make responsive web pages
- You can write JavaScript in a browser's developer tools!



3

HTML + CSS + JavaScript

index.html

```
1 <!DOCTYPE html>
2 <html>
3 <head>
4   <link rel="stylesheet" href="styles.css">
5   <script src="script.js"></script>
6 </head>
7 <body>
8   <h1 id="title">Hello World!</h1>
9
10
11 </body>
12 </html>
```

styles.css

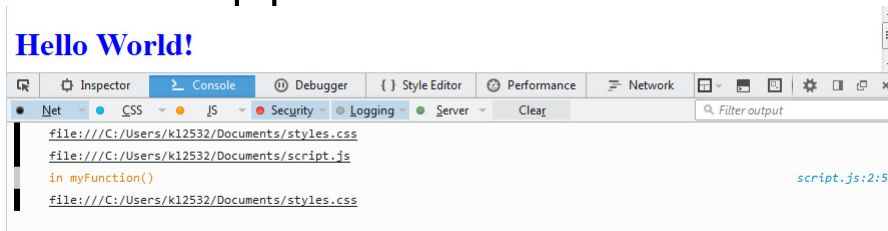
```
1 h1{
2   color: blue;
3 }
4
```

script.js

```
1 function myFunction() {
2   console.log("in myFunction()");
3   document.getElementById("title").style.color = "red";
4   var pElem = document.createElement("p");
5   pElem.innerHTML = "Hello Javascript!";
6   document.body.appendChild(pElem);
7 }
8
9 myFunction();
```

4

What happened?

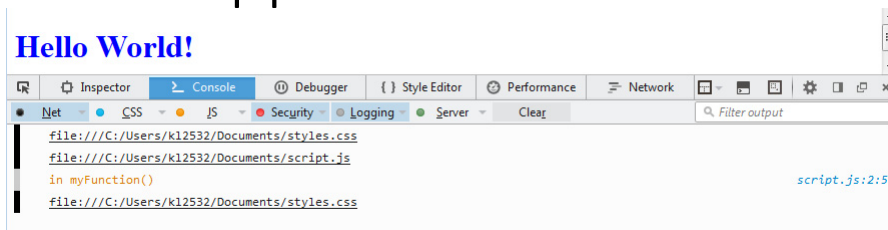


index.html

```
1 <!DOCTYPE html>
2 <html>
3 <head>
4   <link rel="stylesheet" href="styles.css">
5   <script src="script.js"></script>
6 </head>
7 <body>
8
9   <h1 id="title">Hello World!</h1>
10
11 </body>
12 </html>
```

5

What happened?



index.html

Order matters!

```
1 <!DOCTYPE html>
2 <html>
3 <head>
4   <link rel="stylesheet" href="styles.css">
5   <script src="script.js"></script>
6 </head>
7 <body>
8
9   <h1 id="title">Hello World!</h1>
10
11 </body>
12 </html>
```

6

HTML + CSS + JavaScript

index.html

```
1 <!DOCTYPE html>
2 <html>
3 <head>
4   <link rel="stylesheet" href="styles.css">
5 </head>
6 <body>
7   <h1 id="title">Hello World!</h1>
8   <script src="script.js"></script>
9 </body>
10 </html>
```

styles.css

```
1 h1{
2   color: blue;
3 }
4
```

script.js

```
1 function myFunction() {
2   console.log("in myFunction()");
3   document.getElementById("title").style.color = "red";
4   var pElem = document.createElement("p");
5   pElem.innerHTML = "Hello Javascript!";
6   document.body.appendChild(pElem);
7 }
8
9 myFunction();
```

7

jQuery

A **JavaScript library** that simplifies

- HTML document traversal and manipulation
- Event handling
- Animation
- AJAX (Asynchronous JavaScript and XML)

with an easy-to-use API

<http://api.jquery.com/>

8

Adding jQuery

- Within your directory

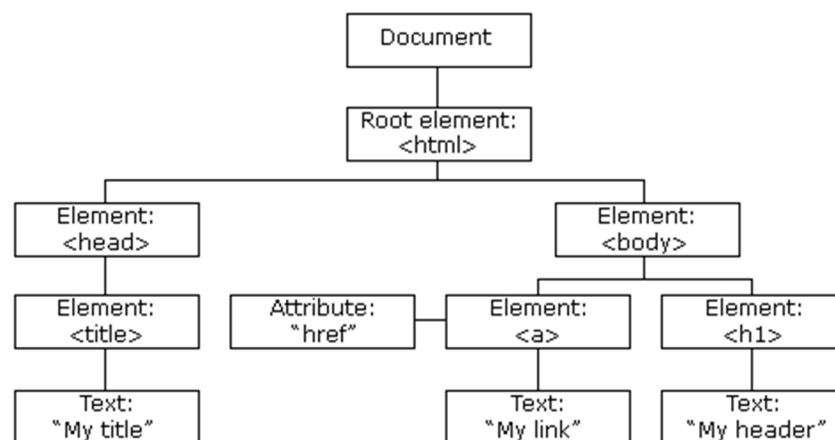
```
1 <head>
2 <script src="jquery-3.1.0.js"></script>
3 </head>
4
```

- With a CDN (content delivery network)

```
1 <head>
2 <script src="https://ajax.googleapis.com/ajax/libs/jquery/3.1.0/jquery.min.js"></script>
3 </head>
4
```

9

DOM Traversal and Manipulation



10

Hello World (revisited)

Let's rewrite this using jQuery!

Hint: check out <http://api.jquery.com/>

index.html

```
1 <!DOCTYPE html>
2 <html>
3 <head>
4   <link rel="stylesheet" href="styles.css">
5   <script src="script.js"></script>
6 </head>
7 <body>
8
9   <h1 id="title">Hello World!</h1>
10
11 </body>
12 </html>
```

styles.css

```
1 h1{
2   color: blue;
3 }
4
```

script.js

```
1 function myFunction() {
2   console.log("in myFunction()");
3   document.getElementById("title").style.color = "red";
4   var pElem = document.createElement("p");
5   pElem.innerHTML = "Hello Javascript!";
6   document.body.appendChild(pElem);
7 }
8
9 myFunction();
```

11

Hello World (revisited)

.css(propertyName, value)

Returns: [jQuery](#)

Description: Set one or more CSS properties for the set of matched elements.

🔗 .css(propertyName, value)

version added: 1.0

propertyName

Type: [String](#)

A CSS property name.

value

Type: [String](#) or [Number](#)

A value to set for the property.

(old) script.js

```
1 function myFunction() {
2   console.log("in myFunction()");
3   document.getElementById("title").style.color = "red";
4   var pElem = document.createElement("p");
5   pElem.innerHTML = "Hello Javascript!";
6   document.body.appendChild(pElem);
7 }
8
9 myFunction();
```

.appendTo(target)

Returns: [jQuery](#)

Description: Insert every element in the set of matched elements to the end of the target.

🔗 .appendTo(target)

version added: 1.0

target

Type: [Selector](#) or [htmlString](#) or [Element](#) or [Array](#) or [jQuery](#)

A selector, element, HTML string, array of elements, or jQuery object; the matched set of elements will be inserted at the end of the element(s) specified by this parameter.

(with jQuery) script.js

```
1 function myFunction() {
2   console.log("in myFunction()");
3   $('#title').css('color', 'red');
4   $('<p>Hello Javascript!</p>').appendTo( "body" );
5 }
6
7 myFunction();
8
```

12

Event Handling

```
1 <!DOCTYPE html>
2 <html>
3 <body>
4
5     <button onclick="myFunction()">Click Me</button>
6     <p id="demo"></p>
7
8 <script>
9 function myFunction() {
10     document.getElementById("demo").innerHTML = "Hello World";
11 }
12 </script>
13
14 </body>
15 </html>
16
```

13

Event Handling

`$(document).ready(function())`

.ready(handler)

Returns: [jQuery](#)

Description: Specify a function to execute when the DOM is fully loaded.

 **.ready(handler)**

version added: 1.0

handler

Type: [Function\(\)](#)

A function to execute after the DOM is ready.

14

Hello World (with Event Handling)

How can we use the `$(document).ready(function())`?

index.html

```
1 <!DOCTYPE html>
2 <html>
3 <head>
4   <link rel="stylesheet" href="styles.css">
5   <script src="https://ajax.googleapis.com/ajax/libs/jquery/3.1.0/jquery.min.js"></script>
6   <script src="script.js"></script>
7 </head>
8 <body>
9
10  <h1 id="title">Hello World!</h1>
11
12 </body>
13 </html>
```

script.js

```
1 function myFunction() {
2   console.log("in myFunction()");
3   $('#title').css('color', 'red');
4   $("<p>Hello Javascript!</p>" ).appendTo( "body" );
5 }
```

15

AJAX

Aynchronous JavaScript and XML

Two major features:

1. Make requests to server without reloading the page
2. Receive and work with data from the server

```
1 $.ajax({
2   url: "/api/getWeather",
3   data: {
4     zipcode: 97201
5   },
6   success: function( result ) {
7     $( "#weather-temp" ).html( "<strong>" + result + "</strong> degrees" );
8   }
9 });
10
```

16

Objects

```
var person = {  
  firstName:"John",  
  lastName:"Doe",  
  age:20,  
  eyeColor:"brown"  
};
```

- What's the output of the following lines?
 >> person.firstName
 >> person.lastName
 >> person["age"]

Objects are mutable

- How would we change the person's eyeColor to "blue"?

17

Objects

```
var person = {  
  firstName:"John",  
  lastName:"Doe",  
  age:20,  
  eyeColor:"brown"  
};
```

- What's the output of the following lines?
 >> person.firstName
 >> person.lastName
 >> person["age"]

Objects are mutable

- How would we change the person's eyeColor to "blue"?
 >> person.eyeColor = "blue"

18

Let's build a weather app!

What's the weather now?
Location:
Clouds

API Documentation

<http://openweathermap.org/current>

Get an API key

https://home.openweathermap.org/users/sign_in

19

What's the weather now?

Location:

Clouds

API Documentation

<http://openweathermap.org/current>

Get an API key

https://home.openweathermap.org/users/sign_in

index.html

```
1 <!DOCTYPE html>
2 <html>
3 <head>
4   <link rel="stylesheet" href="styles.css">
5   <script src="https://ajax.googleapis.com/ajax/libs/jquery/3.1.0/jquery.min.js"></script>
6 </head>
7 <body>
8
9   <h1 id="title">What's the weather now?</h1>
10  <form id="search-form">
11    <label for="location">Location:</label>
12    <input type="text" id="location">
13    <button type="submit">Submit</button>
14  </form>
15  <p id="weather"></p>
16
17  <script src="script.js"></script>
18
19 </body>
20 </html>
21
```

20