

Intro to JavaScript

JavaScript: the DOM // intro to programming in JavaScript

What is JavaScript?

1. Initially created as a browser-only language (where the browser is the interpreter)
2. Does all of the same things as other 'real' programming languages do (Turing complete).
3. An interpreted language (like Python) – no compiler
4. Now used in a variety of different contexts (it can be run anywhere that a JavaScript engine is installed)
5. Many languages “transpile” to JavaScript (e.g. Dart, TypeScript, CoffeeScript), meaning that you can program in another language, and then run a ‘transpiler’ to convert your code to JavaScript so that your browser can understand it.

“Client-Side” JavaScript

What is JavaScript’s job within the **browser**?

1. Can respond to user events
2. Can manipulate the DOM by adding/removing/modifying/deleting:
 - a. elements
 - b. attributes
 - c. style properties
 - d. content
3. Can pull down resources from any server (for which it is authorized) and inject content into the DOM
4. Can post content from the browser to a server
5. Can manipulate data (which JS sees as lists of objects)
6. Can do all the computations that an ordinary language can do (but heavy computations are typically delegated to server processes)

The Possibilities and Limitations of “Client-Side” JavaScript

1. Can JavaScript from your browser access your file system?
 - a. No – that’d be a huge security risk. Why?
2. Can JavaScript access your camera, microphone, or current location?
 - a. Only with your permission
3. Can JavaScript store information about you that the site can access later?
 - a. Yes – Cookies and localStorage (you can view these via the browser inspector)
4. Can JavaScript transmit information about your browsing interactions back to the server?
 - a. Yes – b/c JavaScript can (1) “listen” to any user event (mousemove, click, drag, scroll, etc.), and (2) post information to a server, it can collect and transmit fine-grained information about your browsing behavior
5. Can any website access the information that another website has gathered about you?
 - a. It’s possible, but not through client-side JavaScript (see next slide)

Browser's Local Storage / Cookies

Tracking Company

Third Party Cookies

Google

Information Services Website

- Google-assigned user id
- Other metadata

- Third Party user id

Amazon

Online shopping Website

- Amazon-assigned user id
- Other metadata

- Third Party user id

X

“Server-Side” JavaScript (Not in this Class, but FYI)

What is JavaScript’s job on a server (e.g. Node.js)?

1. Node.js lets developers use JavaScript on a server – to generate web pages before the page is sent to the user's web browser (developed in 2009)
2. Analogous to the role that other web frameworks fill – Java’s Spring, Django (Python), Flask (Python), Ruby on Rails, etc.
3. Can connect to databases, communicate with other server processes, and interoperate with other languages via sockets and processes on a server.
4. V8 is the JavaScript compiler (Open Sourced via Chrome). Compiles JavaScript directly to native machine code before executing it

Client-Side JavaScript Schedule

Week 5 (this week):

1. Intro to DOM manipulation.
2. Intro to computer programming with JavaScript:
 - a. variables, expressions, and statements
 - b. data types

Week 6

1. Functions and Events
2. Templating

Week 7

1. AJAX (interacting with Servers)
2. Working with data and APIs

JavaScript Readings / Videos

LinkedIn Learning — see website for playlists

Document Object Model (DOM)

[Reminder from Lecture 4]

A way of representing a document, like a web page, in a way that can be understood by a human and by a computer. **Javascript** can directly manipulate the DOM dynamically.

Selectors

Recall from CSS: selectors are ways of targeting elements in a web page so that we can apply styles to them.

Remember these...

```
body {  
    color: grey;  
}
```

```
h1, li {  
    text-transform: uppercase;  
    display: inline-block;  
    color: #999999;  
}  
  
.title-bar {  
    padding: 5px;  
    background-color: #EEEEEE;  
}  
  
#profile {  
    width: 100px;  
    float: left;  
    margin-right: 20px;  
}
```

```
...  
1  <body>  
2      <div class="title-bar">  
3          <h1>Welcome, Malik</h1>  
4            
5          <hr>  
6      </div>  
7      <div>Right  
8          <ul>  
9              <li>list item 1</li>  
10             <li>list item 2</li>  
11             <li>list item 3</li>  
12         </ul>  
13     </div>  
14 </body>  
...
```

```
body {  
    color: grey;  
}  
  
h1, li {  
    text-transform: uppercase;  
    display: inline-block;  
    color: #999999;  
}  
  
.title-bar {  
    padding: 5px;  
    background-color: #EEEEEE;  
}  
  
#profile {  
    width: 100px;  
    float: left;  
    margin-right: 20px;  
}
```

```
...  
1  <body>  
2      <div class="title-bar">  
3          <h1>Welcome, Malik</h1>  
4            
6          <hr>  
7      </div>  
8      <div>Right  
9          <ul>  
10             <li>list item 1</li>  
11             <li>list item 2</li>  
12             <li>list item 3</li>  
13         </ul>  
14     </div>  
...
```

```
body {  
    color: grey;  
}  
  
h1, li {  
    text-transform: uppercase;  
    display: inline-block;  
    color: #999999;  
}  
  
.title-bar {  
    padding: 5px;  
    background-color: #EEEEEE;  
}  
  
#profile {  
    width: 100px;  
    float: left;  
    margin-right: 20px;  
}
```

```
...  
1 <body>  
2     <div class="title-bar">  
3         <h1>Welcome, Malik</h1>  
4           
5         <hr>  
6     </div>  
7     <div>Right  
8         <ul>  
9             <li>list item 1</li>  
10            <li>list item 2</li>  
11            <li>list item 3</li>  
12        </ul>  
13    </div>  
14 </body>  
...
```

```
body {
    color: grey;
}

h1, li {
    text-transform: uppercase;
    display: inline-block;
    color: #999999;
}

.title-bar {
    padding: 5px;
    background-color: #EEEEEE;
}

#profile {
    width: 100px;
    float: left;
    margin-right: 20px;
}
```

```
...
1  <body>
2      <div class="title-bar">
3          <h1>Welcome, Malik</h1>
4          
6      </div>
7      <div>Right
8          <ul>
9              <li>list item 1</li>
10             <li>list item 2</li>
11             <li>list item 3</li>
12         </ul>
13     </div>
14 </body>
...
```

JavaScript also supports element targeting...

Recall from CSS: selectors are ways of targeting elements in a web page so that we can apply styles to them.

Method	Example	Returns
<code>getElementById()</code>	<code>document.getElementById("my_element")</code>	single element
<code>querySelector()</code>	<code>document.querySelector("#my_element")</code> <code>document.querySelector("p")</code> <code>document.querySelector(".my-announcements")</code>	single element
<code>querySelectorAll()</code>	<code>document.querySelectorAll("p")</code>	list of elements
<code>getElementsByTagName()</code>	<code>document.getElementsByTagName("div")</code>	list of elements
<code>getElementsByClassName()</code>	<code>document.getElementsByClassName(".panel")</code>	list of elements

[DEMO](#)

And once you target an element,
you can change it...

DOM Manipulation: Some (of the many) attributes you can set

These are but a few. You can set any element attribute using JavaScript

Attribute	Example	Elements
className	<code>document.querySelector("div").className = "panel";</code>	all
innerHTML	<code>document.querySelector("div").innerHTML = "hi!";</code>	all
src (for images)	<code>document.querySelector("img").src = "some_image_url"</code>	images only
href (for links)	<code>document.querySelector("a").href = "http://site.com";</code>	links only
...

DOM Manipulation: Style properties that you can set

These are but a few. You can set any style property using JavaScript

Property	Example
width	<code>document.querySelector("div").style.width = "200px";</code>
height	<code>document.querySelector("div").style.height = "200px";</code>
background color	<code>document.querySelector("div").style.backgroundColor = "hotpink";</code>
border width	<code>document.querySelector("div").style.borderWidth = "5px";</code>
padding	<code>document.querySelector("div").style.padding = "10px";</code>
display	<code>document.querySelector("div").style.display = "none";</code>
...	...

A quick look-ahead...

Quick Preview of Functions and Events

We will be going over both functions and events in JavaScript **next week**, however we're going to **start using them today**. You don't need to know how to write them, but you do need to know how to read them...

1. A function is an encapsulated grouping of programming statements that you can invoke on demand...
2. An event handler is a way that allows you to attach a function to an event

Function Example

A function is an encapsulated grouping of programming statements that you can invoke on demand...

```
const sayHello = () => {  
    document.querySelector('h1').style.color = 'white';  
    document.querySelector('h1').innerText = 'Hello!';  
};  
  
const sayGoodbye = () => {  
    document.querySelector('h1').innerText = 'Bye!';  
};  
  
sayHello();  
sayGoodbye();  
sayHello();  
sayGoodbye();
```

Event Handler Example

An event handler is a way that allows you to attach a function to an event. There are many different events that a browser allows you to hook into: click, mouseover, mouseout, drag, scroll, etc.

EXAMPLE:

```
<button onclick="sayHello()">click me</button>
```

Summary of the Process of DOM Manipulation

1. Target an element using one of the selector methods.
 - a. For now, let's use `document.querySelector()`
2. Specify what you want to change about the element:
 - a. A style property?
 - b. An attribute?
 - c. What goes inside of the element?
3. **Also note:** usually DOM manipulation happens inside of an event handler (which we'll be talking more about next week)!

Download Today's Code

Activity 1: Manipulating Style Properties

Open 01-style-property-demo

1. Modify each of the event handlers so that when the button is clicked, the body's background color changes to the corresponding color.
2. How would you switch the font of the h1 tag when the user clicks on the button?

Activity 2: Manipulating HTML Element Attributes

Open 02-attribute-demo

1. Modify each of the event handlers so that when the image's "src" attribute is set to a different animal image.
2. How would you also modify the paragraph text when each button is clicked?

Activity 3: Together

Open 03-all-of-the-above

- Update the body of the `changeColor()` function so that the panel turns to hotpink.
- Update the body of the `changeTitle()` function so that it changes the title of the webpage to "hi there!"
- Update the body of the `addImage()` function so that it adds an image of a cat to each panel
- update the body of the `clearDivs()` function so that it clears out the image of a wombat for each panel