

JavaScript and the Web

Jerry Cain
CS 106AJ
November 30, 2018
slides courtesy of Eric Roberts

The History of the World Wide Web

- The ideas behind the web are much older than the web itself.
 - In the early 20th century, the Belgian bibliographer Paul Otlet envisioned a universal catalogue that would provide access to all the world's information in an interconnected structure.
 - In 1945, the director of the wartime science program Vannevar Bush published an article entitled "As We May Think," which envisioned an electronic archive of linked information.
 - In the early 1960s, computer visionary Ted Nelson coined the terms *hyperlink* and *hypermedia*.
- The modern web was developed in 1989 by Tim Berners-Lee at CERN, the European particle physics laboratory in Geneva, Switzerland. Berners-Lee developed the first version of the *Hypertext Markup Language (HTML)*.
- Use of the web grew quickly after the release of Mosaic browser in 1993 and Netscape Navigator in 1994.

The Document Object Model

- When the browser reads a web page, it first translates the text of the web page into an internal data structure that is easier to manipulate under the control of a JavaScript program. That internal form is called the *Document Object Model*, or *DOM*.
- The DOM is structured into a hierarchy of data objects called *elements*, which usually correspond to a paired set of tags.
- The relationship between the HTML file and the internal representation is similar to the one between the external data file and the internal data structure in any data-driven program. The browser acts as a driver that translates the HTML into an internal form and then displays the corresponding page.
- Unfortunately, the DOM is poorly designed, giving rise to a structure that is difficult to understand. The best strategy is to learn only those parts of the DOM you need.

Writing Text to a <div> Element

- The strategy we'll use in today's examples uses only two features of the DOM, both of which are reasonably simple:
 - *Naming an element in the HTML file.* It is often necessary to refer to a specific element in the web page from inside the JavaScript code. To do so, you need to include an *id* attribute in the HTML tag for that element that gives that element a name. JavaScript code can then find that element by calling `document.getElementById(id)`.
 - *Adding HTML content to an existing element.* The HTML code inside an element is available by selecting the *innerHTML* field of the element. The result is a JavaScript string that you can examine and modify.
- The code on the next slide writes the string "hello, world" into the <div> element whose *id* attribute is "log".

An Improved Version of HelloWorld

```
<!DOCTYPE html>
<html>
<head>
<title>Hello World</title>
<script type="text/javascript">
function sayHello() {
let div = document.getElementById("log");
div.innerHTML = "hello, world";
}
</script>
</head>
<body onload="sayHello()">
<div id="log">
<!-- This is where the text eventually goes -->
</div>
</body>
</html>
```

Simulating a Countdown

```
<!DOCTYPE html>
<html>
<head>
<title>Hello World</title>
<script type="text/javascript">
function countdown(n) {
for (let i = n; i >= 0; i--) {
log(i);
}
}
function log(str) {
let div = document.getElementById("log");
div.innerHTML += str + "<br/>";
}
</script>
</head>
<body onload="countdown(10)">
<div id="log"></div>
</body>
</html>
```

Exercise: Factorial Table

Factorial Table	
0!	1
1!	1
2!	2
3!	6
4!	24
5!	120
6!	720
7!	5040
8!	40320
9!	362880
10!	3628800

The End