JAVASCRIPT AND THE DOM

LOGISTICS

PLAN FOR TODAY

- Share back MP0
- Intro to JavaScript
- Break
- Intro to the DOM, play with Codepen
- Lab time

SHARE BACK!

IN GROUPS

Get into small groups and show each other your web sites

IN GROUPS

Get into small groups and show each other your web sites

Volunteer to show in front of the class?

INTRO TO JAVASCRIPT

WHAT IS JAVASCRIPT?

JavaScript is a programming language that lets you add complexity to your web pages.

WHY USE JAVASCRIPT?

WHY USE JAVASCRIPT?

- **Automate content generation:** No more manual HTML!

WHY USE JAVASCRIPT?

- **Automate content generation:** No more manual HTML!
- **Customize behavior:** Run code when certain events occur

WHY USE JAVASCRIPT?

- **Automate content generation:** No more manual HTML!
- **Customize behavior:** Run code when certain events occur
- **Interact with the browser:** Use browser APIs, e.g. for notifications

WHY USE JAVASCRIPT?

- **Automate content generation:** No more manual HTML!
- **Customize behavior:** Run code when certain events occur
- **Interact with the browser:** Use browser APIs, e.g. for notifications
- **Web APIs:** Incorporate functionality from third-party APIs

BRIEF HISTORY

BRIEF HISTORY

- Web pages used to only be static!

BRIEF HISTORY

- Web pages used to only be static!
- In 1995 Netscape decided to add a scripting language

BRIEF HISTORY

- Web pages used to only be static!
- In 1995 Netscape decided to add a scripting language
- Brendan Eich notoriously designed and implemented the first version in 10 days

BRIEF HISTORY

- Web pages used to only be static!
- In 1995 Netscape decided to add a scripting language
- Brendan Eich notoriously designed and implemented the first version in 10 days
- Evolved into the most popular programming language in the world today!

JAVASCRIPT != JAVA

“The choice of the JavaScript name has caused confusion, implying that it is directly related to Java. At the time, the dot-com boom had begun and Java was the hot new language, so Eich considered the JavaScript name a marketing ploy by Netscape.”

https://en.wikipedia.org/wiki/JavaScript#History

JAVASCRIPT IS THE WORLD'S MOST POPULAR LANGUAGE

2022 Stack Overflow Developer Survey

IN SUMMARY

JavaScript is the *common language* that you can use to connect your personal page, the browser, and other libraries and frameworks.

ADDING JAVASCRIPT TO YOUR PAGE

INTERNAL: IN A SCRIPT TAG!

```
<script>
  console.log("Hello world!");
</script>
```

EXTERNAL: IN ANOTHER FILE!

index.html:

```
<script src="script.js" defer></script>
```

script.js:

```
console.log("Hello World!");
```

THE CONSOLE

Write to the console using `console.log()`

```
<script type="module">
  console.log("I can do math", 1 + 2 + 3 * 7);
  console.log({ favoriteFood: "potato", bestFriend: "potato" });
</script>
```

A QUICK TOUR OF JAVASCRIPT

COMMENTS

```
// I am a single-line comment
let r = 25;
```

```
/*
  I am also
  a comment
*/
let x = 15;
```

TWO WAYS OF DEFINING VARIABLES

```
// Let is for when a value will change in the future
let index = 5;

// Const is for when it a value will not change
const numFingers = 10;
```

TWO WAYS OF DEFINING VARIABLES

```
// Let is for when a value will change in the future
let index = 5;

// Const is for when it a value will not change
const numFingers = 10;
```

Use `const` when you can, and use `let` when you have to.

PRIMITIVE DATA TYPES

```
// There are seven primitive data types

const myName = "Hannah"; // String
const myNum = 3.4; // Number
const logical = true; // Boolean

let x; // undefined - absence of a value
let y = null; // null - absence of an object

// Wont use these as often
const z = 9233440071992547409 91n; // Bigint - for really big numbers
const sym1 = Symbol(); // Symbol - when you need a unique value
```

STRINGS

```
// Can use double quotes or single quotes
let str = "Hello";
let str2 = 'Single "quotes" are ok too';

// Template literals are extremely useful
let phrase = `can embed another ${str}`;

console.log(str.length); // 5
```

BOOLEANS

```
let x = true;
let y = false;

console.log(x === y); // false

let isGreater = 4 > 1;

alert(isGreater); // true
```

NULL

```
let age = null; // represents “nothing”, “empty” or “value unknown”
```

UNDEFINED

```
let greeting;

alert(greeting); // shows "undefined"
```

NON-PRIMITIVE TYPE: OBJECTS

```
// Objects are for more complex data structures
// Think of them as collections of properties
let user = {
  name: "hannah",
  age: 584,
  loggedIn: true,
};
```

ARRAYS

An array is an ordered collection of elements.

```
// Some different ways of making an array:
const a = [1, 2, 3];

let b = [];
b.push(1, 2, 3);

const c = Array.of(1, 2, 3);

// Arrays can hold any value, even values of different types
const d = [1, "Banana", ["a", "b"], true];
console.log(e.length); // 4
```

MULTI-DIMENSIONAL ARRAYS

```
// Arrays of arrays

const matrix = [
  [1, 2, 3],
  [4, 5, 6],
  [7, 8, 9],
];

matrix[0][0]; // 1
matrix[2][0]; // 7
```

CONDITIONALS

```
const browserType = "mozilla";

if (browserType.includes("zilla")) {
  console.log("Found zilla!");
} else {
  console.log("No zilla here!");
}
```

LOOPS

```
// Low-level way of looping, more verbose
for (let i = 0; i < 10; i++) {
  console.log(i);
}

// In the future, we will cover more ways of looping
```

FUNCTIONS

Functions are a set of statements that can take an input and produce an output.

```
function square(num) {
  return num * num;
}

let newNum = square(4);
console.log(newNum); // 16

// In the future we will cover more ways you can make functions
```

BRACES AND SEMICOLONS

Unlike Python, JavaScript uses curly braces to denote the start and end of a statement.

```python
# Python

if (5 > 7):
    print("math is broken")
```

```javascript
// JavaScript

if (7 > 5) {
  console.log("math is fixed");
}
```

BREAK!

THE DOM

WHAT IS THE DOM?

WHAT IS THE DOM?

The Document Object Model

WHAT IS THE DOM?

The Document Object Model

The DOM is the browser’s internal model of our HTML.

WHAT IS THE DOM?

The Document Object Model

The DOM is the browser’s internal model of our HTML.

We use the DOM API to manipulate pages in the browser.

WHAT CAN YOU DO WITH THE DOM?

WHAT CAN YOU DO WITH THE DOM?

- Query for existing elements

WHAT CAN YOU DO WITH THE DOM?

- Query for existing elements
- Create and add new elements

WHAT CAN YOU DO WITH THE DOM?

- Query for existing elements
- Create and add new elements
- Listen for events

WHAT CAN YOU DO WITH THE DOM?

- Query for existing elements
- Create and add new elements
- Listen for events
- Much more…

USING THE DOM API

```
// Access the DOM API using "document"

document.
```

CREATING ELEMENTS

```
let bookContainer = document.getElementById("bookmarks");

let child = document.createElement("div");
child.innerHTML = bookmark.url;
bookContainer.appendChild(child);
```

Codepen: Appending children to a div with javascript

QUERYING THE DOM

```
let selector = ".red-box";

// Get all elements with the class "red-box"
let boxes = document.querySelectorAll(selector);

// Get the first element with the class "red-box"
let firstBox = document.querySelector(selector);
```

Codepen: Querying the DOM and changing colors

ADDING AN EVENT LISTENER

```
function sayHi() {
  alert("hi!");
}

const button = document.querySelector("#hi-button");
button.addEventListener("click", sayHi);
```

Codepen: Adding Event Listeners to elements

INTRODUCING MP1

MP1: BROWSER EXTENSION

For the next two weeks, you will be working on MP1.

Instead of me giving you a bunch of JS to memorize, we are going to do the project the way you might on your own.

MP1 STEPS

MP1 STEPS

1. Think of what you want to make.

MP1 STEPS

1. Think of what you want to make.
2. Figure out how to make it.

NEXT CLASS

- Share ideas for extensions
- Talk about how extension development will work
- Live demos of extensions

Until then, you are encouraged to play around with the examples.

LAB TIME!