Week 8

JavaScript 101

We've passed the halfway point!

HTML

Structure

CSS

Design

JavaScript

Function

JavaScript

Programming language designed in 2 weeks

Not really related to Java

Initially, used to add some interactivity to webpages

Now used everywhere, from servers to IoT devices

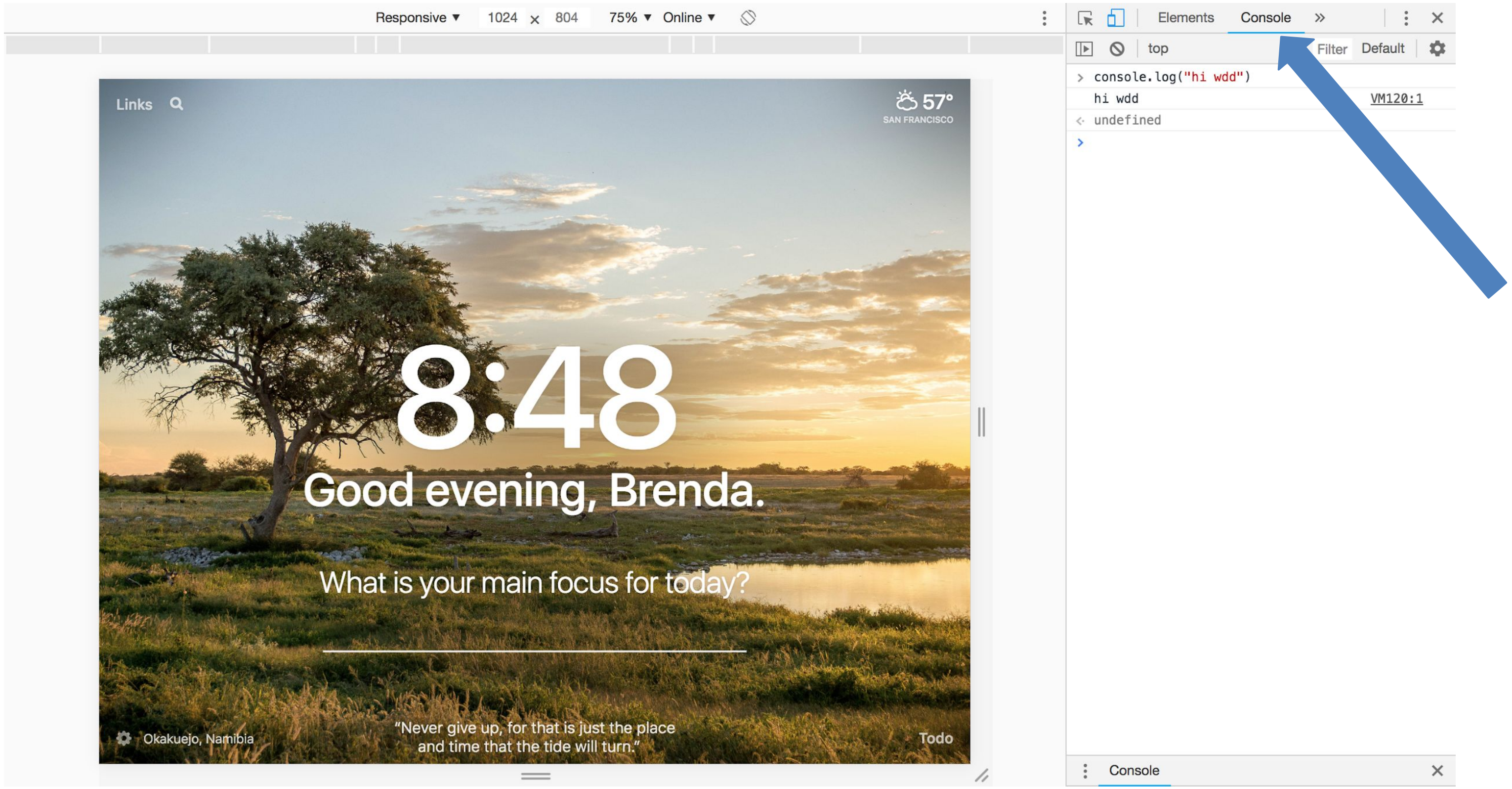
What can JavaScript do?

A brief tour of JavaScript’s magic 🚀

Declarative vs. Imperative.

HTML & CSS vs. JavaScript

Get to know the JavaScript Console



```
> console.log("hi wdd")
  hi wdd                                   VM120:1
<· undefined
>
```

The JS console is your playground, where you can test ideas.

JavaScript 101

"Printing" things to the console output

```
console.log('blah');
```

Comments

```
// hello, this is a comment.
/* this is also
   a comment */
```

Let’s write some JavaScript!

A few primitive types

The building blocks of JS (and other programming languages)

- **`number:`** `1, 2, 3, 0.8, 100, 1.6`
- **`string:`** `"hello", "wdd", "wowza cool class"`
- **`boolean:`** `true, false`

Number

Examples:

- `24601` (decimal), `0x3ebd35` (hexadecimal)

Some mathematical operations:

- Regular: `+`, `-`, `*`, `/`
- Power: *`base`* `**` *`exponent`*
- Modulo: *`dividend`* `%` *`divisor`* (reasonable with positive numbers)
  `5 % 2 == 1` because `5 / 2 == 2` (floor div), remainder `1`

Boolean

Two possible values: `true`, `false`

Expressions that evaluate to booleans, often times comparisons:

- Loose equality operator → `==`, `!=`
    - `1 + 5 == 6` evaluates to `true`
    - `1 == 2` evaluates to `false`
    - `1 + 5 != 2` evaluates to `true`
- Other relational operators → `>`, `<`, `>=`, `<=`

Logic operations

Or → `||`

Just one thing in the expression has to be true

- `true || false`
  evaluates to `true`
- `1 + 1 == 4 || 1 + 1 == 2`
  evaluates to `true`
- `8 == 8 || 1 == 1`
  evaluates to `true`
- `2 == 0 || 1 == 8`
  evaluates to `false`

And → `&&`

Everything in the expression has to be true

- `true && false`
  evaluates to `false`
- `1 + 1 == 4 && 1 + 1 == 2`
  evaluates to `false`
- `8 == 8 && 1 == 1`
  evaluates to `true`
- `2 == 0 && 1 == 8`
  evaluates to `false`

Variables

We can **declare** (once) variables with unique names to hold values for later use

Format:

```
let variableName = value;
```

Example:

```
let numStudents = 120;
```

To reassign a different value: (notice that we don't use `let` here because of reassigning)

```
numStudents = 120;
```

```
// We start with 20 potatoes
let numPotatoes = 20;

// Print it out in the console
console.log("We have", numPotatoes, "potatoes");

// We sold 10 potatoes during the day
// The equal symbol here means assignment
numPotatoes = numPotatoes - 10;

// Print out how much we have left
console.log("We have", numPotatoes, "potatoes");
```

https://playcode.io/140537?tabs=console&script.js&output

Constants

Again, we can **declare** (once) variables with unique names to hold values for later use

Format:

```
const variableName = value;
```

Example:

```
const jacobsFireCode = 140;
```

Once initialized, we **cannot reassign** the variable to a different value :(

```
// A gold potato has a weight of 200 pounds
const goldPotatoWeight = 200;

// Someone comes at night and wanted to change it
:(
goldPotatoWeight = 10;

// TypeError:
// Attempted to assign to readonly property.
```

Let's take 4 minutes for a little practice

https://playcode.io/286149?tabs=console&script.js&output

Functions

Functions are reusable pieces of code
We can define our own functions too!

Syntax:

```
function functionName(arg1, ...) {
    // Do something...
    return returnValue;
}
```

```
function multiply2(number) {
    return number * 2;
}

console.log(1);                              // 1
console.log(multiply2(1));                   // 2
console.log(multiply2(multiply2(1)));        // 4

let mul2 = multiply2;

console.log(mul2(4));                        // 8
```

HTML □ JavaScript

Linking HTML and JavaScript file

How will the HTML file know where to find its codin' ?!

At the end of the `<body>` tag, add this line:

```
<script src="path/to/your/script.js"></script>
```

*No need to memorize this — I'd just copy and paste or something like that

Event handling: A primer

```
<img id="panic-button">

<script type="text/javascript">
  function panicButtonClicked() {
      alert("Ahh! Somebody just clicked the panic button :o")
  }

  // We want the browser to run panicButtonClicked() for us when someone clicks
  // the button with id "panic-button"
  document.getElementById("panic-button").onclick = panicButtonClicked;
</script>
```

https://playcode.io/140602?tabs=script.js&output

Setting inline style in JavaScript

```
<img id="pumpkin">
<div>Width: <input id="pumpkin-width" type="range" min="10" max="200" value="100"></div>

<script type="text/javascript">
 // Note that document.getElementById("id-of-some-element") gives you an element
 // Then you can use element.style.cssPropertyName = cssPropertyValue to change its style

 function pumpkinSizeUpdated() {
   const width = document.getElementById("pumpkin-width").value;
   const pumpkin = document.getElementById("pumpkin");
   pumpkin.style.width = width + "px";
 }

 // We want the browser to run pumpkinSizeUpdated() when the range slider is changed immediately
 document.getElementById("pumpkin-width").oninput = pumpkinSizeUpdated;
 </script>
```

See demo on next slide...

🙋‍♀️ 🙋‍♂️

Questions?

👋

Attendance