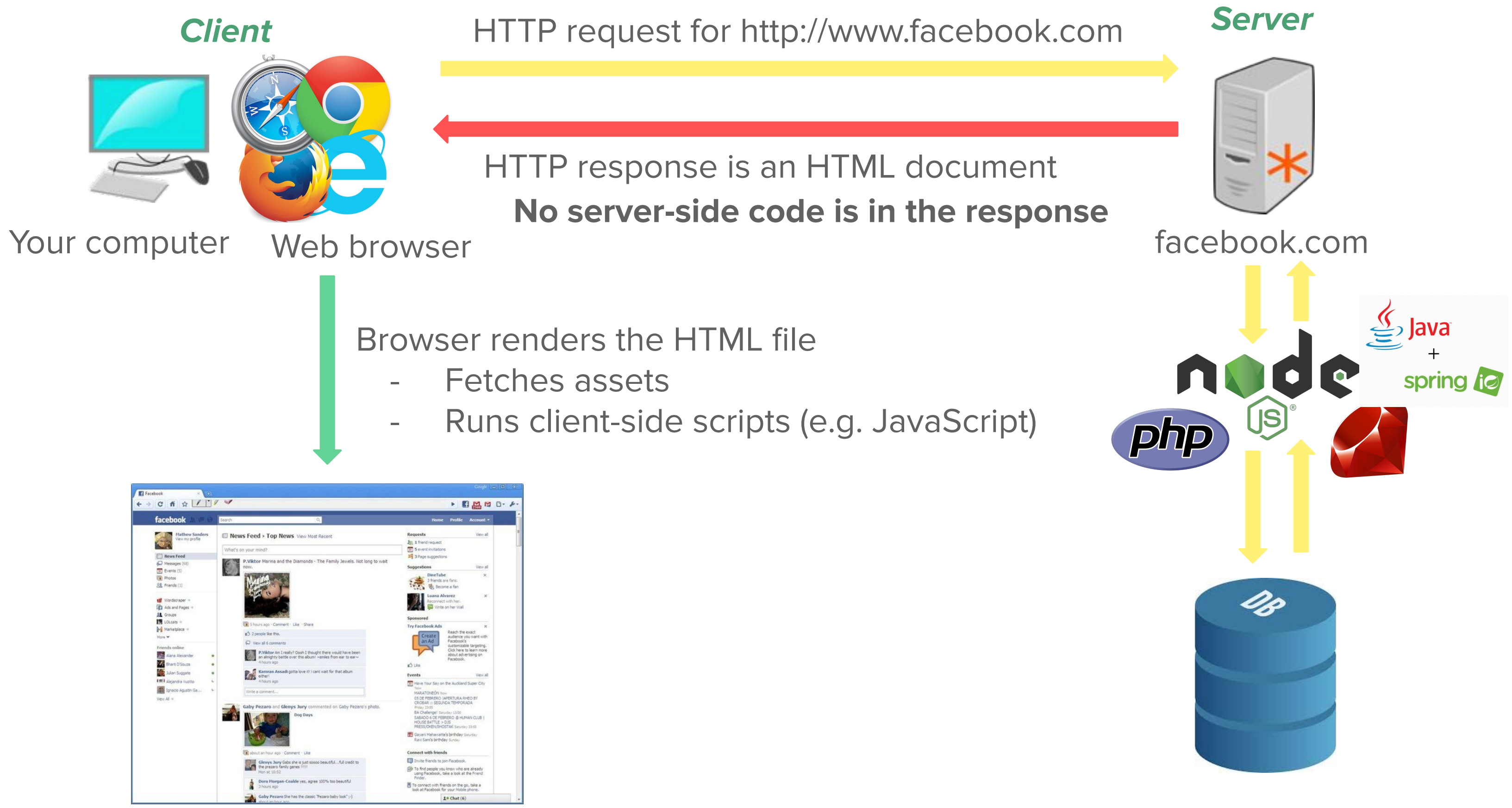
Lecture 07: Intro to JS, JS Events and Traversal

ITP 303 Full-Stack Web Development

Intro to JavaScript

The journey so far...



Client

Your computer

Web browser

HTTP request for http://www.facebook.com

HTTP response is an HTML document

No server-side code is in the response

Server

facebook.com

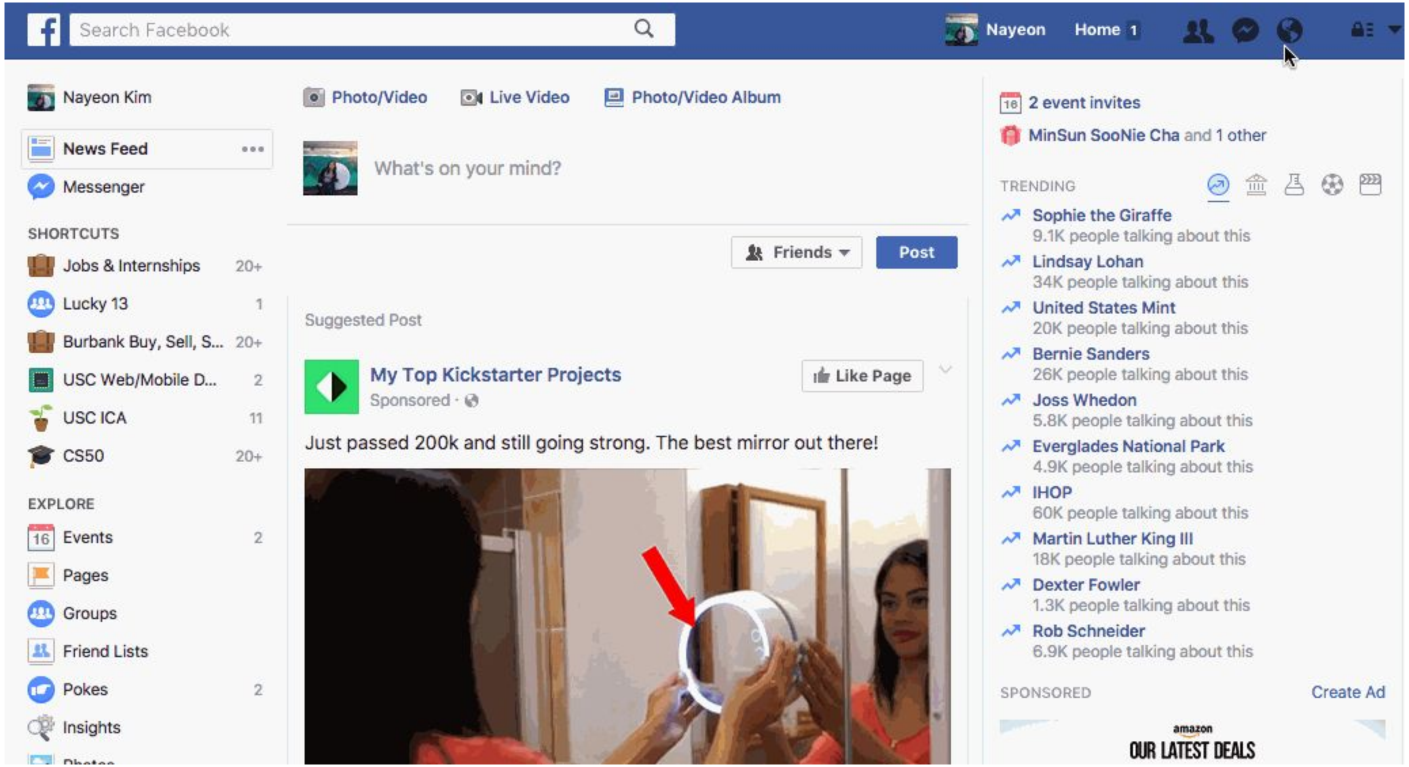
Browser renders the HTML file

- Fetches assets
- Runs client-side scripts (e.g. JavaScript)

JavaScript (JS)

- A programming language that can be read and executed by the browser.
- While many ideas are borrowed from the language Java, **Java and JavaScript are are two entirely different languages**.
- Primarily used on client-side, but with NodeJS on the backend, it’s possible to build an entire web application with just JavaScript
- Some key things JS is good at:
  - DOM manipulation
  - Listening to user events (click, hover, keypress, etc)
  - Client-server communication without reloading pages
  - And more…

JavaScript in live action



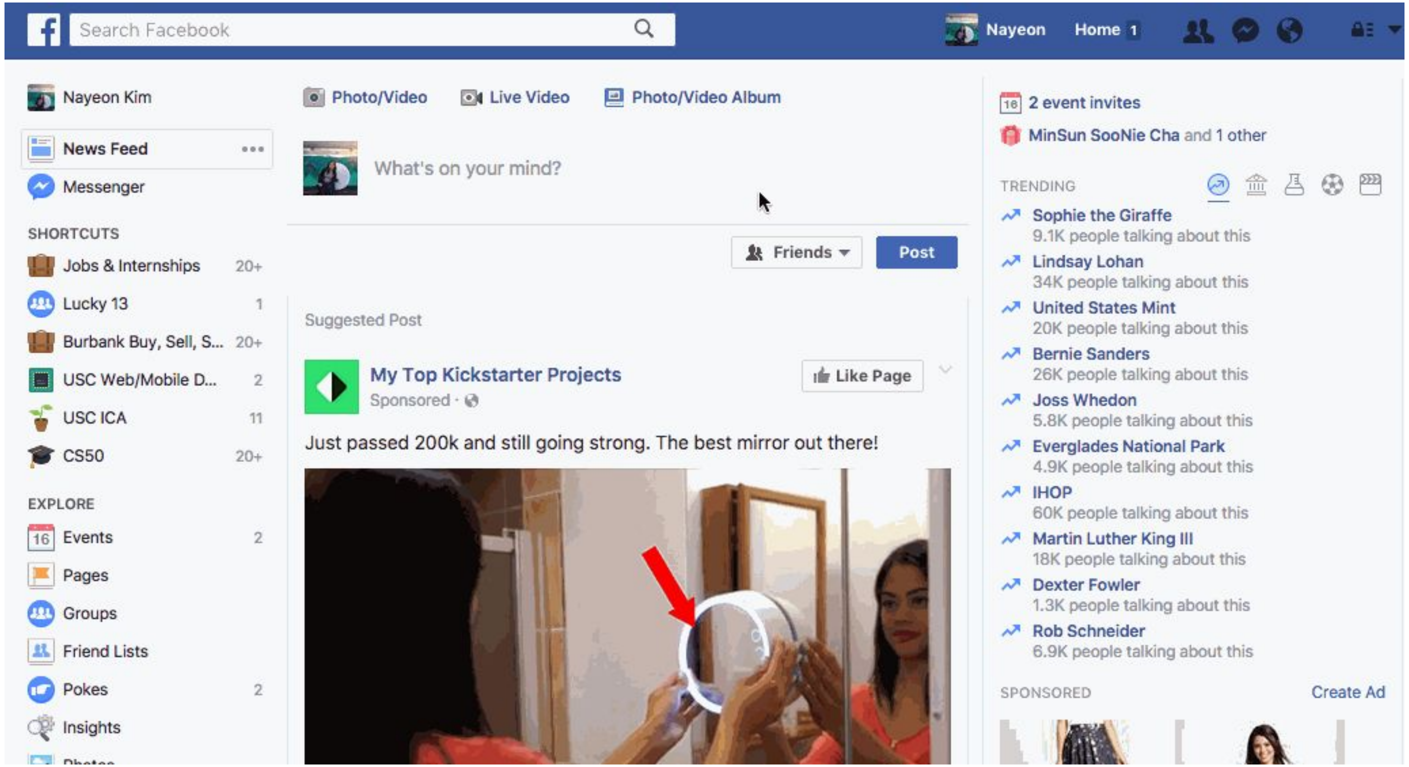
MinSun SooNie Cha and 1 other

My Top Kickstarter Projects
Sponsored ·

Just passed 200k and still going strong. The best mirror out there!

Sophie the Giraffe	9.1K people talking about this
Lindsay Lohan	34K people talking about this
United States Mint	20K people talking about this
Bernie Sanders	26K people talking about this
Joss Whedon	5.8K people talking about this
Everglades National Park	4.9K people talking about this
IHOP	60K people talking about this
Martin Luther King III	18K people talking about this
Dexter Fowler	1.3K people talking about this
Rob Schneider	6.9K people talking about this

JavaScript in live action



Search Facebook

Nayeon Home 1

Nayeon Kim

News Feed

Messenger

SHORTCUTS

Jobs & Internships	20+
Lucky 13	1
Burbank Buy, Sell, S...	20+
USC Web/Mobile D...	2
USC ICA	11
CS50	20+

EXPLORE

Events	2
Pages	
Groups	
Friend Lists	
Pokes	2
Insights	

Photo/Video Live Video Photo/Video Album

What's on your mind?

Friends Post

Suggested Post

My Top Kickstarter Projects
Sponsored ·

Like Page

Just passed 200k and still going strong. The best mirror out there!

2 event invites

MinSun SooNie Cha and 1 other

TRENDING

Sophie the Giraffe
9.1K people talking about this

Lindsay Lohan
34K people talking about this

United States Mint
20K people talking about this

Bernie Sanders
26K people talking about this

Joss Whedon
5.8K people talking about this

Everglades National Park
4.9K people talking about this

IHOP
60K people talking about this

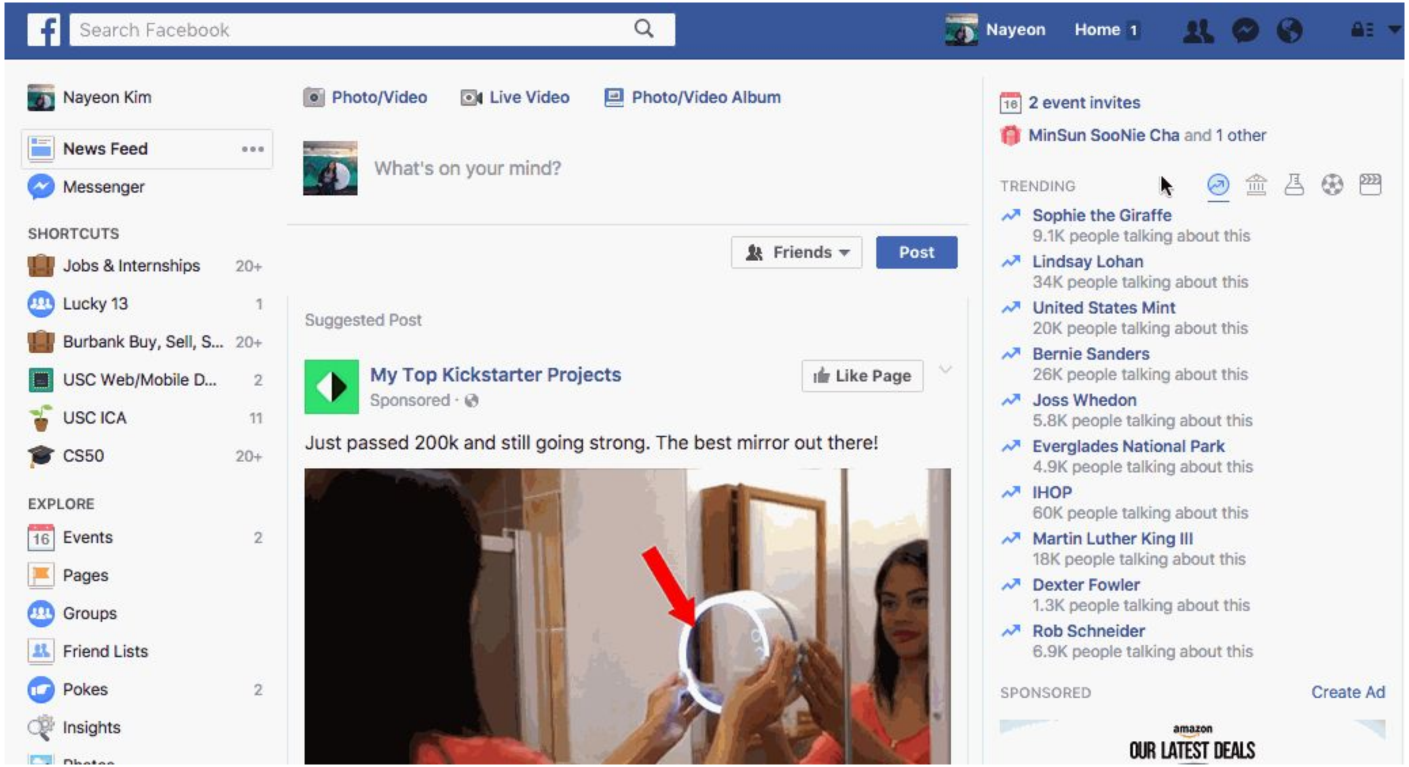
Martin Luther King III
18K people talking about this

Dexter Fowler
1.3K people talking about this

Rob Schneider
6.9K people talking about this

SPONSORED Create Ad

JavaScript in live action



What's on your mind?

Suggested Post

My Top Kickstarter Projects
Sponsored ·

Just passed 200k and still going strong. The best mirror out there!

2 event invites
MinSun SooNie Cha and 1 other

TRENDING

Sophie the Giraffe
9.1K people talking about this

Lindsay Lohan
34K people talking about this

United States Mint
20K people talking about this

Bernie Sanders
26K people talking about this

Joss Whedon
5.8K people talking about this

Everglades National Park
4.9K people talking about this

IHOP
60K people talking about this

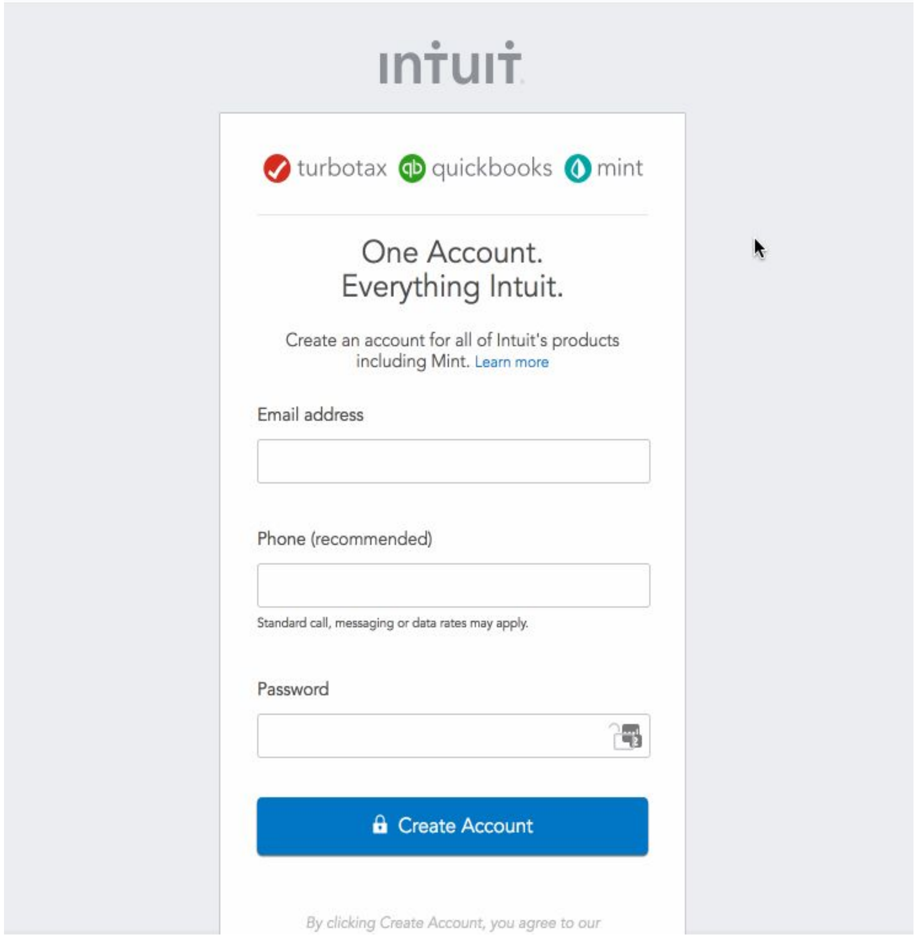
Martin Luther King III
18K people talking about this

Dexter Fowler
1.3K people talking about this

Rob Schneider
6.9K people talking about this

SPONSORED

JavaScript in live action



intuit

turbotax quickbooks mint

One Account.
Everything Intuit.

Create an account for all of Intuit's products including Mint. Learn more

Email address

Phone (recommended)

Standard call, messaging or data rates may apply.

Password

Create Account

By clicking Create Account, you agree to our

JS syntax

```
var name = “nayeon”;

var favoriteNumber = 2;

var isInstructor = true;
```

JS syntax

```
let name = “nayeon”;

let favoriteNumber = 2;

let isInstructor = true;
```

Resource: More on let

JS syntax

```
let book = {
  title: “Jane Eyre”,
  author: “Charlotte Bronte”,
  published: 1847
}
```

JS syntax

```
if(5 > 7) {
  // some code
}
else {
  // some code
}
```

JS syntax

```
for (let i = 0; i<list.length; i++) {

  // code to iterate here

}
```

JS syntax

```
let numberArray = [1,2,3,4,5];
```

JS syntax

```
let numberArray = [1,2,3,4,5];

numberArray.push(6);

// now numberArray is has 1,2,3,4,5,6
```

JS syntax

```
function sayHello() {

  console.log(“Hello!”);

}
```

JS syntax

```
let sayHello = function() {

  console.log(“Hello!”);

}
```

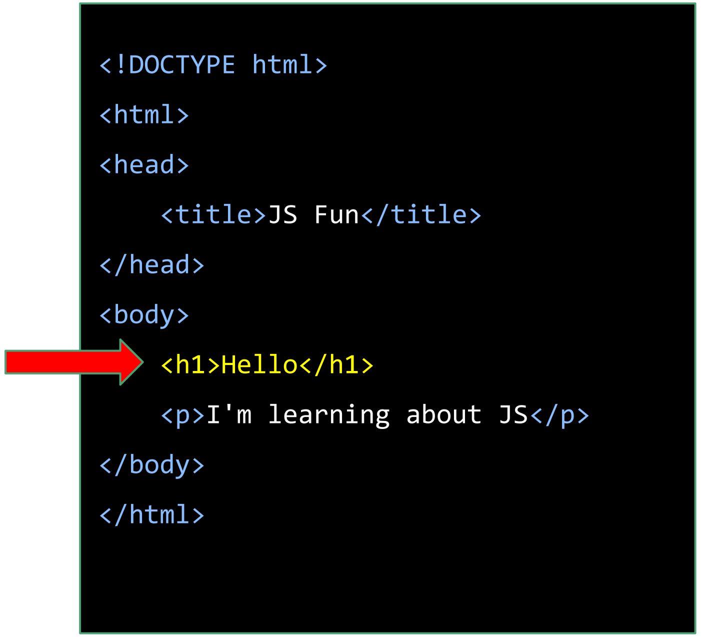
JS syntax

```
let sayHello = function() {
  console.log(“Hello!”);
}
// To call the function
sayHello();
```

Two key features of JS

1. Searching and changing elements in the DOM
2. Listening to user events

Searching elements??? What does that mean?



```
<!DOCTYPE html>
<html>
<head>
    <title>JS Fun</title>
</head>
<body>
    <h1>Hello</h1>
    <p>I'm learning about JS</p>
</body>
</html>
```

JavaScript allows us to easily **find** any specific element in our HTML file.

To fully understand this, we need to first learn about the **Document Object Model**

Document Object Model (DOM)

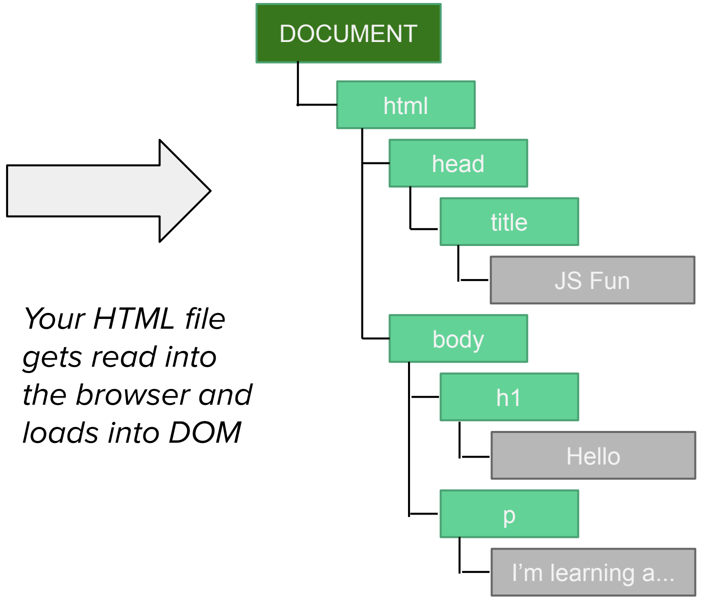
- A tree-like structure that represents a web page which can be utilized quickly access elements using JS

Document Object Model (DOM)

- A tree-like structure that represents a web page which can be utilized quickly access elements using JS

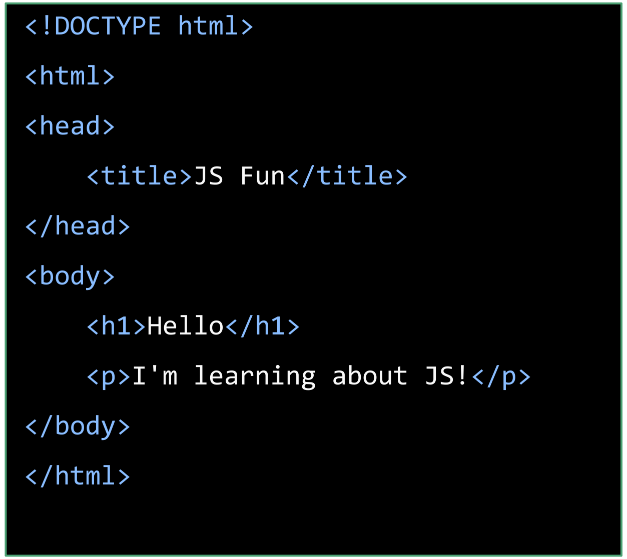
```html
<!DOCTYPE html>
<html>
<head>
    <title>JS Fun</title>
</head>
<body>
    <h1>Hello</h1>
    <p>I'm learning about JS!</p>
</body>
</html>
```

Your HTML file gets read into the browser and loads into DOM

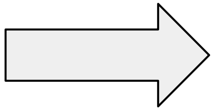


Document Object Model (DOM)

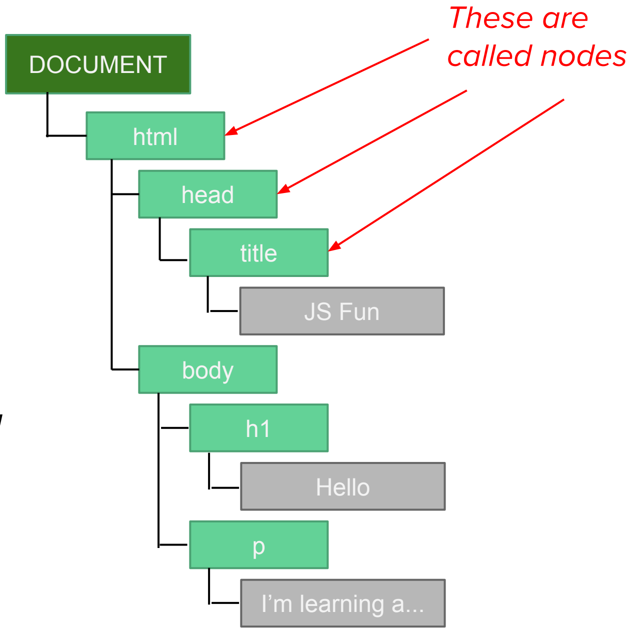
- A tree-like structure that represents a web page which can be utilized quickly access elements using JS



```
<!DOCTYPE html>
<html>
<head>
    <title>JS Fun</title>
</head>
<body>
    <h1>Hello</h1>
    <p>I'm learning about JS!</p>
</body>
</html>
```

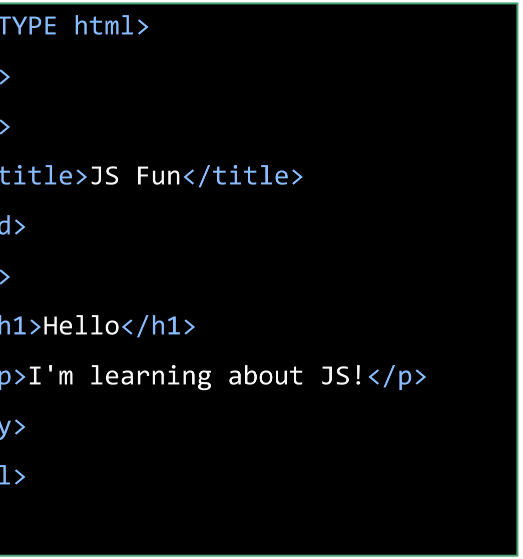


Your HTML file gets read into the browser and loads into DOM

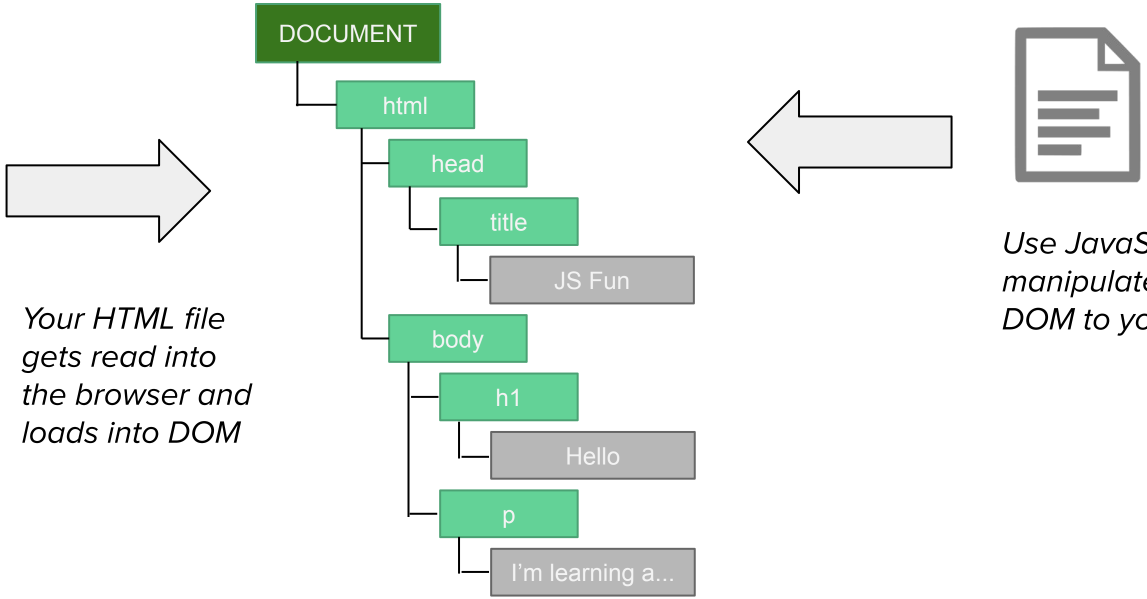


Ok… so what does JS do?

- JavaScript gives us a language to interact with the DOM



```
TYPE html>
>
>
title>JS Fun</title>
d>
>
h1>Hello</h1>
p>I'm learning about JS!</p>
y>
l>
```

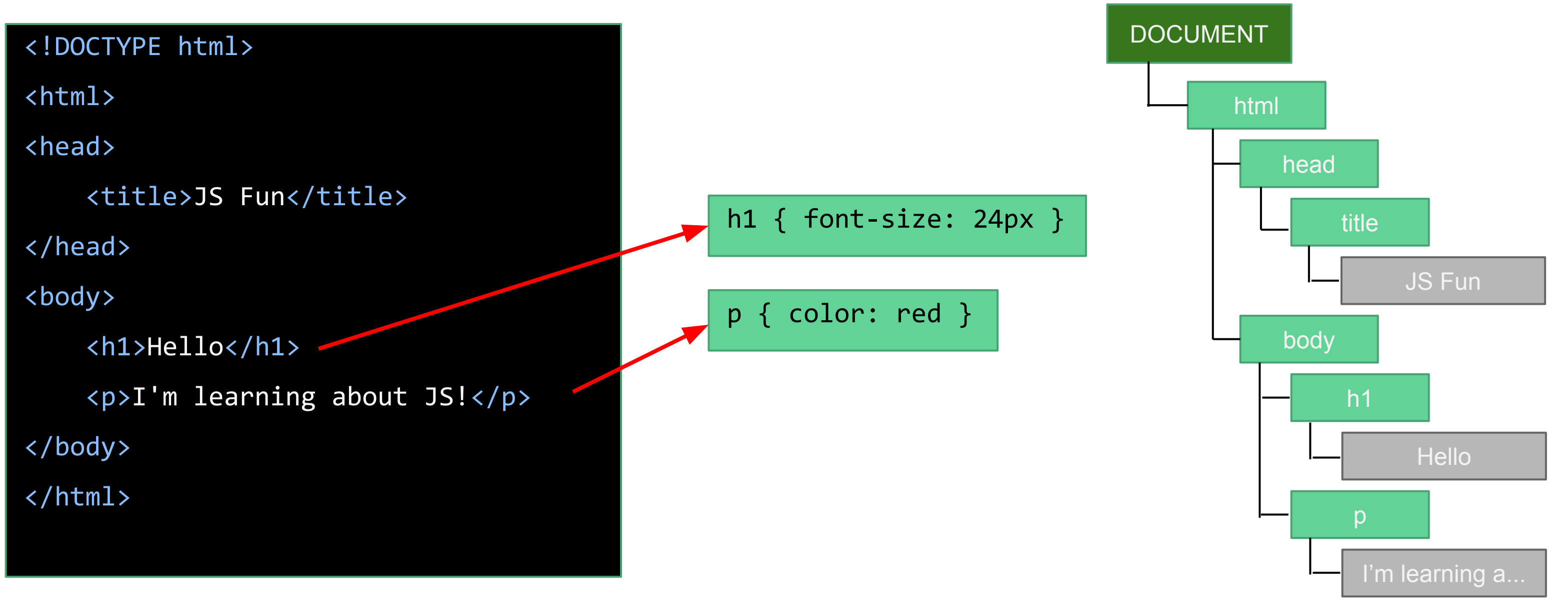


Your HTML file gets read into the browser and loads into DOM

Use JavaScript to manipulate the DOM to your liking

How JS “Finds elements” aka accesses the DOM

- We can select any DOM elements with JS by using the same syntax as CSS selectors



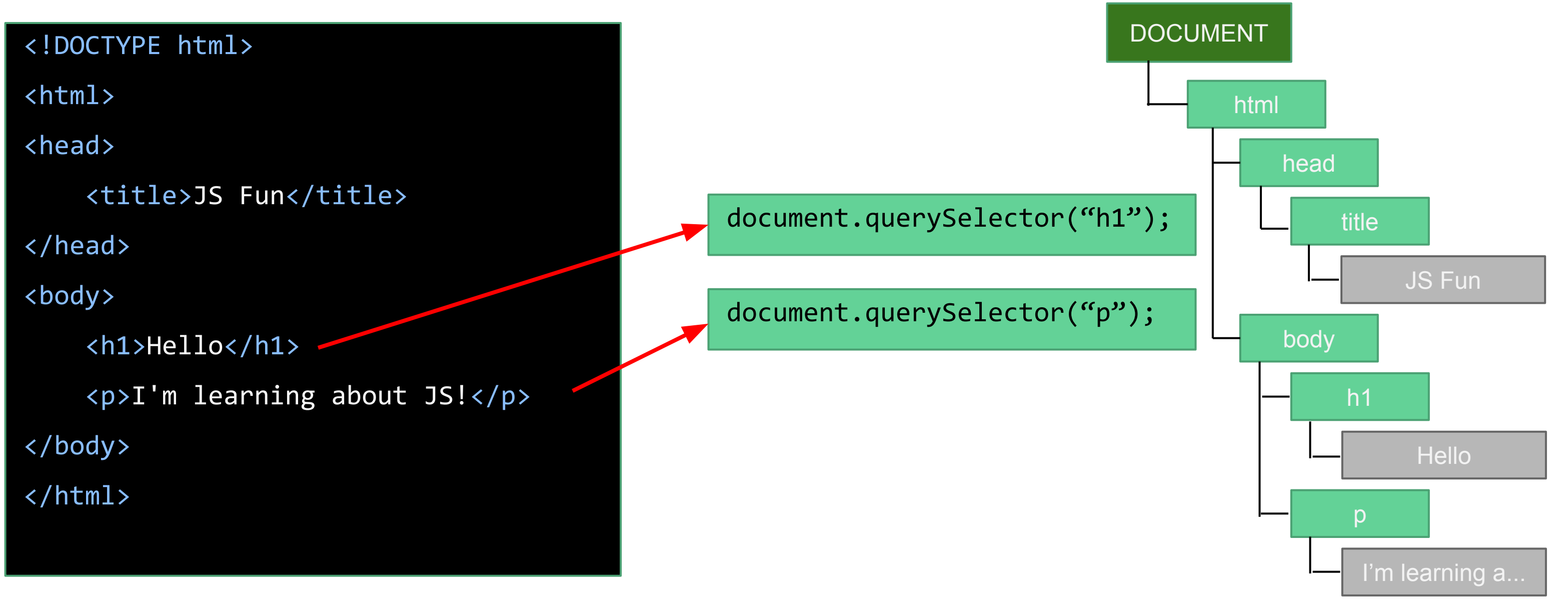
```html
<!DOCTYPE html>
<html>
<head>
    <title>JS Fun</title>
</head>
<body>
    <h1>Hello</h1>
    <p>I'm learning about JS!</p>
</body>
</html>
```

```css
h1 { font-size: 24px }
```

```css
p { color: red }
```

How JS “Finds elements” aka accesses the DOM

- We can select any DOM elements with JS by using the same syntax as CSS selectors



```
<!DOCTYPE html>
<html>
<head>
    <title>JS Fun</title>
</head>
<body>
    <h1>Hello</h1>
    <p>I'm learning about JS!</p>
</body>
</html>
```

```
document.querySelector(“h1”);
```

```
document.querySelector(“p”);
```

Accessing DOM Nodes

There are several ways to select DOM Nodes (HTML Elements) using JS:

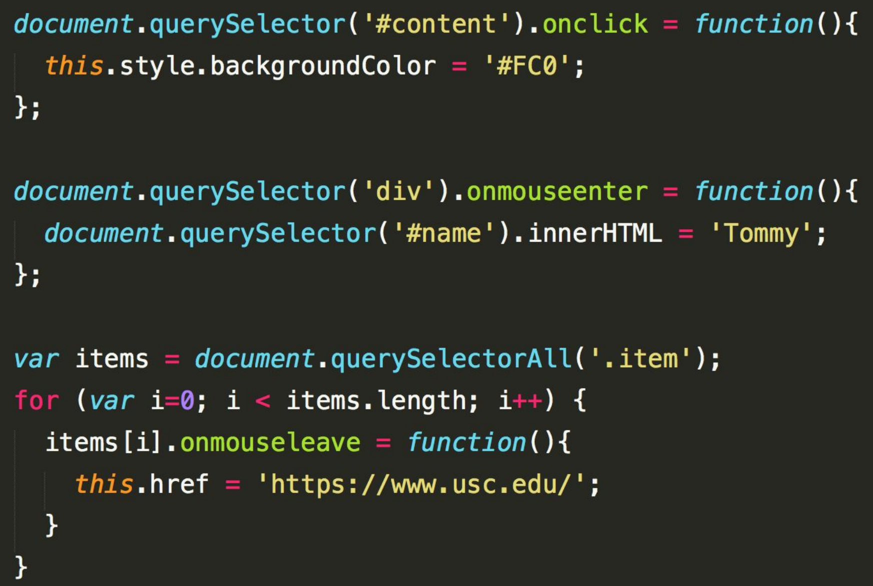
- `document.querySelector(` *`CSS Selector`*`)`
- `document.querySelectorAll(` *`CSS Selector`*`)`
- `document.getElementById(` *`ID`*`)`
- `document.getElementsByClassName(` *`Class`*`)`
- `document.getElementsByTagName(` *`Tag`*`)`

DOM Events

Now that we can “find” elements, we can also wait and listen for an event to trigger the element.

`onclick`	Mouse right-click on an element.
`onmouseenter`	Mouse enters an element.
`onmouseleave`	Mouse leaves an element.
`onmouseover`	Mouse enters an element or its children.
`onmouseout`	Mouse leaves an element or its children.

Resource: Full list of DOM Events.

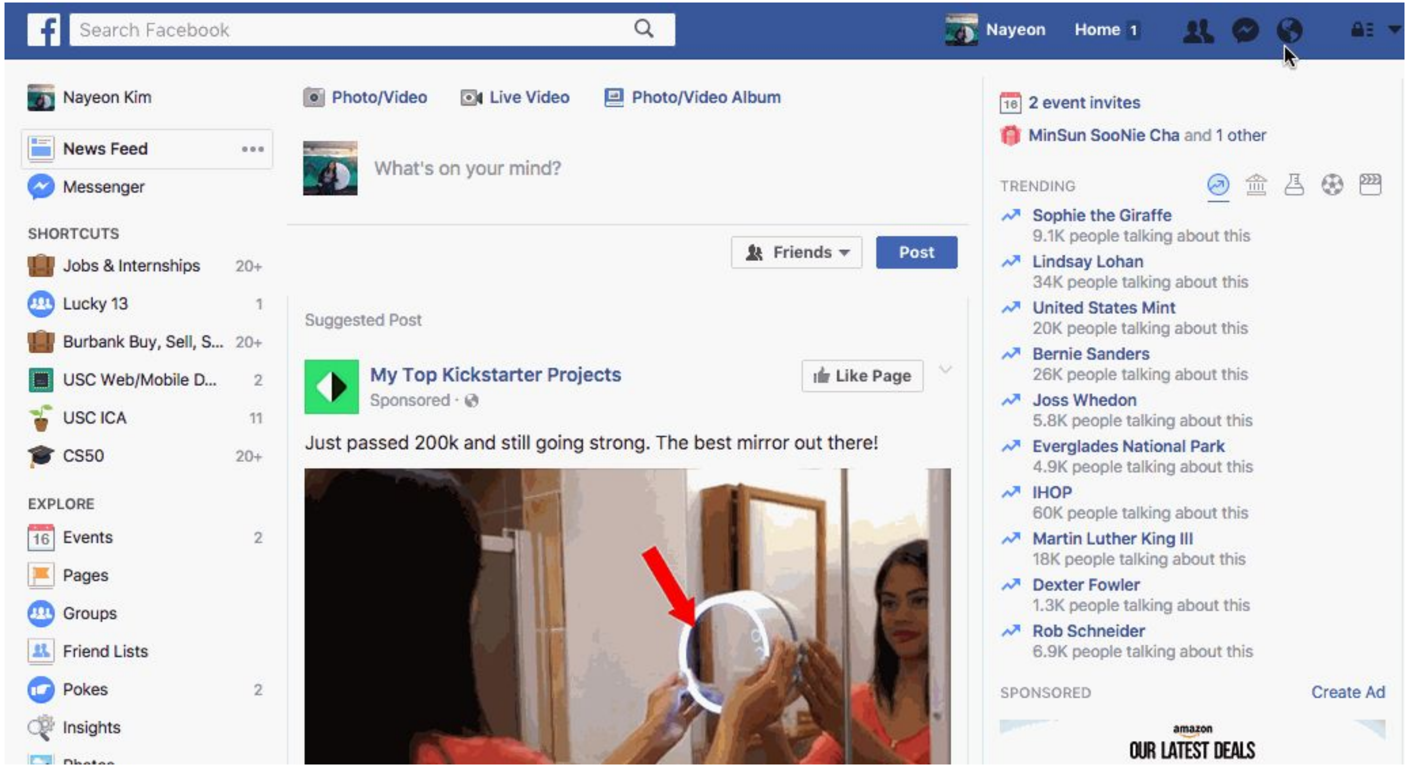


```
document.querySelector('#content').onclick = function(){
  this.style.backgroundColor = '#FC0';
};

document.querySelector('div').onmouseenter = function(){
  document.querySelector('#name').innerHTML = 'Tommy';
};

var items = document.querySelectorAll('.item');
for (var i=0; i < items.length; i++) {
  items[i].onmouseleave = function(){
    this.href = 'https://www.usc.edu/';
  }
}
```

JavaScript in live action



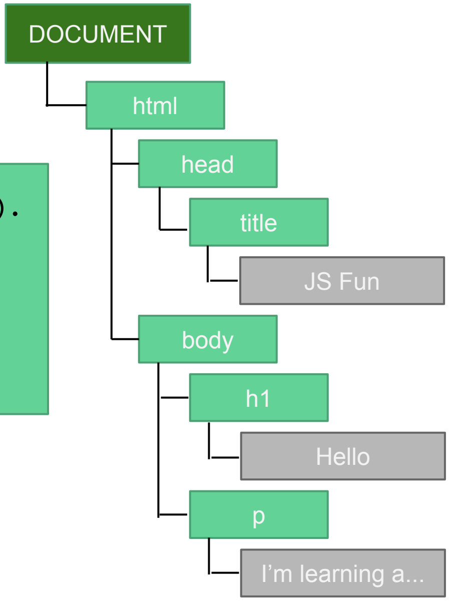
Just passed 200k and still going strong. The best mirror out there!

DOM Traversal is possible too

- Without specifying an element, can find neighboring elements like sibling, parent, child, etc

```
<!DOCTYPE html>
<html>
<head>
    <title>JS Fun</title>
</head>
<body>
    <h1>Hello</h1>
    <p>I'm learning about JS!</p>
</body>
</html>
```

```
document.querySelector(“h1”).
nextSibling;
// returns <p>I’m learning
about JS!</p>
```



DOM Traversal properties

`parentNode`	Parent element.
`children`	Children elements.
`nextSibling`	Next sibling, including whitespace (text) nodes.
`nextElementSibling`	Next sibling, excluding whitespace (text) nodes.
`previousSibling`	Previous sibling, including whitespace (text) nodes.
`previousElementSibling`	Previous sibling, excluding whitespace (text) nodes.