

The background image is an aerial photograph of a road intersection. A winding road with a red and white circular decorative center is visible. The area is surrounded by lush greenery, including many palm trees, and some agricultural fields. A large green building with a blue roof is situated near the intersection.

CASA0017: Web Architecture

Introduction to JavaScript



Steven Gray



- | | | | |
|---|--------------------------|----|------------------------------|
| 1 | Introduction to HTML | 6 | Node.js 1 |
| 2 | Programming JavaScript 1 | 7 | Advanced Node.js 2 |
| 3 | Programming JavaScript 2 | 8 | 3D Visualisation Libraires |
| 4 | Advanced JavaScript 3 | 9 | Realtime Data Visualisations |
| 5 | Sharing Data on the Web | 10 | The Intelligent Web |

Recap

An Example HTML Document

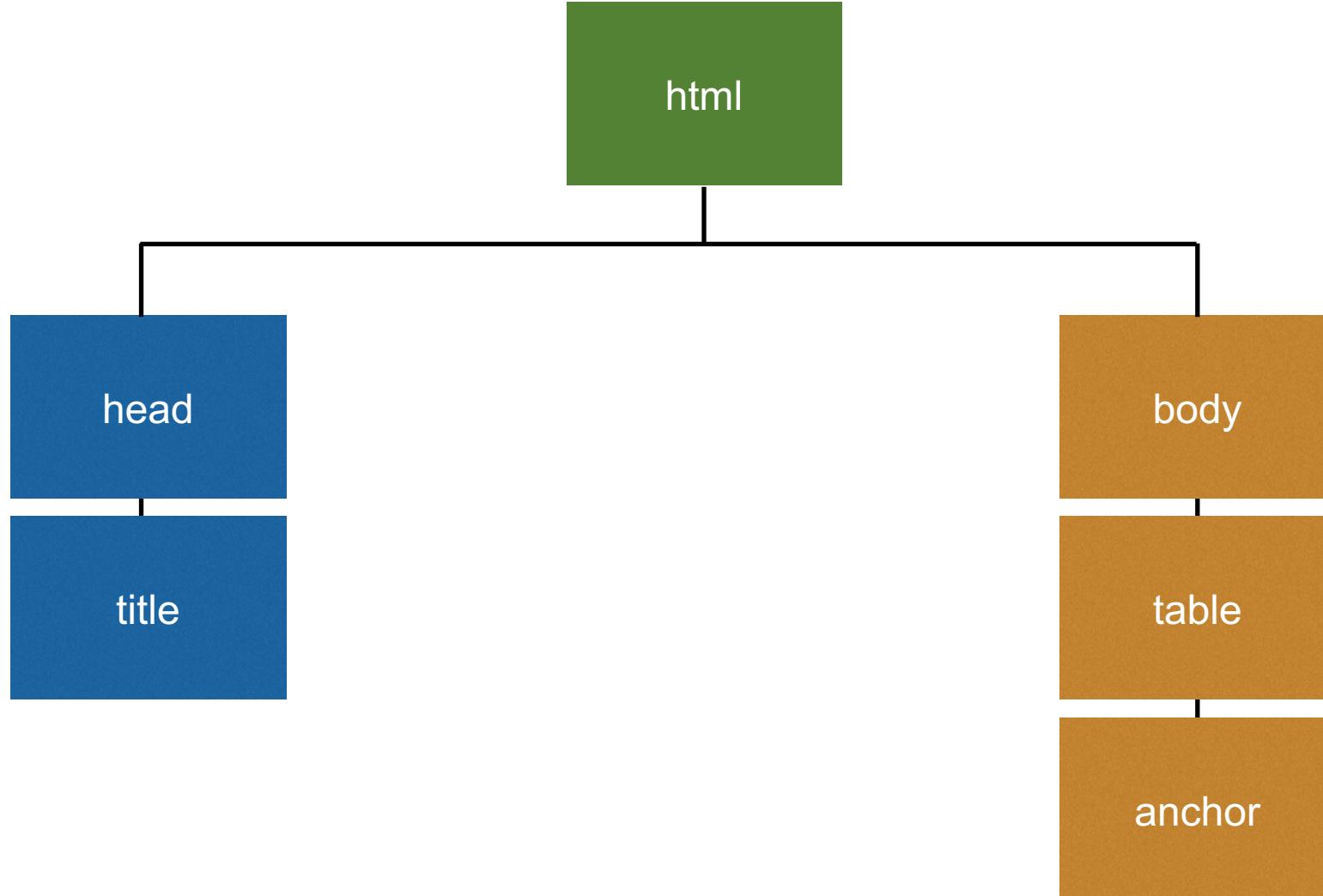
```
<!DOCTYPE HTML PUBLIC "-//W3C//DTD HTML 4.01 Transitional//EN"
  "http://www.w3.org/TR/html4/loose.dtd">
<html>
  <head>
    <meta http-equiv="Content-type" content="text/html; charset=utf-8">
    <title>Test Page</title>
  </head>

  <body>
    <table border="0" cellspacing="5" cellpadding="5">
      <tr><th>Header</th></tr>
      <tr><td>Data</td></tr>
    </table>

    <a href="/css/master%20set.css">Hello World</a>
  </body>
</html>
```

Recap

Example of DOM Tree (previous HTML Document)



What is JavaScript?

Client-side coding

Built by Netscape in 1995 to enable browsers to run sandboxed code.

Birth of the Interactive Web

Different implementations of the language due to different browsers

Web 2.0 heavily relies on JavaScript

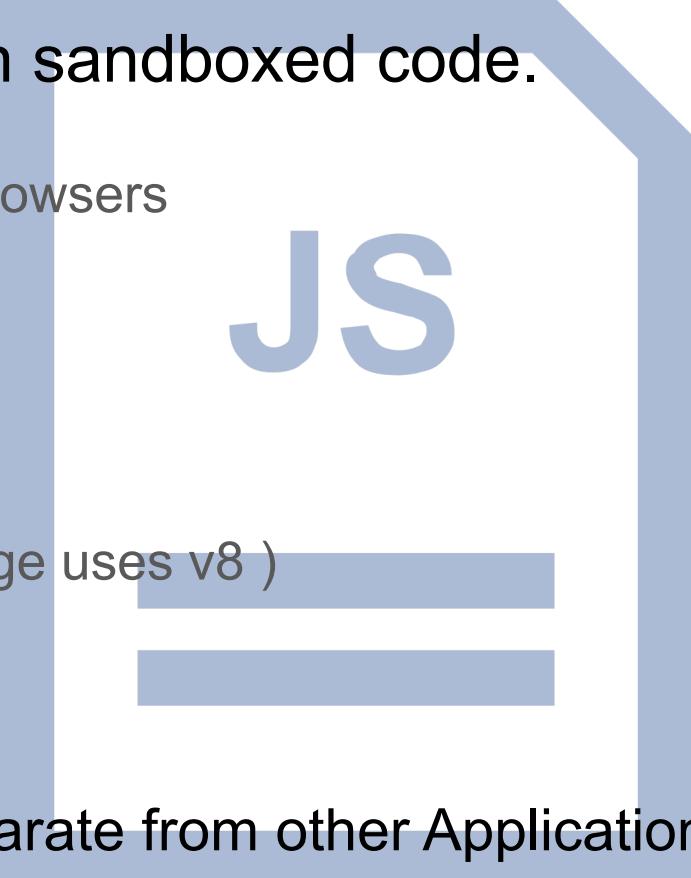
JavaScript in a scripting language

Powered by v8 Engine (Google), JavaScriptCore (Webkit)

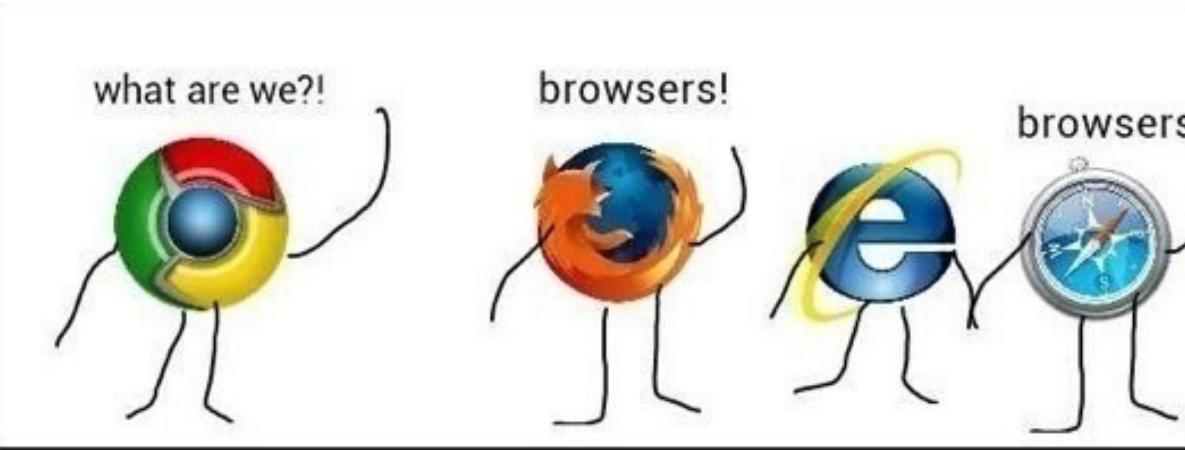
SpiderMonkey (Mozilla), Chakra (Internet Explorer – MS Edge uses v8)

JavaScript is not related Java!

Code runs client side and is executed by the browser separate from other Applications.

A large, stylized blue logo consisting of the letters "JS" in a bold, sans-serif font. The logo is set against a light blue background that has a dark blue vertical bar on the left and a dark blue horizontal bar at the bottom, creating a shape reminiscent of a speech bubble or a window frame.

JS



What is JavaScript

Inline vs External code

```
<script type="text/javascript">
    alert("Hello, world");
</script>
```

Inline code is called in between script tags.

What is JavaScript

Inline vs External code

```
<script type='text/javascript' src='https://code.jquery.com/jquery-3.5.1.js'></script>
<script type="text/javascript" src='https://maps.googleapis.com/maps/api/js?key=YOUR_API_KEY'> </script>
<script type='text/javascript' src='https://momentjs.com/downloads/moment.js'></script>
```

External scripts are referenced as links directly to JavaScript files on the web.

What is JavaScript

Inline vs External code

```
.....  

```

JavaScript Basics

Types, Language and Features

Language Features

Asynchronous Language

Dynamically Typed or Loosely typed

Easy Syntax – Everything is an object

Event Driven

Libraries Available

Many Libraries available to solve many problems.

Utilise these libraries to extend your code.

Stackoverflow and GitHub are your friends.

!! JavaScript is not Java !!

JavaScript Basics

What every good web developer should know

JavaScript Basics

Comments

```
// I am a comment - I will not run  
var imAVariable = "I do run";
```

Comments start with // - Always comment your code.

JavaScript Basics

Variables and Math

```
var foo = 5;  
var apples = 10;  
var pears = 20;  
var title = "Learning JavaScript is fun";
```

Variables have to be prefixed with keyword var.
Lines end with semi-colons.

JavaScript Basics

Variables and Math

```
var foo = 5;  
var apples = 10;  
var pears = 20;  
var title = "Learning JavaScript is fun";
```

```
var total = apples + pears;  
var number = 10/3;  
var whatAmI = 10/0;
```

All basic operators are valid. (e.g. + - / * = ^)

JavaScript Basics

Output variables and Debugging

```
var foo = 5;  
var apples = 10;  
var pears = 20;  
var title = "Learning JavaScript is fun";  
console.log(foo);
```

```
var total = apples + pears;  
console.log("Total # of fruit is: " + total);
```

Writes out values to the JavaScript Console.

JavaScript Basics

Strings

```
var title = "Learning JavaScript is boring.";  
var end = "Go forth and learn. ";
```

```
var newString = title + end;  
newString.replace("boring ", "fun");  
title.replace("boring", "fun").trim();
```

Functions can be applied to any object.

Use variable reference to find out all functions available to an object.

JavaScript Basics

String Concatenation

```
var url = "wa.celabs.org";
var endPoint = "maps";

console.log("https://" + url + "/" + endPoint + "/page" + "/" + 5 + ".html");
```

Result: <https://wa.celabs.org/maps/page/5.html>

JavaScript Basics

Equality and if statements

```
if(true){  
    console.log("This happens if variable is true");  
}else{  
    console.log(" This happens if variable is false");  
}
```

Let's Play Game of Truth

$10 \geq 5$	true	$"a" == 'a'$	true
$100 < 10$	false	$"A" === a$	false
$10 === (6+4)$	true	$"10" == (6+4)$	true
$10 != 5$	true	$\text{Infinity} == \text{Infinity}$	true
$"" == 0$	true	$2b !2b$	true

JavaScript Basics

Loops

```
var i = 1;  
for(var k=1; k<=10; k++){  
    console.log(i);  
    i = i + 1;  
}  
// What is i?
```

For Loop

```
var i = 1;  
while (i < 10) {  
    console.log(i);  
    i = i + 1;  
}  
// What is i?
```

While Loop

JavaScript Basics

Functions

```
imAFunction();
doSomething(variable);

function addMe(a, b){
    return(a+b);
}

var result = addMe(3,2);
```

Functions return values and are used to group code together.
If you have to do something more than once, write a function!

JavaScript Basics

Arrays

```
var emptyArray = [];
```

```
var shoppingList = ['Milk', 'Bread', 'Beans'];
```

```
shoppingList.length;  
shoppingList.push('Cheese');  
shoppingList.sort();
```

Arrays store objects in an object and each item can be of any type.

JavaScript Basics

Objects

```
var jedi = {  
    name: "Yoda",  
    age: 899,  
    talk: function () { console.log("may the force be with you"); }  
};
```

```
jedi.name;  
jedi.age;  
jedi.talk();
```

Objects are collections of variables

JavaScript Basics

Scope

```
var a = 10;  
  
function add2(a){  
    var b = 2;  
    return (a+b);  
}  
  
var c = a + b;
```

Uncaught ReferenceError: b is not defined
at <anonymous>:8:13

What is the value of c?

JavaScript Basics

Scope

```
var a = 10;  
var b = 3;
```



```
function add2(a){  
    var b = 2;  
    return (a+b);  
}
```



```
add2(4);  
var c = a + b;
```

What is the value of c now?

JavaScript Basics

Scope

```
var a = 10;  
var b = 3;  
  
function add2(a){  
    var b = 2;  
    return (a+b);  
}  
  
a = add2(4);  
var c = a + b;
```



What is the value of c now?

JavaScript for the Web

Combining JavaScript code and HTML Elements

Making JavaScript Easier

Using libraries to help us to code

```
<script type='text/javascript' src='https://code.jquery.com/jquery-3.5.1.js'></script>
```

jQuery is a library that exposes extra functions to make life a bit easier for the web developer.

Learning jQuery

What does it do?

```
jQuery("#title").html("Hello World");  
jQuery("a").css("color", "red");
```

```
$(".section").hide();  
$(".menuItems").hide();
```

Note:
The \$ symbol is just shorthand for the jQuery variable. You will get used to this in various libraries.

jQuery works with Selectors - it selects elements and you can edit them

<https://api.jquery.com>

Learning jQuery

Waiting until the page is ready

```
$(document).ready(function() {  
    doSomething();  
});
```

This function waits until the DOM has loaded and then executes our functions
Why do you think we need this?

Learning jQuery

Functions to make web development easier

```
var shoppingList = ['Milk', 'Bread', 'Beans'];

for(var i=0; i < shoppingList.length; i++){
    console.log(shoppingList[i]);
}
```

Standard For Loop, to walk (or iterate) over an array.
Written in Pure JavaScript.

Learning jQuery

Functions to make web development easier

```
var shoppingList = ['Milk', 'Bread', 'Beans'];

$.each(shoppingList, function(key, value){
    console.log(value);
});
```

jQuery.each – Easy loops
<https://api.jquery.com/each>

Learning jQuery

Functions to make web development easier

```
$("#title").html("Hello World");
```

jQuery.html – Writes Contents into the DOM

<https://api.jquery.com/html>

Learning jQuery

Functions to make web development easier

```
$("button").click(function(){
    var div=$("#div");
    div.animate({height: '300px', opacity: '0.4'}, "slow");
    div.animate({width: '300px', opacity: '0.8'}, "slow");
    div.animate({height: '100px', opacity: '0.4'}, "slow");
    div.animate({width: '100px', opacity: '0.8'}, "slow");
});
```

jQuery.animate – Can Chain Animations to make your site shine.

<https://api.jquery.com/animate>

Workshop: Week 2

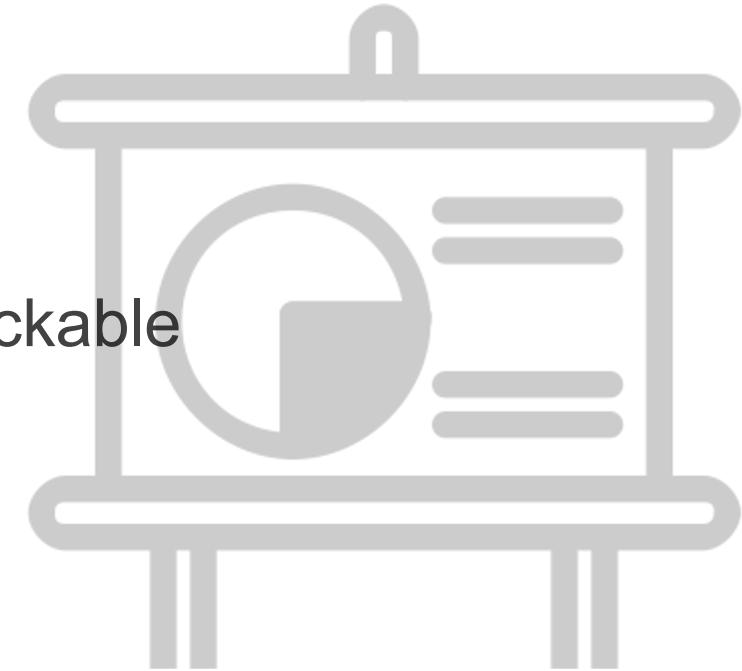
Make your static website, interactive

- **Build an Interactive Viewer for Flickr Photos**

- Load Interactive Google Map
- Get some data from an API
- Draw markers on the map and make them clickable

- **JavaScript Skills**

- Experiment with Developer Tools
- Learn simple commands and libraries to make JS easier



Next Week

Learn about JavaScript and Interactive Web

3

Client-side programming: JavaScript II

Learn more Advanced JavaScript interacts with the DOM to provide an interactive experience on a static page

Any Questions?

Steven Gray
steven.gray@ucl.ac.uk
@frog0
<https://www.sjg.dev>

