Front-end <Web /> Development

# Lesson 7: JS (Part 1)

University of Tehran

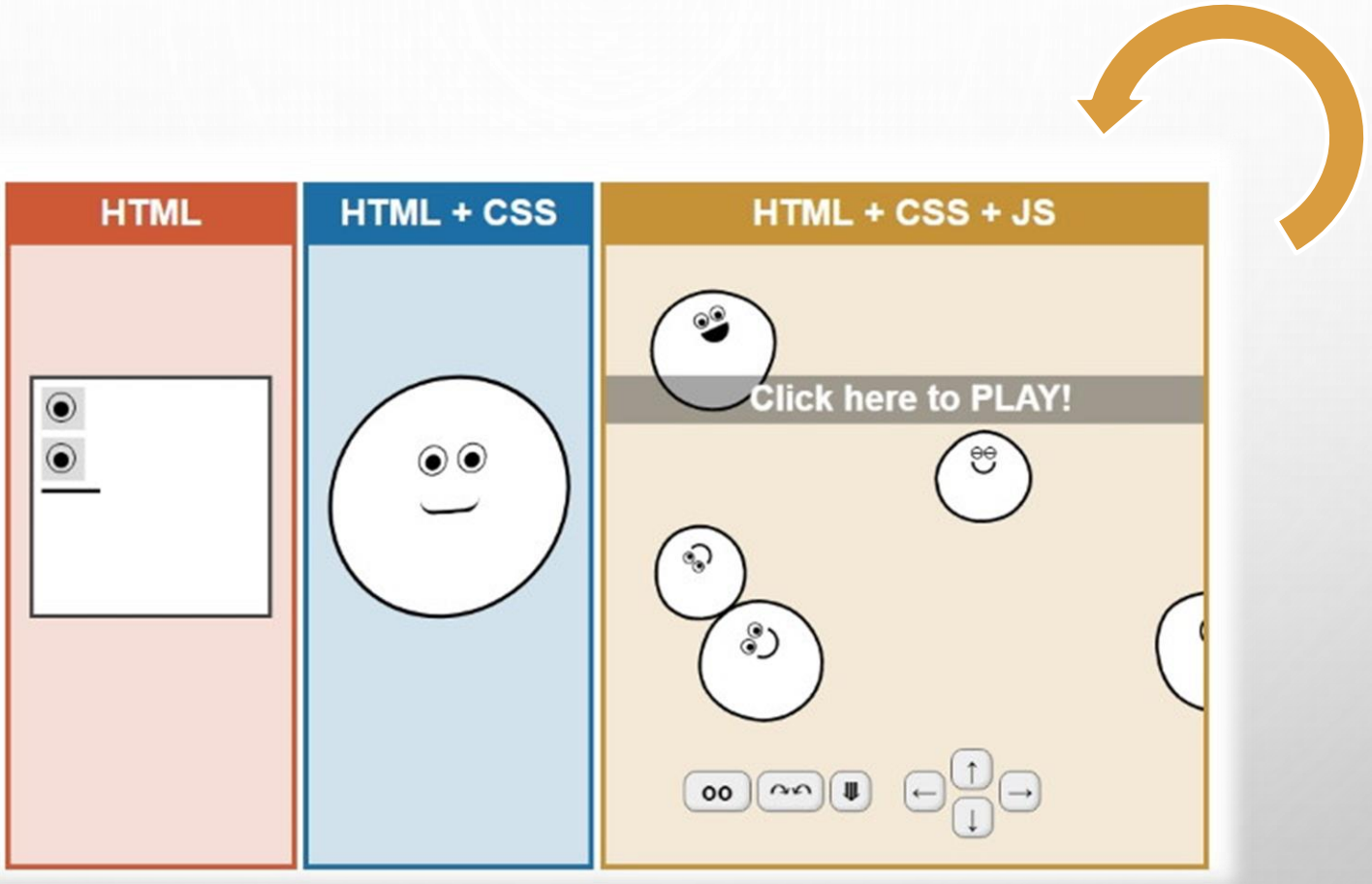
ACM
Summer of Code 2024

Lecturer:
Misagh Mohaghegh

# A Short Review…

- Responsive Web Design
- Viewport Meta
- Media Queries
- Breakpoints
- Desktop vs Mobile first design
- Fixed width container
- Bootstrap
- Reboot, breakpoints, containers
- Responsive Grid
- Margin & Padding (.pt-2)

# Time for…



# JavaScript

# JavaScript

# JavaScript

The primary tool to program the behavior of web pages.

- Created by Brenden Eich in 1995, became an ECMA standard in 1997.
- JavaScript and Java are completely different languages.
- ECMAScript is the name of the language standard, JavaScript is the implementation.
- JavaScript is an interpreted programming language.
- JS can be used outside of the browser using the Node.js environment.

# Version History

Version History: (ECMAScript)

- **ES1** (1997): The initial release.
- **ES1** (1998): Editorial changes.
- **ES3** (1999): Regex, try/catch, switch, do-while.
- **ES4** (does not exist)
- **ES5** (2009): "use strict", array and string methods, JSON support.
- **ES6** / **ECMAScript 2015**: let and const, arrow functions, classes, promises, template literals.
- **ECMAScript 2016**: exponential ** operator.
- **ECMAScript 2017**: …
- A new version every year.

# Version History

Version History: (ECMAScript)

These can be categorized as follows:

- The original JavaScript **ES1, ES2, ES3** (1997-1999)
- The first main revision **ES5** (2009)
- The second revision **ES6** (2015)
- The yearly additions **ECMAScript 20XX** (2016-2024…)

# Adding JavaScript

We can use the <script> tag to add JavaScript to our page.

This can be internal or using an external file.

The <script> tag is usually put at the end of the <body> element, this is to make sure the entire DOM is loaded and JS can see the whole HTML page.

```
<body>
    <!-- Document here -->
    <script src="file.js"></script>
</body>
```

# Adding JavaScript

We can also put the script in the <head> element and use the **defer** attribute.

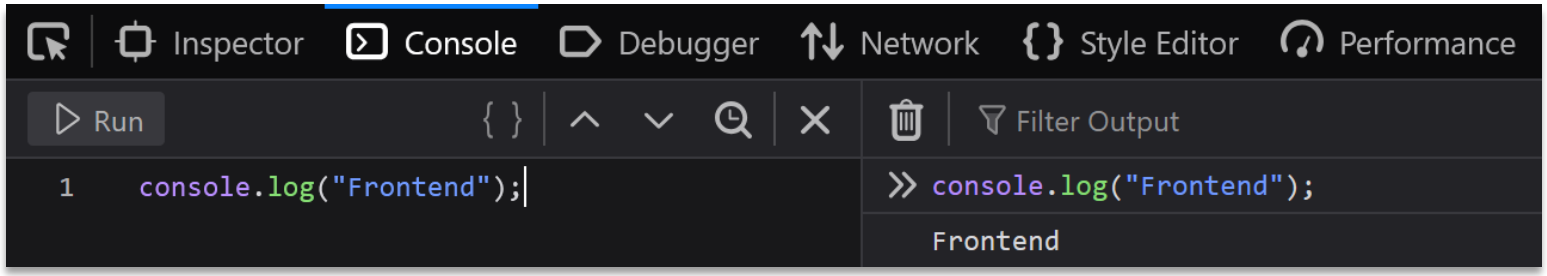
This makes the script only execute after the HTML is parsed.

This is almost the same as putting the <script> element at the end of body, but requires newer support.

```
<head>
    <!-- Metadata here -->
    <title>Test</title>
    <script src="file.js" defer></script>
</head>
```

# The Browser Console

In the browser dev tools (the "inspect" panel) there is a JavaScript console that we can write JS code and also check some outputs.




The **console.log(...)** function takes a parameter and prints it in the dev tools console.

# The Node.js Environment

Node.js is a JavaScript runtime environment that allows running JS outside of a browser.

This allows us to use JavaScript as a programming language not just for DOM manipulation, but things such as algorithms, creating web servers, scripts, and more.

```
C:\Users\Lenoidea\Desktop>node
Welcome to Node.js v20.16.0.
Type ".help" for more information.
> console.log("Hi");
Hi
undefined
>
```

# TypeScript

JavaScript is a dynamically typed language, which means a variable can be a number or a string at any time.
(x = "test" and x = 10)

TypeScript (TS) is a superset of JavaScript developed by Microsoft.
A programming language that adds static typing and some more features to JavaScript.
This makes your code less error prune
TypeScript transpiles to JavaScript which can be ran in browsers.

# Babel

Babel is a JavaScript compiler that turns JS code into JS code that does not use newer features.
Each year, a new version of JavaScript is released which adds some features.
The browsers need to update to add support for those features.

If we just start using newly added features, a lot of people who have not yet updated their browser cannot process our JS code and our website will not work for them.

Babel allows us to use new JS features to program, and compiles our code into a code that does not use those features. They get replaced by widely available features to make sure our website works for most.

# The Programming Language

# Comments

The comments for web technologies are as follows:

```
<head>
    <!-- HTML Comment -->
    <style>
        /* CSS Comment */
    </title>
    <script>
        // JS Comment
        /*
        JS multi-line
        comment (not used as much)
        */
    </script>
</head>
```

# Variables

JavaScript is a dynamically typed programming language.

To create a variable, we can use the **var**, **let**, or **const** keywords.

let and const were added in ES6 and provide local-scope variables like most languages.

```
var x; // function-scope variable declaration
let y; // local-scope variable declaration
y = 5; // variable assignment
let z = "test"; // declare and assign
const a = "const"; // const variable
```

# Variable Types

There are 5 JavaScript variable types:

```
let a = "string";   // String
let a = 'string';
let b = 10;         // Number
let b = 4.12;
let c = true;       // Boolean
let c = false;
let d = [2, "str"]; // Array
let d_first = d[0]; // Array element access
let e = { name: "test", done: true } // Object
```

# Operators

We have all the basic programming operators:

```
let a = 2 + 8; // also -, *, /, %, **
let b = 'hello ' + 'world';
a += 4; // also -=, *=, ...
a++; // also a--;
a == b; // equality check (value)
a === b; // equality check (value and type)
a != b; // also !==
a > b; // also >=, <, <=
if (a && b) // also ||
a & b // also |, ~, ^, <<, >> (bitwise)
```

# Code!

It is best to learn a programing language in code.

```
CODE!
```

Thank you for your attention.