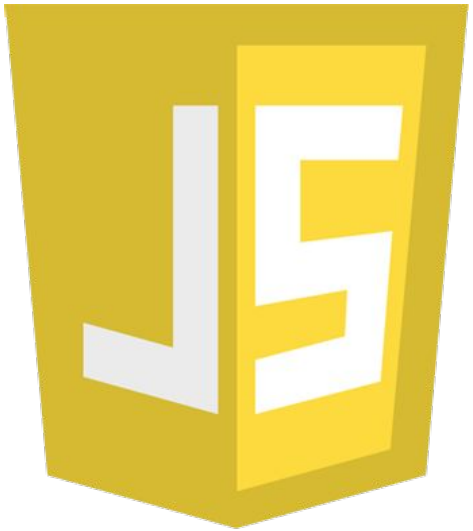
Front End Development
Module 3 - Day 6



Introduction to JavaScript

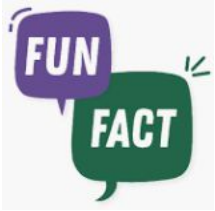
Can You?**

- Compare and contrast the use and nature of JavaScript Variables with those in Java/C#
- Describe the JavaScript Data Types, Falsy & Truthy
- Select and use JavaScript Operators and Arithmetic
- Perform basic Type Conversion tasks
- Declare and manipulate objects in JavaScript

Welcome to a dynamically-typed language!

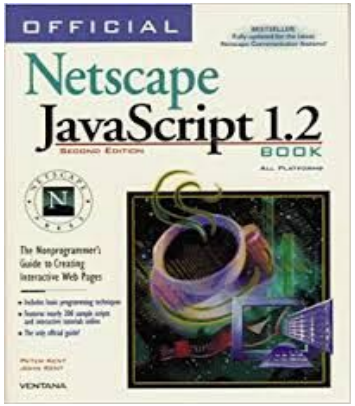
When is an int not an int? When it's JavaScript and it gets reassigned a string!

***Please refer to the LMS & W3Schools for even more JavaScript ins & outs*



We Know some Java...

The names are almost the same, so the languages are, too, right?



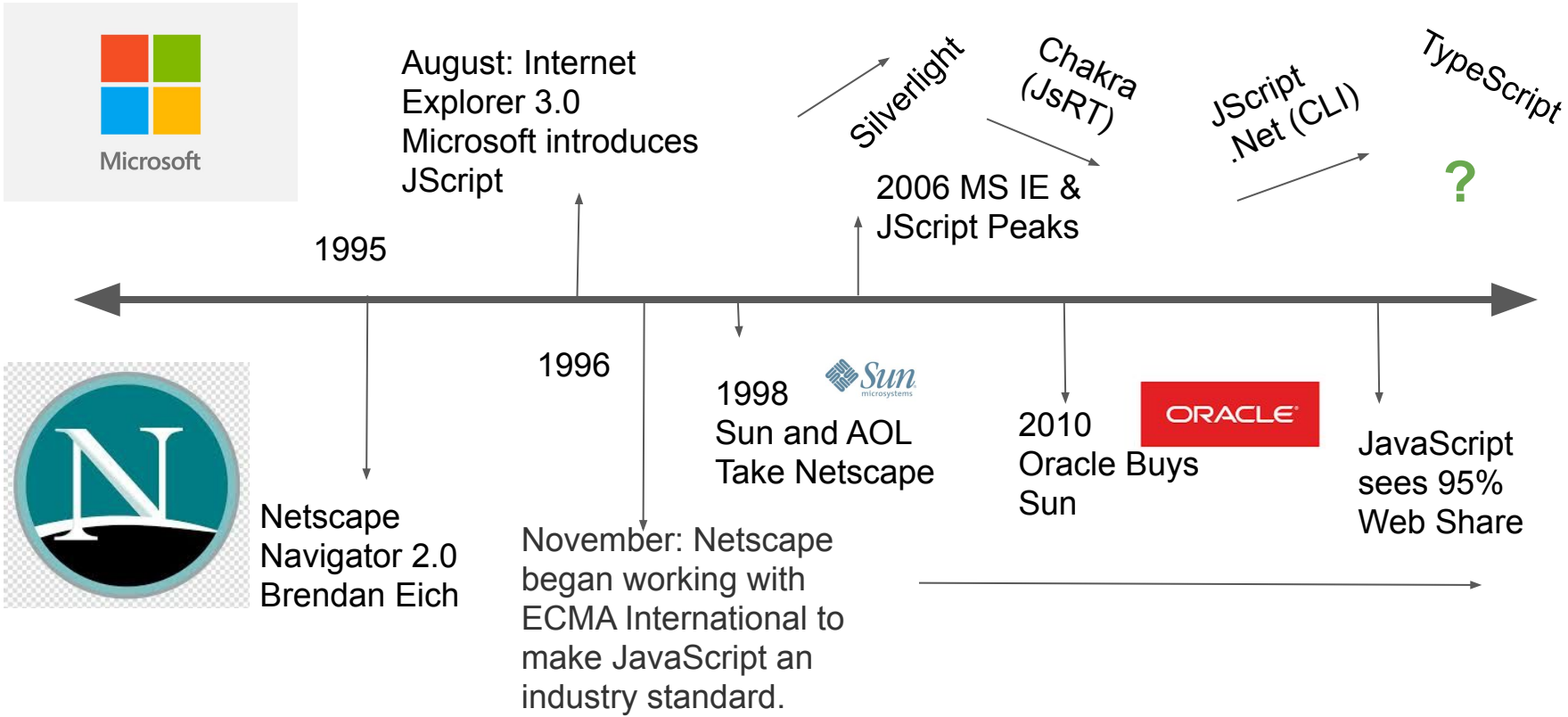
Netscape created Mocha, later renamed it to LiveScript, and then....

It was renamed to JavaScript when Netscape and Sun hatched a marketing plan, so the Sun Java name was carried forward by JavaScript.

RIP Netscape…
I remember you fondly.

...No!

Client Side Browser Scripting Timeline



JavaScript: Variable Name Rules

- Valid characters:
  - A-Z a-z
  - 0-9
  - underscore _
  - dollar sign $
- Must start with a letter, _, or $
- Are case-sensitive
- May not be a reserved keyword
- Camel-Case
- Caps with Underscore if Constant

Yes!	No!
userName	UserName
option06	06option
upsell_2	upsell-2
$UserAccount **	

**Note that the $ can also serve as a function shortcut for various JQuery functions. Consider avoiding using it for variables for clarity. More on that in another lecture… maybe

JavaScript: Declaring Variables

- Keywords: **let, const, var**
- Example: **let firstName = "Bill";**
  const SLICE_OF_PI = 3.1415926535
- Use **let** when the value needs to be changeable
- Use **const** when the value must not change after it's assigned
- Avoid using **var**
  - It was used a lot in the past
  - It has complicated scope considerations (maybe more on that later)
  - See Hoisting in the Textbook and on MDN for more (https://developer.mozilla.org/en-US/docs/Glossary/Hoisting)

JavaScript: Data Types

- Number
- String
- Boolean
- Symbol
- Objects
  - Function
  - Array
  - Date
  - RegEx

- null
- undefined
- NaN

MDN Number and Date Reference

Loosely Typed

- JavaScript is a loosely typed language
- We don’t specify a data type when declaring variables
- The data type is inferred when we assign it
- The data type of a variable can change

A value's data type also affects the operations that are valid on that value. For example, a value of type number can be multiplied by another number, but not by a string - even if that string contains *only* a number, such as the string "2".

If you are unsure of the type of a value, you can use the typeof operator.

Boolean Logic - Truthy and Falsy

Falsy: a value that is considered false in a Boolean context

- **false** The keyword false
- **0, -0** The number zero
- **0n** BigInt value 0
- **"", '', ``** Empty string(s)
- **null** Nulls
- **undefined** Declared and unassigned
- **NaN** not a number

Truthy: a value that is considered true in a Boolean context

All values are **truthy unless falsy**

```
1  if (true)
2  if ({})
3  if ([])
4  if (42)
5  if ("0")
6  if ("false")
7  if (new Date())
8  if (-42)
```

Boolean Logic - Truthy and Falsy: Null & Undefined

When checking for `null` or `undefined`, beware of the differences between equality (==) and identity (===) operators, as the former performs type-conversion.

(https://developer.mozilla.org/en-US/docs/Web/JavaScript/Reference/Global_Objects/null)

```
typeof null          // "object" (not "null" for legacy reasons)
typeof undefined     // "undefined"
null === undefined   // false
null  == undefined   // true
null === null        // true
null == null         // true
!null                // true
isNaN(1 + null)      // false
isNaN(1 + undefined) // true
```

Lecture Code:
Intro to JavaScript