



JavaScript



Redes y Sistemas Distribuidos

Juan A. Fraire



JS - Introducción

- Capa de aplicación
 - Paradigma **cliente/servidor** - servicios **web**: recursos accesibles vía URL - **JS, CSS y HTML**
- JavaScript (**JS**)
 - Se origina para dar páginas web interactivas del lado **cliente** (browser, front-end)
 - Hoy usado en lado **servidor** también (Node.js, back-end)
 - Alto nivel
 - Compila en **tiempo de ejecución** (típicamente)
 - Multi-paradigma: **orientado a eventos**, **funcional**, e **imperativo**
 - Sintaxis basada en **{ }** y **;**
 - Gran cantidad de **APIs**
 - I/O de red, almacenamiento y gráficos provistas por browser
 - **Tipado dinámico** (ver **Typescript** para tipado estático)

Lenguajes

Java?

A pesar de tener “Java” en su nombre, poco tiene que ver con JS. Fué sólo una movida de marketing de su creador

JS - Bases (1/3)

- JS en HTML

- `<script type="text/javascript">...</script>`

- Importar JS externo

- `<script src="filename.js"></script>`

- Funciones

- ```
function sumar(a, b) {
 return a + b;
}
x = sumar(1, 2);
```

- Edición del DOM

- ```
document.getElementById("elementID").  
innerHTML = "Hello World!";  
// actualiza a un elemento HTML:  
<p id="elementID">"Texto viejo"</p>
```

- Eventos

- `<button onClick="myFunction();">
 Click here
</button>`
- Otros eventos: onmouseover, onkeydown, onkeypress, onload, onfocus, oncopy...

- Output

- `console.log(a);` // consola browser
- `document.write(a);` // escribe al HTML
- `alert(a);` // cartel alerta
- `confirm("Really?");` // dialogo si/no
- `prompt("Age?", "0");` // dialogo input

JS - Bases (2/3)

- Tipo de datos

- `var age = 18;` // número
- `var name = "Jane";` // string
- `var truth = false;` // boolean
- `var a = ["a","b"];` // array
- `var a; typeof a;` // undefined
- `var a = null;` // value null
- `var student = {` // objeto
 - `first:"Jane",` // propiedades
 - `last:"Doe",`
 - `fullName : function() {`
 - `return this.first + this.last; }`
- `};`
- `student.last = "Smith";`
- `student[last]+= " II";`

- Variables

- `var c = "Hi" + " " + "Joe";` // "Hi Joe"
- `var d = 1 + 2 + "3";` // "33"
- `var h = function(){};` // funcion
- `const PI = 3.14;` // constant
- `let z = 'zzz';` // scope local

- Valores

- `false, true` // boolean
- `18, 3.14, 0b101, 0xF6, NaN` // number
- `"flower", 'John'` // string
- `undefined, null , Infinity` // special

JS - Bases (3/3)

- Condicionales

- `if ((age >= 14) && (age < 19)) {
 status = "Eligible."
} else {
 status = "Not eligible."
}`
- `switch (new Date().getDay()) {
 case 6: // if (day == 6)
 text = "Saturday";
 Break;
 case 0: // if (day == 0)
 text = "Sunday";
 Break;
 default: // else...
 text = "Whatever";
}`

- Loops

- `html = "";
for (var i of custOrder) {
 html += "" + i + "";
}`
- `var i = 1;
while (i < 100) {
 i *= 2;
 document.write(i + ", ");
}`
- `var i = 1;
do {
 i *= 2;
 document.write(i + ", ");
} while (i < 100)`
- Instrucciones `break`; y `continue`;

JS - Ejemplos

- **Caso 1:** "Hola mundo" en consola
- **Caso 2:** "Hola mundo" en browser
- **Caso 3:** Variables y estructuras:
 - <https://www.youtube.com/watch?v=Q9fwkpxr3Dw>
- **Case 4:** Actualización de contenidos en browser

JS - Actualidad

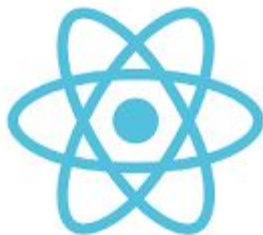


Angular

Front-end framework (MVC)

Open-source

Operated by Google
TypeScript-based
Single Page Apps (SPA)
Two-way data binding
(Apple, Nike, MS)



React

UI library (View only)

Open-source

Created by Facebook
High-incoming traffic
Virtual DOM
React Native
(Tweeter, Netflix)



Node.js

Back-end

Open-source

JS to CLI
High-tput async I/O
Real-time web
(PayPal, Netflix, LinkedIn)

