



JAVASCRIPT

PROGRAMACIÓN III - TUP UTN FRLR



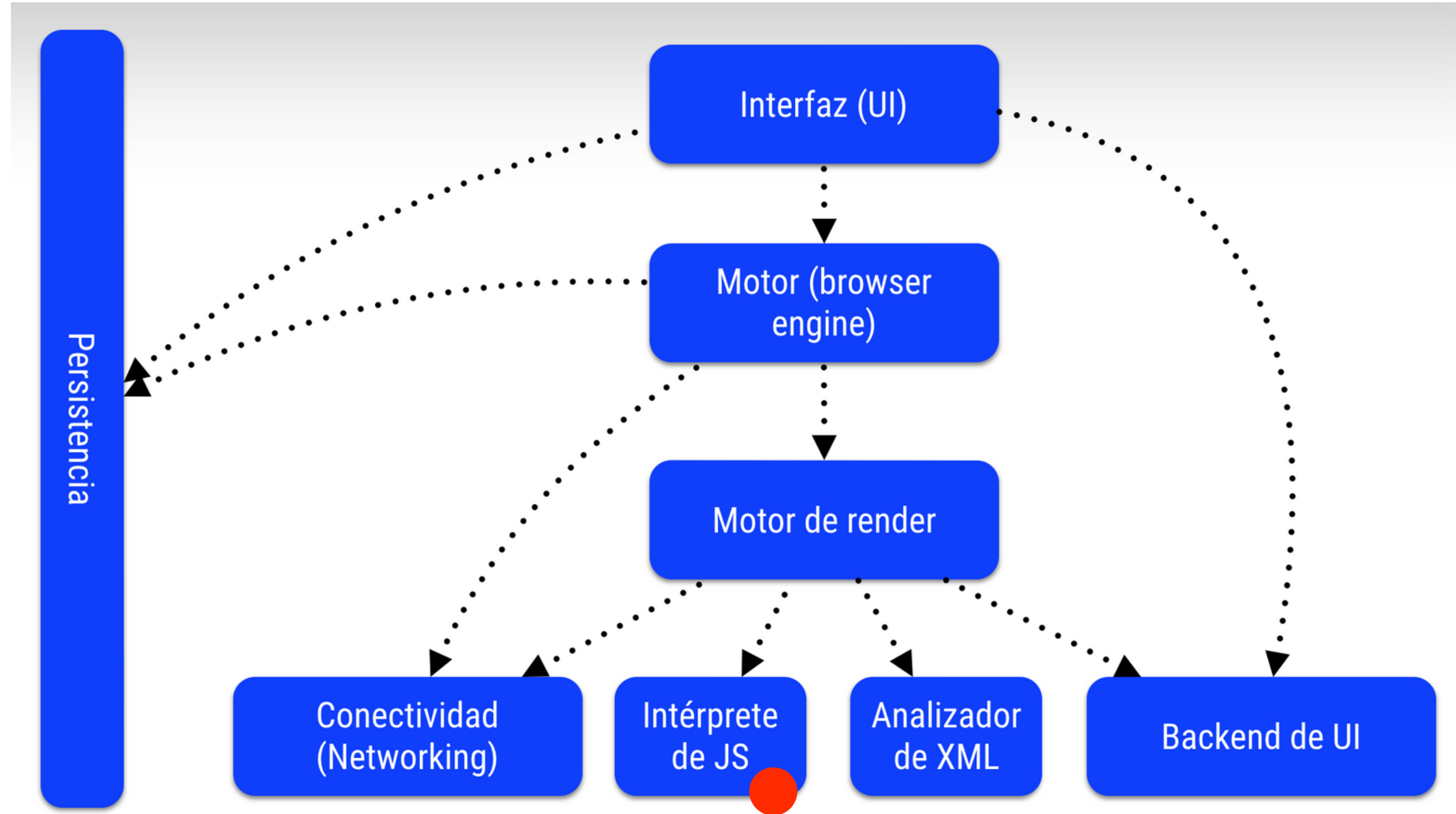


¿Qué es JavaScript?

JavaScript es un lenguaje de programación de scripts (secuencia de comandos) orientado a objetos.

Los Scripts corren sobre el navegador ó browser quien tiene un **INTÉRPRETE** para poder ejecutarlos.

Cada browser tiene su propio intérprete.

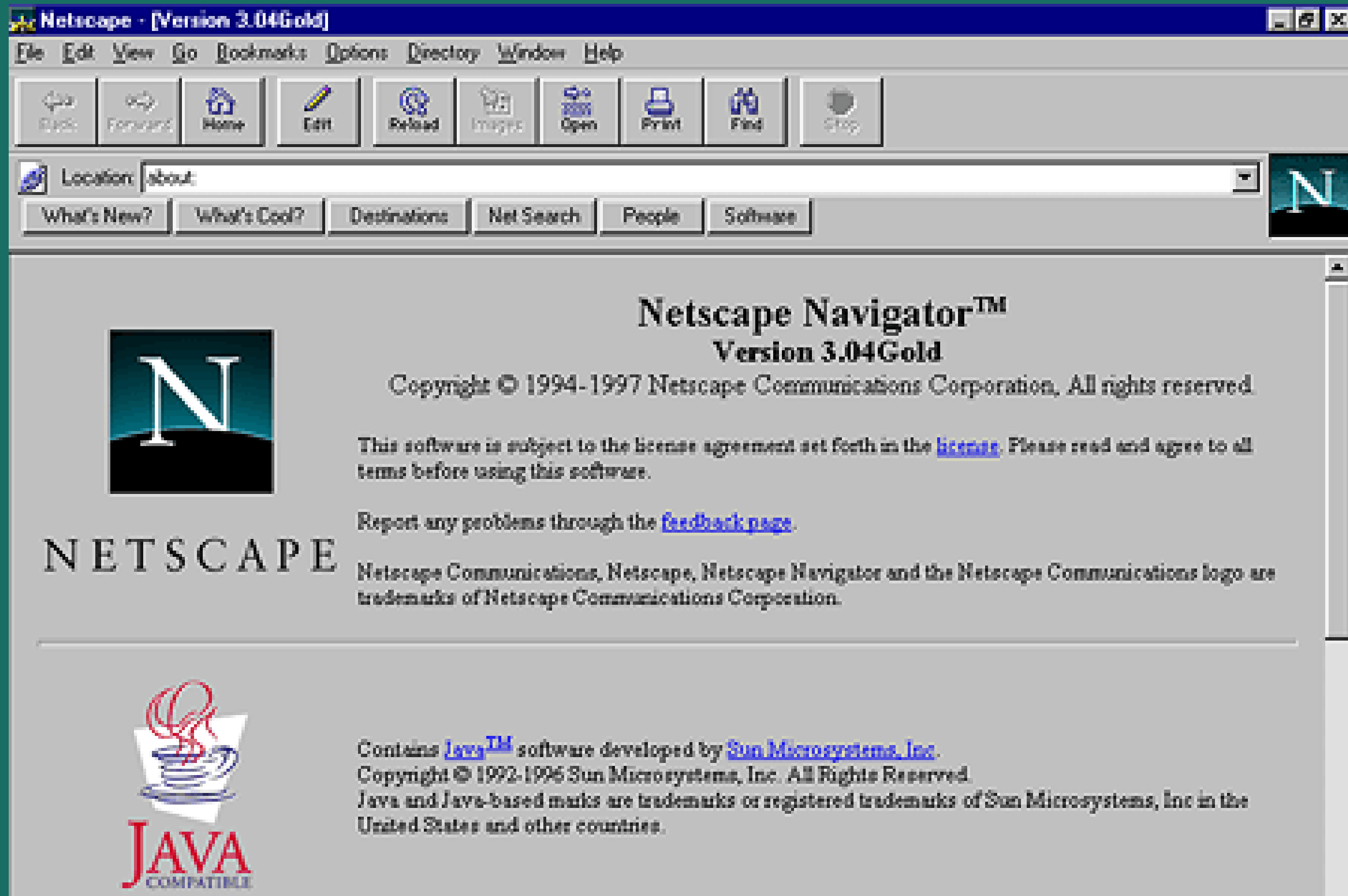


NACIMIENTO DE JS

JavaScript fue desarrollado originalmente por Brendan Eich de Netscape con el nombre de **Mocha**, el cual fue renombrado posteriormente a **LiveScript**, para finalmente quedar como **JavaScript**.

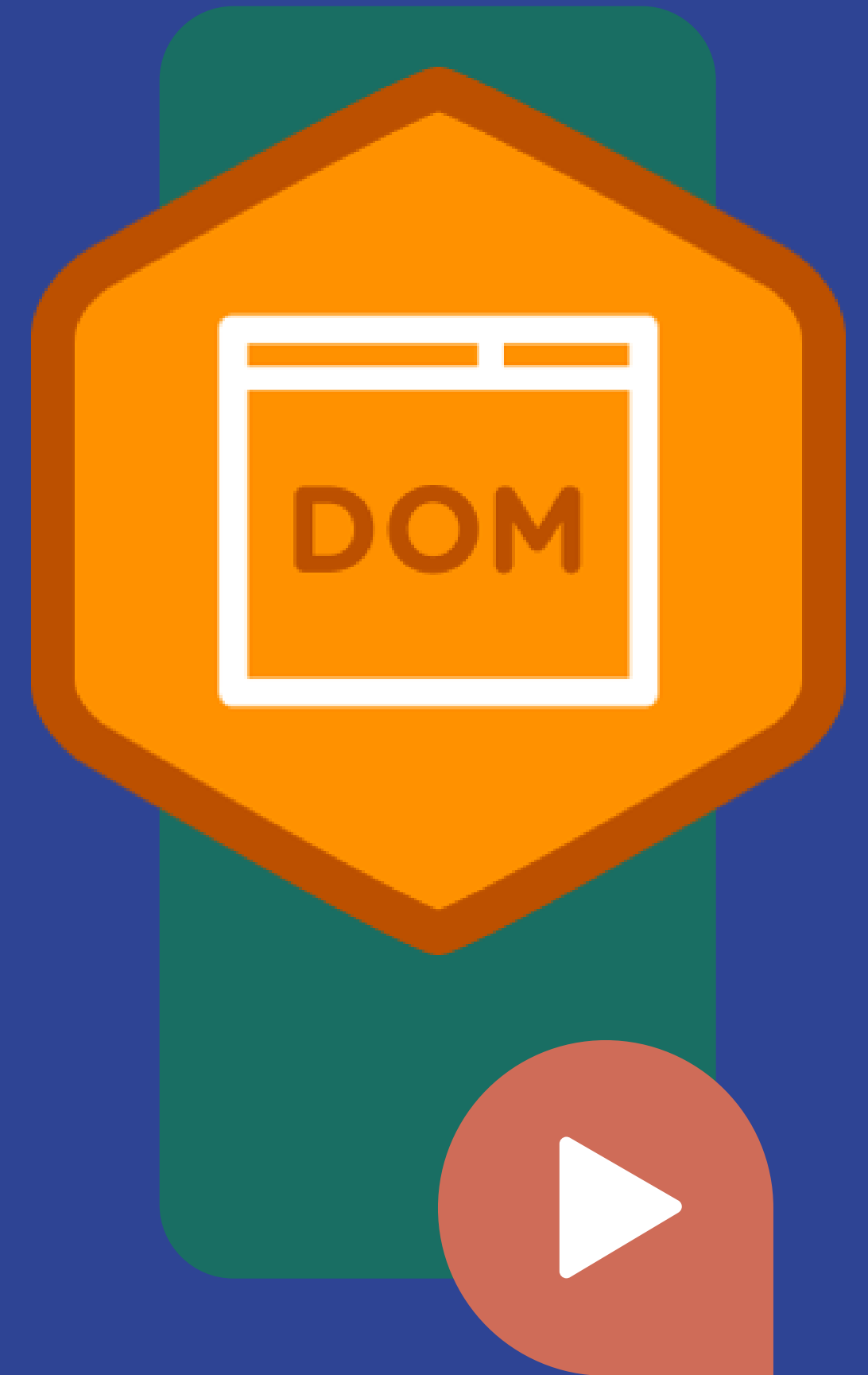
La denominación produjo confusión, dando la impresión de que el lenguaje es una prolongación de Java, y se ha caracterizado por muchos como una estrategia de mercadotecnia de Netscape para obtener prestigio e innovar en el ámbito de los nuevos lenguajes de programación web.

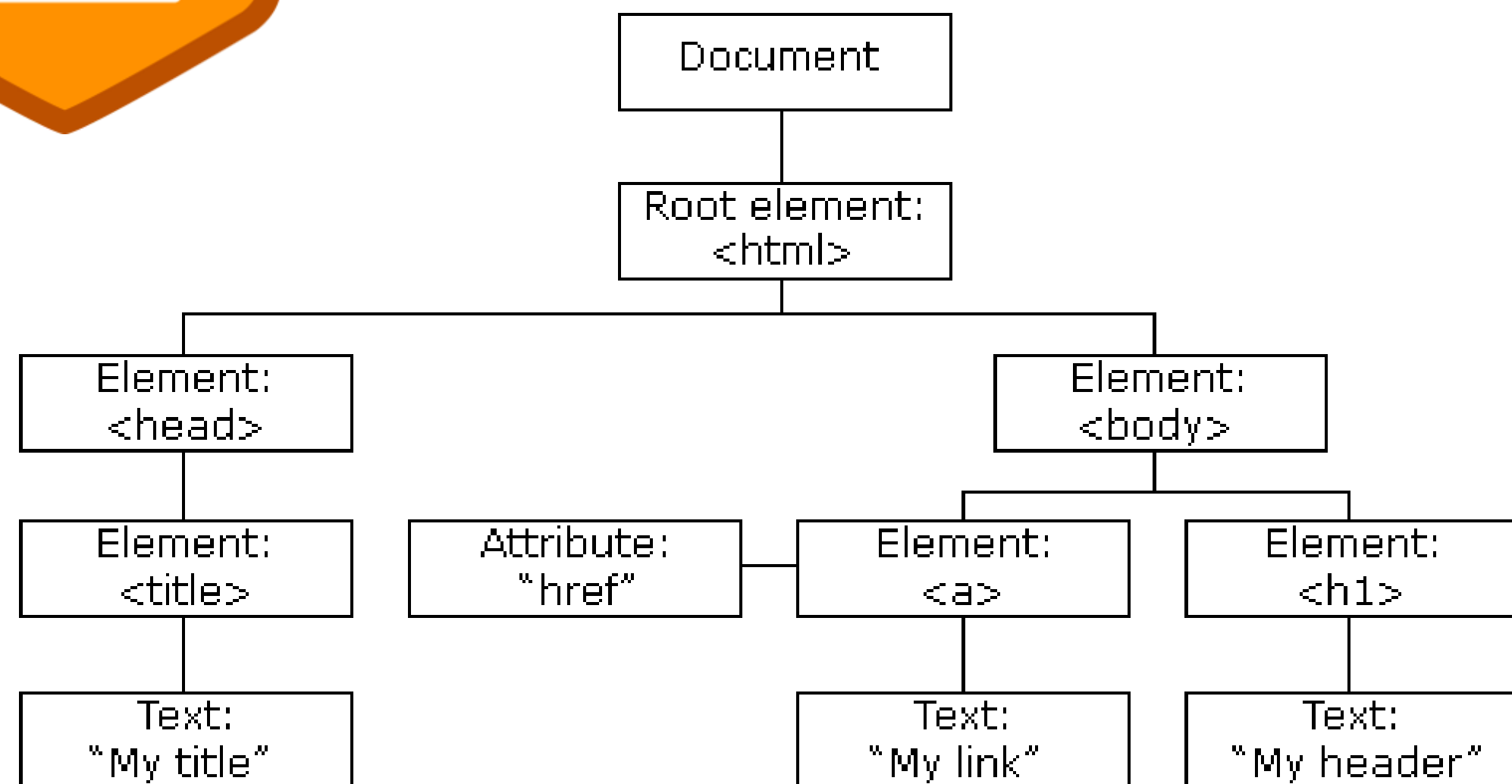




DOM - Standard

Para evitar estas incompatibilidades, el World Wide Web Consortium diseñó el estándar **Document Object Model** (DOM, o Modelo de Objetos del Documento en español), que incorporan Konqueror, las versiones 6 de Internet Explorer y Netscape Navigator, Opera la versión 7, Mozilla Application Suite y Mozilla Firefox desde su primera versión.





Gracias al **DOM**, Javascript puede acceder y modificar todos los elementos de un documento HTML. Para ello, el navegador crea una representación interna de la página, un árbol de nodos.



ECMAScript

En 1997 los autores propusieron **JavaScript** para que fuera adoptado como estándar de la **European Computer Manufacturers Association ECMA**, que a pesar de su nombre no es europeo sino internacional, con sede en Ginebra. En junio de 1997 fue adoptado como un estándar ECMA, con el nombre de **ECMAScript**. Poco después también como un estándar ISO

ECMAScript 6

JS



VERSIONES DE JS

Las versiones de Javascript se basan en las de la ECMAScript (que abreviaremos como ES). Por lo tanto, se encuentran:

- **ES 1 y ES 2**, que son los inicios de Javascript.
- **ES 3** (publicada en diciembre de 1999), que es funcional en todos los navegadores (excepto las versiones anteriores de Internet Explorer).

ECMAScript 6

JS



VERSIONES DE JS

- **ES 4**, que fue abandonada debido a los grandes cambios que no fueron apreciados.
- **ES 5** (publicada en diciembre de 2009), que es la versión más reciente liberada.
- **ES 6** (publicada en 2015) Es la versión actual que corre en los navegadores modernos.

ECMAScript 6

JS



EN RESUMEN



- JavaScript es un lenguaje de programación interpretado, es decir, que necesita un intérprete para ser ejecutado.
- JavaScript se utiliza principalmente en páginas web.
- Javascript está normalizado por ECMA International como el nombre ECMAScript Language Reference.
- Hay otros lenguajes derivadas del ECMAScript como ActionScript, EX4 o JScript.NET.
- La última versión del estándar está basado en ECMAScript 6, lanzado en 2015.

HTML

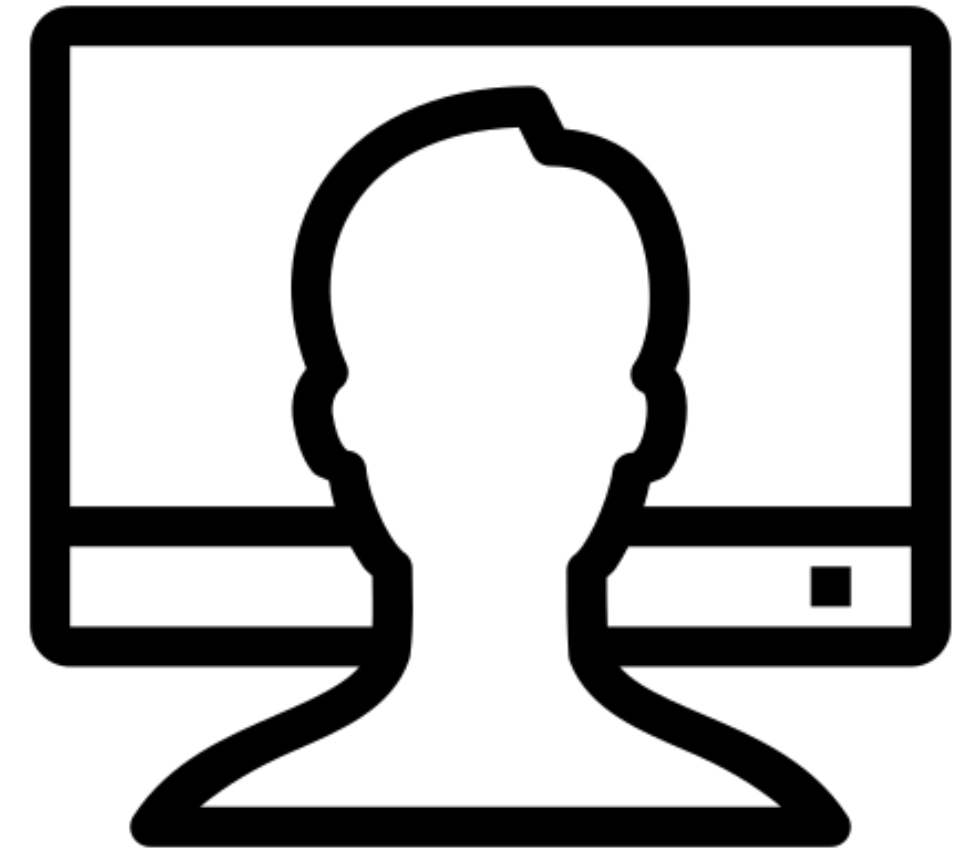


HTML Y JAVASCRIPT

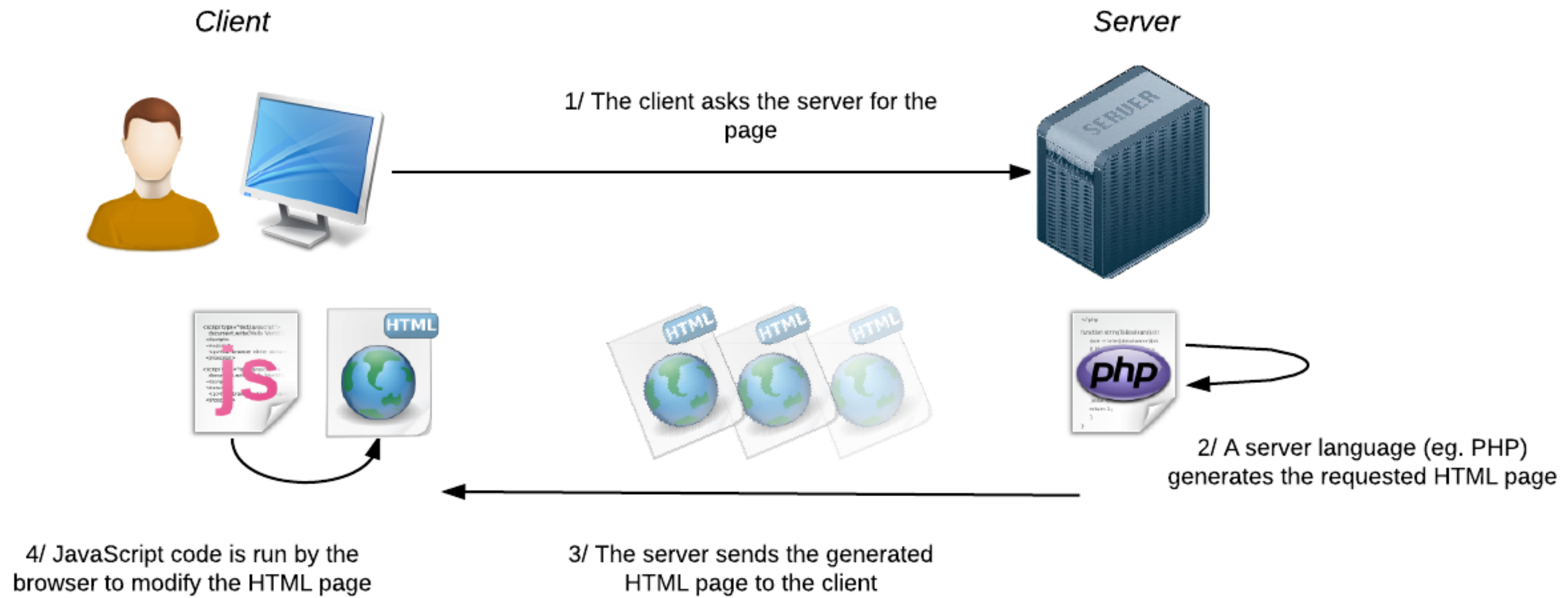
Javascript actualmente es principalmente utilizado en internet, junto con las páginas web (HTML o XHTML). Javascript está directamente incluido en la página web (o en un archivo externo) y mejora una página HTML, añadiendo interacción del usuario, animación, ayudas a la navegación, tales como:

- **Mostrar / ocultar el texto.**
- **Deslizamiento de imágenes.**
- **Crear presentaciones de diapositivas.**
- **Crear burbujas de información.**

CLIENT-SIDE LANGUAGE



De JavaScript se dice que es un lenguaje del lado del cliente, es decir que los scripts son ejecutados por el navegador del usuario (cliente). Esto difiere de los llamados lenguajes de script del lado del servidor que son ejecutadas por el servidor web. Este es el caso de lenguajes como PHP.



¡HOLA MUNDO!

```
<!DOCTYPE html>
<html>
<head>
  <title>Hello World!</title>
</head>
<body>
  <script>
    alert('Hello world!');
  </script>
</body>
</html>
```

Declaración alert()

`alert ()` es una declaración simple, llamada función, que muestra un cuadro de diálogo que contiene un mensaje. este mensaje se coloca entre comillas, entre los paréntesis de la function alert ().

COMENTARIOS

```
<script>  
  // Este es mi primera función...  
  /* Comentario en más de una  
  línea*/  
  alert('Hello world!');  
</script>
```

VARIABLES

```
<script>
```

```
var miVariable;  
miVariable=2;
```

```
var miVariableNumero=2;  
var miVariableTexto="Hola Mundo!";
```

```
</script>
```

Tipado Dinámico

A diferencia de muchos lenguajes, JavaScript es un lenguaje de tipado dinámicamente. Esto significa, generalmente, que cualquier declaración de variables se hace con la palabra clave `var` independientemente de su contenido.

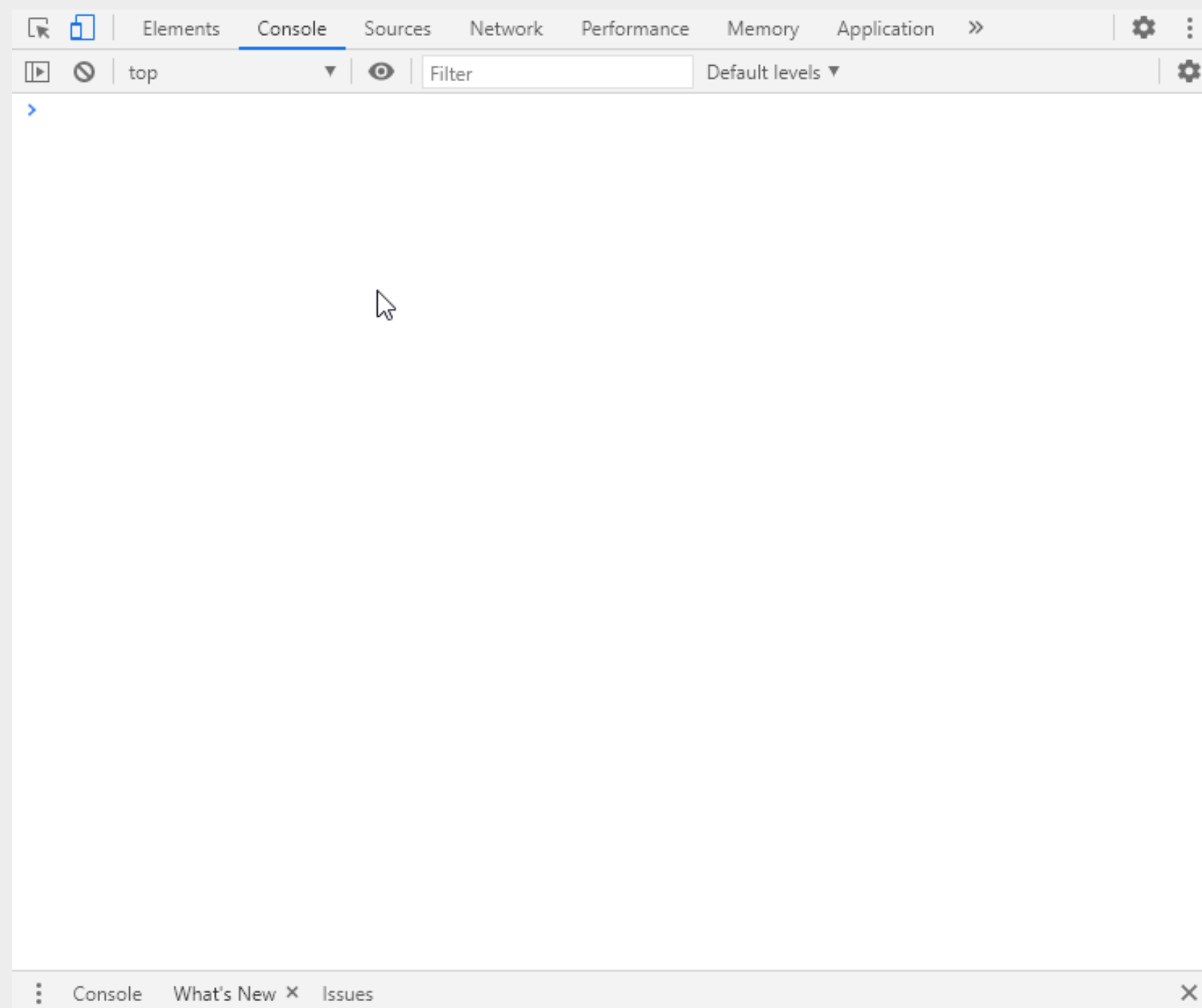
CONCATENACIÓN DE STRINGS

```
<script>  
  var textoUno="Hola";  
  var textoDos="Mundo";  
  var concatenacion = textoUno + " " +textoDos;  
</script>
```

`console.log()`

- `console.log()` es un método que escribe un mensaje en la consola.
- Dicho método se utiliza con propósitos de prueba.

(Tiene que estar abiertas las herramientas de Desarrollador del navegador (F12) para ver los mensajes del log en consola)



CONSOLE.LOG

```
<script>  
  var textoUno="Hola";  
  var textoDos="Mundo";  
  var concatenacion = textoUno + " " +textoDos;  
  console.log(concatenacion);  
</script>
```

DEBUGGER

- **debugger** es un método que nos permite colocar un punto de control para verificar el estado del script en ese momento
- Sirve para corregir e identificar errores.

(Tiene que estar abiertas las herramientas de Desarrollador del navegador (F12) para usar el Debugger)

DEBUGGER

```
<script>
```

```
var textoUno="Hola";
```

```
debugger;
```

```
var textoDos="Mundo";
```

```
debugger;
```

```
var concatenacion = textoUno + " " +textoDos;
```

```
debugger;
```

```
console.log(concatenacion);
```

```
</script>
```

CONDICIONALES

```
<script>
  if (juegoNombre == "Counter") {
    console.log("Nombre Incorrecto");
  } else {
    console.log("Nombre Correcto");
    if (cantidadJugadores > 0) {
      console.log("Hay " + cantidadJugadores + " Jugadores");
    } else {
      console.log("No hay Jugadores Conectados");
    }
  }
</script>
```

OPERADORES CONDICIONALES

- && - AND
- || - OR
- == - IGUAL
- != - DISTINTO

OPERADORES TERNARIOS

- **Un resultado de la condicion**

```
jugadorNombre == "Counter" ? console.log("True") : console.log("False");
```

- **Varios resultados de la condicion**

```
jugadorNombre == "Counter" ? (console.log("True"), alert("True!"))  
: (console.log("False"), alert("False!"));
```

SWITCH CASE

<script>

```
var juegoNombre = "Counter-Strike";  
switch (juegoNombre) {  
  case "Counter-Strike":  
    console.log(":D");  
    break;  
  case "Age of Empires":  
    console.log(":)");  
    break;  
  default:  
    console.log(":(");  
    break;  
}
```

</script>

FUNCIÓN

```
<script>  
    function imprimirPantalla(texto){  
        alert(texto);  
    }  
    imprimirPantalla("Hola Mundo!");  
</script>
```

FOR LOOP

```
<script>  
    function imprimirConsola(texto){  
        console.log(texto);  
    }  
    for (var i = 0; i < 9; i++) {  
        n += i;  
        imprimirConsola(n);  
    }  
</script>
```

WHILE LOOP

```
<script>  
    var n = 0;  
    var x = 0;  
    while (n < 3) {  
        n++;  
        x += n;  
    }  
    console.log(x);  
</script>
```


DO WHILE LOOP

```
<script>  
  var result = "";  
  var i = 0;  
  
  do {  
    i = i + 1;  
    result = result + i;  
  } while (i < 5);  
  
  console.log(result);  
</script>
```

ARRAYS

```
var frutas = ["Manzana", "Banana"];
```

- **Recorrer ARRAY**

```
frutas.forEach(function(elemento, indice, array) {  
    console.log(elemento, indice);  
});
```

- **Agregar Elemento** -> `frutas.push("Cereza");`
- **Quitar el Primer Elemento** -> `frutas.shift();`
- **Quitar el Último** -> `frutas.pop();`
- **Agregar al Inicio** -> `frutas.unshift("Pera");`
- **Conocer el índice ó posición** -> `console.log(frutas.indexOf("Pera"));`

JSON

JSON (acrónimo de **JavaScript Object Notation**, «**notación de objeto de JavaScript**») es un formato de texto sencillo para el intercambio de datos. Se trata de un subconjunto de la notación literal de objetos de JavaScript, aunque, debido a su amplia adopción como alternativa a XML, se considera (año 2019) un formato independiente del lenguaje.

JSON

```
{
  "orders": [
    {
      "orderno": "748745375",
      "date": "June 30, 2088 1:54:23 AM",
      "trackingno": "TN0039291",
      "custid": "11045",
      "customers": [
        {
          "custid": "11045",
          "fname": "Sue",
          "lname": "Hatfield",
          "address": "1409 Silver Street",
          "city": "Ashland",
          "state": "NE",
          "zip": "68003"
        }
      ]
    }
  ]
}
```

JSON

```
var fruta = {  
  "nombre": "Banana",  
  "cantidad": 5,  
  "precio": 20  
};  
  
console.log("Nombre de la Fruta", fruta.nombre);  
  
var frutas = [{  
  "nombre": "Banana",  
  "cantidad": 5,  
  "precio": 20  
},  
{  
  "nombre": "Manzana",  
  "cantidad": 3,  
  "precio": 50  
},  
];  
  
//Recorrer  
frutas.forEach(function(elemento, indice, array) {  
  console.log("Nombre", elemento.nombre);  
  console.log("Cantidad", elemento.cantidad);  
  console.log("Precio", elemento.precio);  
});
```

¿DUDAS?

