

Bootcamp Java Developer

Fase 2 - Java Web Developer
Módulo 19

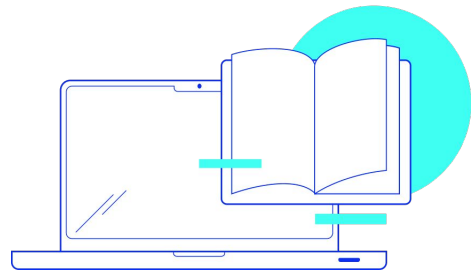
- **Arquitectura en pipeline (basada en filtros):** consiste en ir transformando un flujo de datos en un proceso comprendido por varias fases secuenciales. La entrada de cada una es la salida de la anterior.
- **Atributos HTML On:** atributos estándar HTML para el registro de eventos. Sirven para asociar la ejecución de una función a un evento que ocurra en la página (como el clic de un botón o la escritura de un campo de texto).
- **Babel:** herramienta que permite transformar el código JavaScript de última generación (o con funcionalidades extra) a JavaScript que cualquier navegador o versión de Node.js entienda.
- **Browser Object Model:** estructura de datos de tipo *Object* que representa al navegador en el que se está ejecutando el código JavaScript. Se accede a través del objeto *Window*.
- **Call Stack (pila de llamadas):** estructura de datos dinámica que almacena información sobre las funciones activas de un programa. En esta estructura, se almacena información sobre las funciones y subrutinas que se están ejecutando, que están pausadas o que deban ser ejecutadas en determinado momento.
- **Clases:** en JavaScript, son “funciones especiales” que proveen una sintaxis mucho más clara y simple para crear objetos y lidiar con la herencia.

- **Document Object Model:** es una **estructura de datos (M)**, de tipo **Object (O)**, que representa al **documento HTML** en el que se está ejecutando el código JavaScript (**D**). Se accede a través del objeto *Document*.
- **Función pipe():** se utiliza para vincular operadores entre sí. Toma como argumentos las funciones que desea combinar y devuelve una nueva función que ejecuta las funciones compuestas en secuencia.
- **Iterador:** objeto que permite recorrer una colección y, al finalizar, devolver un valor.
- **Location:** objeto de solo lectura que contiene información acerca de la ubicación HTTP actual de la página. Esto es, acerca de la ruta actual.
- **npm:** es el registro de software más grande del mundo con, aproximadamente, 3 mil millones de descargas por semana. Contiene más de 600000 paquetes (bloques de construcción de código).
- **Objeto Proxy:** permite crear un intermediario para otro objeto, que puede interceptar y redefinir las operaciones fundamentales del objeto interceptado.

- **Operadores (RxJs):** funciones que se basan en la base de los observables para permitir una manipulación sofisticada de las colecciones. Por ejemplo, RxJS define operadores como **map()**, **filter()**, **concat()** y **flatMap()**.
- **Promesas:** objeto que representa una tarea asincrónica y simplifica su gestión al tipificar sus estados posibles en pendiente, resuelta o rechazada.
- **Prototipo:** en JavaScript cada objeto tiene una propiedad privada (referida como su **[[Prototype]]**) que mantiene un enlace a otro objeto llamado su prototipo. Ese objeto tiene su propio prototipo, y así sucesivamente hasta que se alcanza uno cuyo prototipo es null. Por definición, **null** no tiene prototipo, y actúa como el enlace final de esta cadena de prototipos.

- **SPA (Single Page Application):** es una aplicación web que cabe en una sola página y tiene el propósito de dar una experiencia más fluida a los usuarios, como si fuera una aplicación de escritorio. Todos los códigos de HTML, JavaScript, y CSS se cargan una sola vez.
- **Storage API:** la interfaz de la API de almacenamiento web proporciona acceso al almacenamiento de la sesión o al local para un dominio particular. Esto le permite, por ejemplo, agregar, modificar o eliminar elementos de datos almacenados.
- **Tipo de dato abstracto:** construcciones creadas por el desarrollador en base a los tipos de dato primitivos y otros tipos abstractos. Sirven para representar de forma más fiel la realidad que el programa se propone emular o automatizar.
- **Tipo de dato primitivo:** son proporcionados por el lenguaje y permiten construir tipos de datos abstractos y estructuras de datos.
- **Transpiladores:** programas capaces de traducir el código de un lenguaje hacia otro, o de una versión hacia otra.

- **TypeScript:** lenguaje de programación construido sobre JavaScript con características típicas de lenguajes fuertemente tipados. Añade a JavaScript una fase extra en el proceso de construcción y una característica propia de los lenguajes compilados: que el código que se escriba no sea el que finalmente se ejecute.
- **Webpack:** empaquetador de módulos (*bundler*), que además tiene muchas otras funciones.



**Ahora sí,
¡Comencemos!**

