

Modul 1: Pengenalan JavaScript & Integrasi

Sejarah, Evolusi, Arsitektur Engine, dan Implementasi ke HTML

Kusuma Web Academy

28 Juni 2026

Sejarah, Evolusi, & Peran Vital JavaScript

- **Kelahiran Awal:** Diciptakan oleh Brendan Eich di Netscape dalam waktu 10 hari pada tahun 1995. Awalnya bernama Mocha, berubah menjadi LiveScript, hingga akhirnya bermerek dagang **JavaScript** sebagai taktik pemasaran bersama Java.
- **Standarisasi ECMAScript:** Untuk menghindari perang browser, badan standar Ecma International merumuskan spesifikasi standar bahasa pemrograman yang dinamai **ECMAScript**. Lompatan terbesar terjadi pada rilis **ES6 (ES2015)** yang membawa fitur modern.
- **Peran Vital Modern:** JavaScript bertransformasi dari sekrip animasi sederhana menjadi bahasa pemrograman paling populer di dunia. JS kini menguasai Front-end (React, Vue) dan juga Back-end server (Node.js).

Memahami Arsitektur Engine JavaScript

Bagaimana browser komputer menjalankan berkas teks JavaScript Anda?

- **JavaScript Engine:** Kompiler bawaan browser (seperti Google V8 pada Chrome/Node.js, SpiderMonkey pada Firefox) yang bertugas mem-parsing teks kode mentah menjadi kode biner mesin.
- **JIT (Just-In-Time) Compilation:** JS tidak murni diinterpretasikan baris-per-baris, melainkan dikompilasi secara dinamis saat runtime berjalan untuk efisiensi kecepatan eksekusi yang optimal.
- **Single-Threaded & Non-Blocking:** JS hanya memiliki satu jalur eksekusi antrean utama (*call stack*), namun mampu menangani operasi berat secara asinkronus menggunakan bantuan sistem *Event Loop*.

Metode Integrasi Skrip JavaScript ke HTML

Terdapat dua metode utama yang diakui standar industri dalam peletakan dokumen JavaScript:

1. Internal JavaScript (Menggunakan Tag Script)

```
<script>  
    console.log("Halo dari Kusuma Web!");  
</script>  
\
```

2. External JavaScript (Standar Industri Terpisah & Teratur)

```
<!-- Disambungkan di bagian akhir sebelum tag penutup body -->  
<script src="assets/js/main.js"></script>  
\
```

Praktik Langkah Pertama: Konsol Browser

Ikuti langkah-langkah praktis berikut untuk memulai eksekusi mandiri:

- 1 Buatlah berkas baru bernama `index.html` dan masukkan kerangka dasar dokumen HTML5.
- 2 Buatlah berkas baru bernama `app.js` di dalam satu folder yang sama.
- 3 Tuliskan kode berikut pada `app.js`:

```
console.log("Sistem Kusuma Web Berhasil Terintegrasi!");  
\
```
- 4 Hubungkan berkas `app.js` di dalam `index.html` sebelum tag penutup `</body>`.
- 5 Jalankan di browser, klik kanan halaman, pilih **Inspect**, lalu pilih menu **Console** untuk melihat hasil outputnya.