

Modul 3: Klasifikasi Tipe Data & Struktur Memori

Tipe Data Primitif, Tipe Data Referensi, Alokasi Memori, dan Type Coercion

Kusuma Web Academy

28 Juni 2026

Klasifikasi Tipe Data Primitif

Tipe data primitif menyimpan nilai tunggal secara langsung di dalam memori *Stack* komputer:

- **String**: Kumpulan data teks berurutan yang wajib diapit oleh kutip satu (' '), kutip dua (" "), atau backticks (` `) untuk fitur interpolasi string.
- **Number**: Mewakili angka desimal maupun bulat (tidak dibedakan antara float atau integer di JS).
- **Boolean**: Nilai kebenaran mutlak yang hanya memiliki dua kemungkinan: `true` atau `false`.
- **Null**: Representasi sengaja dari ketidakadaan nilai (kosong) yang sengaja disetel oleh developer.
- **Undefined**: Menandakan suatu variabel yang sudah dideklarasikan namun belum pernah diisi nilai apapun ke dalamnya.

Tipe Data Referensi & Struktur Memori

Tipe data referensi tidak menyimpan nilainya di dalam memori *Stack*, melainkan di memori *Heap*:

- **Konsep Pointer:** Variabel pada Stack hanya menyimpan alamat penunjuk (*reference pointer*) yang mengarah langsung ke objek aslinya yang tersimpan di dalam memori Heap komputer.
- **Array:** Kumpulan item terurut yang diidentifikasi menggunakan nomor indeks (mulai dari indeks 0).
- **Object:** Koleksi pasangan kunci (*key*) dan nilai (*value*) yang mampu menampung struktur data kompleks secara representatif.

Implikasi Penting Mutabilitas:

Tipe data referensi bersifat **mutable** (dapat dimodifikasi). Jika variabel referensi disalin ke variabel baru, perubahan pada salah satu variabel akan mempengaruhi variabel lainnya karena keduanya merujuk ke alamat Heap yang sama.

Type Conversion vs Type Coercion

Perubahan tipe data dapat terjadi secara sengaja maupun otomatis di balik layar:

- **Type Conversion (Explicit):** Dilakukan sengaja oleh developer menggunakan fungsi bawaan global JS.

```
let teksAngka = "125";  
let angkaAsli = Number(teksAngka); // String dikonversi ke Number
```

- **Type Coercion (Implicit):** Dilakukan otomatis di balik layar oleh engine JavaScript saat melakukan operasi antar tipe berbeda.

```
let hasil = "Kusuma Web " + 10; // Output: "Kusuma Web 10" (String)  
let hasilPerkalian = "5" * 2; // Output: 10 (Number, dipaksa otomatis!)
```

Latihan Praktik Identifikasi Tipe Data

Tuliskan baris pengujian tipe berikut pada file kerja JavaScript Anda untuk mempelajari perilakunya:

```
let nama = "Kusuma";  
let kuota = 25;  
let lunas = true;  
let dataSiswa = ["Deri", "Budi"];  
  
console.log(typeof nama);      // Output: "string"  
console.log(typeof kuota);    // Output: "number"  
console.log(typeof lunas);    // Output: "boolean"  
console.log(typeof dataSiswa); // Output: "object" (Array diidentifikasi objek)
```