






```
1 console.clear();
2
3 // 🧑 Mastering JavaScript Data Types – A Frontend Developer's Essentials 📖
4
5 // ✅ Data Types are classified into two main categories: Primitive and Non-Primitive
6
7 /*
8
9   ◆ Primitive Data Types:
10
11     ✓ Immutable (cannot be changed)
12
13     ✓ Stored in Stack memory
14
15     ✓ Fast access
16
17   ◆ Non-Primitive Data Types:
18
19     ✓ Mutable (can be changed)
20
21     ✓ Stored in Heap memory
22
23     ✓ Slower access
24
25   This understanding is crucial when dealing with performance, memory leaks and reference vs value behavior in frontend
26   development.
27 */
28
29 // ✅ Primitive Data Types in JavaScript
30
31 // 1 Number: For all numeric values (integers, decimals and exponential notation)
32
33 let score = 99;
34
35 let pi = 3.14159;
36
37 console.log("score:", score);
```

```
38
39 console.log("Type:", typeof score);
40
41 console.log("pi:", pi);
42
43 console.log("Type:", typeof pi);
44
45 // 2 String: Sequence of characters wrapped in quotes. We use double quotes when it's a sentence or a paragraph and single
  quotes when it's a single word.
46
47 let fullName = "Heet Gudhka";
48
49 let nickName = 'Heet';
50
51 console.log("fullName:", fullName);
52
53 console.log("Type:", typeof fullName);
54
55 console.log("nickName:", nickName);
56
57 console.log("Type:", typeof nickName);
58
59 // 3 Boolean: true or false – perfect for logic & UI conditions
60
61 let isLoggedIn = true;
62
63 let hasPermission = false;
64
65 console.log("isLoggedIn:", isLoggedIn);
66
67 console.log("Type:", typeof isLoggedIn);
68
69 console.log("hasPermission:", hasPermission);
70
71 console.log("Type:", typeof hasPermission);
72
73 // 4 Undefined: Declared but not assigned any value
74
```

```
75 let currentTask;
76
77 console.log("currentTask:", currentTask);
78
79 console.log("Type:", typeof currentTask);
80
81 // 5 Null: Explicit absence of value – used intentionally
82
83 let selectedTheme = null;
84
85 console.log("selectedTheme:", selectedTheme);
86
87 console.log("Type:", typeof selectedTheme); // typeof null → 'object' (legacy bug in JS)
88
89 // 6 BigInt: For values larger than 2^53 - 1
90
91 const worldPopulation = 1234567890123456789012345678901234567890n;
92
93 console.log("worldPopulation:", worldPopulation);
94
95 console.log("Type:", typeof worldPopulation);
96
97 // 7 Symbol: Unique and immutable – used for object keys and constants
98
99 const appID = Symbol("app_id");
100
101 console.log("appID:", appID);
102
103 console.log("Type:", typeof appID);
104
105 // 🔍 Bonus Tip: Max Safe Integer in JavaScript
106
107 console.log("Max Safe Integer:", Number.MAX_SAFE_INTEGER);
108
109 /*
110
111 Non Primitive Data Types in JavaScript
112
```

```
113 It includes:
114
115  Object
116
117  Array
118
119  Function
120
121  Date
122
123  Set
124
125 Stay tuned as we deep dive into how non-primitive data types work in JavaScript.
126
127 */
```