JavaScript Day 2 Notes – Data Types & Strict Mode

1. "use strict" – Enforcing Modern JavaScript

js

CopyEdit

"use strict";

- **Forces JavaScript to follow modern rules.**
- **Prevents mistakes** (like using undeclared variables).
- **Example:**

js

CopyEdit

"use strict";

x = 10; // ❌ This will give an error because x is not declared.

---

2. Writing Clean Code (Readability Matters!)

Bad Practice:

js

CopyEdit

console.log(3

+

3);

✅ **Good Practice:**

js

CopyEdit

console.log(3 + 3); // Easy to read & understand

- Always write code in a **clear & readable** way!

---

3. JavaScript Data Types (Primitive Data Types)

(A) Number

- Stores numbers (integers, decimals, etc.).
- **Example:**

js

CopyEdit

let age = 21; // Number

(B) BigInt

- Used for **very large numbers** (beyond 2^53).
- **Example:**

js

CopyEdit

let bigNumber = 12345678901234567890n; // Add 'n' at the end

(C) String

- Stores text (inside " " or ' ' or ` `).
- **Example:**

js

CopyEdit

let name = "Pranay"; // String

(D) Boolean

- Stores **true** or **false** (Used for conditions).
- **Example:**

js

CopyEdit

let isLoggedIn = false; // Boolean

(E) Null

- A **standalone empty value** (not 0, not undefined).
- **Example:**

js

CopyEdit

let temperature = null; // Value is empty (but valid)

(F) Undefined

- A variable **declared but not assigned a value**.
- **Example:**

js

CopyEdit

let user; // Undefined

(G) Symbol

- Used to **create unique values** (Mostly used in **React**).
- **Example:**

js

CopyEdit

let symbol1 = Symbol("id");

let symbol2 = Symbol("id"); // Both are unique!

---

4. Object Data Type

- **Objects** are used to store **multiple values in one variable** (Covered in detail later).

---

5. Checking Data Types (typeof Operator)

- typeof is used to check the type of a variable.
- **Examples:**

js

CopyEdit

console.log(typeof "Pranay"); // Output: string

console.log(typeof 21); // Output: number

console.log(typeof true); // Output: boolean

---

6. Special Cases (null vs undefined)

(A) typeof null

js

CopyEdit

console.log(typeof null); // Output: object

- **Why?** It's a known JavaScript mistake. null is **not really an object**, but JavaScript treats it like one.

(B) typeof undefined

js

CopyEdit

console.log(typeof undefined); // Output: undefined

- undefined means the **variable exists but has no value**.

Summary:

✅ **Use "use strict";** to avoid mistakes.
✅ **Write clean, readable code** for better understanding.
✅ **JavaScript has 7 primitive data types:** number, bigint, string, boolean, null, undefined, symbol.
✅ **Use typeof** to check a variable's data type.
✅ null is an **empty value** (but wrongly shown as an object).
✅ undefined means **a variable has no value assigned**.