⬛ LECTURE 9 — DATE IN JAVASCRIPT

📘 **JavaScript Core — Date & Time Handling**
🧠 *First Principles • Real-Life Examples • Accurate Time Logic*
📅 **Today's Reference Date: 22 January 2026**

⬛ FIRST PRINCIPLE — "DATE KYU ZARURI HAI?"

🔷 Real-Life Soch

Socho:

- Ticket booking time
- Exam submission deadline
- Payment timestamp
- Login / logout time

👉 In sab me **exact date + exact time** matter karta hai

🟨 **Isi problem ko solve karta hai → JavaScript Date Object**

⬛ WHAT IS DATE IN JAVASCRIPT?

🔷 Definition

Date ek **built-in JavaScript object** hai jo:

- Date create karta hai
- Time calculate karta hai
- Date & time ko format aur modify karta hai

```
const d = new Date();

console.log(d);
```

🟨 **Example Output (UTC format):**

```
2026-01-22T00:00:00.000Z
```

⚠️ **Important**

```
typeof d; // "object"
```

⬛ DATE KAHAN SE AATI HAI? (CORE CONCEPT)

🔷 System Clock

JavaScript **system ke clock** se date leta hai

🔷 Internally JavaScript kya karta hai?

JavaScript **milliseconds count** karta hai starting from:

> 🟨 **1 January 1970, 00:00:00 UTC**
> (Isko kehte hain **Unix Epoch**)

🔷 Example — Epoch se Date

```
const d1 = new Date(1000);

console.log(d1);
```

🟨 **Output**

```
1970-01-01T00:00:01.000Z
```

💡 **Tip**

> `new Date(2000)` → Epoch ke **2 seconds baad** ka time

⬛ DATE CREATE KARNE KE TAREEKE

① Current Date & Time

```
const d = new Date();

console.log(d.toString());
```

🟨 **India Time (IST)**

```
Thu Jan 22 2026 05:30:00 GMT+0530 (India Standard Time)
```

---

② String Format Date

```
const d = new Date("2026-01-22");

console.log(d);
```

🟨 **Output**

```
2026-01-22T00:00:00.000Z
```

---

③ Date + Time Together

```
const d = new Date("2026-01-22T10:10:12");
```

---

④ Year, Month, Date Format

```
const d = new Date(2026, 0, 22);

console.log(d.toDateString());
```

🟨 **Output**

```
Thu Jan 22 2026
```

⚠️ **VERY IMPORTANT RULES**

- Month **0-based** hota hai
  - 0 → January
  - 11 → December

📌 Full Signature

```
new Date(year, month, date, hours, minutes, seconds, milliseconds)
```

🟨 Special Case

Agar sirf **ek number** diya → JS use **milliseconds** maan leta hai

---

⬛ DATE FORMATTING (REAL-WORLD USE)

```
const d = new Date();


console.log(d.toDateString());

console.log(d.toString());

console.log(d.toISOString());
```

🟨 Outputs

```
Thu Jan 22 2026

Thu Jan 22 2026 05:30:00 GMT+0530

2026-01-22T00:00:00.000Z
```

💡 Pro Tip

`.toISOString()` → **Backend & API development ke liye best**
(Universal + timezone-safe)

⬛ GET METHODS — DATE SE DATA NIKALNA

```
const d = new Date();


d.getFullYear();       // 2026

d.getMonth();          // 0  (January)

d.getDate();           // 22
```

```
d.getDay();            // 4  (Thursday)

d.getHours();          // 5

d.getMinutes();        // 30

d.getSeconds();        // 00

d.getMilliseconds();   // depends on system
```

⚠️ **REMEMBER**

- `getMonth()` → 0 = Jan
- `getDay()` → 0 = Sunday, 6 = Saturday

⬛ MILLISECONDS — SABSE IMPORTANT CONCEPT

🔷 First Principle

JavaScript **date compare nahi karta**
👉 **milliseconds compare karta hai**

```
d.getTime(); // milliseconds since Jan 1, 1970
```

🔷 Real-Life Example (Ticket Booking)

- Person A → `23815679 ms`
- Person B → `23815678 ms`

👉 **Person B ne 1 millisecond pehle booking ki**

🟨 **Use Cases**

- Accurate comparison
- Time difference calculation
- Logs & history
- Sorting by time

Current Timestamp

```
const now = Date.now();

console.log(now);
```

🟨 **Meaning**

Current date & time in **milliseconds**

⬛ DATE MODIFY KARNA (SET METHODS)

```
const d = new Date();


d.setDate(22);

d.setFullYear(2026);

d.setMonth(0); // January


console.log(d.toString());
```

Auto-Adjustment Feature (Smart JS)

```
let d = new Date(2026, 0, 35);

console.log(d.toDateString());
```

🟨 **Output**

```
Wed Feb 04 2026
```

🧠 **Reason**

JS automatically **overflow adjust** kar leta hai

⬛ SHORT SUMMARY — EK NAZAR ME ⬛

✅ `new Date()` → current date
✅ `new Date(ms)` → epoch milliseconds
✅ `new Date("YYYY-MM-DD")` → string date
✅ Month → 0-based
✅ `getTime()` / `Date.now()` → milliseconds
✅ `get` methods → read date
✅ `set` methods → modify date
✅ `toISOString()` → API-safe format

⬛ FINAL THOUGHT ⬛

🧠 **JavaScript Date = milliseconds ka game**
Isko samajh liya →
👉 **Booking systems, backend logic, APIs sab smooth**
