

# Using Moment.js in JavaScript

I'll show you how to use the `moment` and `moment-timezone` libraries in JavaScript. These are popular libraries for handling dates and times.

1. First, install the packages if you haven't already:

```
npm install moment moment-timezone
```

2. Basic usage examples:

```
// Import the libraries
const moment = require('moment');
const momentTimezone = require('moment-timezone');

// Basic Moment.js usage
// Current date and time
const now = moment();
console.log(now.format('YYYY-MM-DD HH:mm:ss')) // e.g., "2025-03-03 14:30:45"

// Parse a specific date
const date = moment('2025-12-25');
console.log(date.format('MMMM Do YYYY')) // "December 25th 2025"

// Manipulate dates
const tomorrow = moment().add(1, 'days');
console.log(tomorrow.format('ddd, MMM D')) // e.g., "Tue, Mar 4"

// Using Moment-Timezone
// Get time in a specific timezone
const nyTime = momentTimezone.tz('America/New_York');
console.log(nyTime.format('YYYY-MM-DD HH:mm:ss z')) // Includes timezone abbreviation

// Convert time between timezones
const localTime = moment();
const tokyoTime = localTime.clone().tz('Asia/Tokyo');
console.log(`Local: ${localTime.format('HH:mm')}`);
console.log(`Tokyo: ${tokyoTime.format('HH:mm')}`);

// List all available timezones
const timezones = momentTimezone.tz.names();
console.log(timezones); // Array of timezone names
```

Common formatting options: - YYYY: 4-digit year (2025) - MM: 2-digit month (03) - DD: 2-digit day (03) - HH: 24-hour (14) - hh: 12-hour (02) - mm: minutes (30) - ss: seconds (45) - z: timezone abbreviation

Useful methods:

```
// Date manipulation
moment().add(7, 'days');      // Add time
moment().subtract(2, 'hours'); // Subtract time
moment().startOf('month');   // Beginning of period
moment().endOf('month');    // End of period

// Date comparison
moment().isBefore(date);
moment().isAfter(date);
moment().isSame(date);

// Timezone operations
momentTimezone.tz('2025-03-03', 'America/Los_Angeles');
momentTimezone.tz.guess(); // Guess user's timezone
```

Notes: 1. These versions (^2.14.1 and ^0.5.7) are quite old (from around 2016). Consider using the latest versions unless you have specific compatibility requirements. 2. Moment.js is now considered a legacy project in maintenance mode. For new projects, you might want to consider alternatives like date-fns or luxon.