# Node.js - Day 2: Runtime, CLI Apps & Debugging

## ■ Learning Objectives

- Understand what the Node.js runtime does.
- Create and run CLI (Command Line Interface) applications.
- Use console input/output for interactivity.
- Learn basic debugging techniques.
- Build a small interactive guessing game.

## ■ Session 1 (45 min) — Node.js Runtime + CLI Apps

**1. What is the Node.js Runtime?**
Node.js is the environment that allows JavaScript to run outside a browser. It reads and executes JS code, provides built-in modules, and uses the V8 engine for fast performance. Think of the runtime as a kitchen where the chef (V8 engine) executes the recipe (your JavaScript code).

**2. Understanding CLI Apps**
CLI means Command Line Interface — text-based interaction between user and app.
Example:
// cli-example.js console.log("Welcome to my CLI app!"); const name = process.argv[2]; console.log(`Hello, ${name || "stranger"}!`); Run it:
node cli-example.js Dielli Output:
Hello, Dielli!

**3. Using Readline for Interactive Input**
Node's **readline** module allows interactive input.
const readline = require('readline'); const rl = readline.createInterface({ input: process.stdin, output: process.stdout }); rl.question("What's your name? ", (name) => { console.log(`Nice to meet you, ${name}!`); rl.close(); }); Run:
node askname.js

## ■ Session 2 (45 min) — Debugging + Mini Project

**1. Debugging in Node.js**
Common tools:
- console.log() — trace variables
- node inspect filename.js — debug step by step
- debugger keyword — pause code in VS Code
Example:
let number = 5; debugger; console.log(number); In terminal:
node inspect app.js

## ■ Mini Project — Number Guessing Game

Goal: Create a CLI app where the user guesses a random number.
const readline = require('readline'); const rl = readline.createInterface({ input: process.stdin, output: process.stdout }); const randomNumber = Math.floor(Math.random() * 10) + 1; rl.question("Guess a number between 1 and 10: ", (answer) => { if (parseInt(answer) === randomNumber) { console.log("■ Correct! You guessed it!"); } else { console.log(`■ Wrong! The number was ${randomNumber}.`); } rl.close(); }); Challenge: Allow up to 3 attempts before losing.

## ■ Homework

Write a Node.js script that prints your daily schedule.
console.log("■ My Daily Schedule:"); console.log("8:00 AM - Wake up"); console.log("9:00 AM - Breakfast"); console.log("10:00 AM - Coding practice"); console.log("1:00 PM - Lunch"); console.log("3:00 PM - Study more Node.js"); console.log("6:00 PM - Exercise"); console.log("9:00 PM - Chill & Sleep"); Encourage creativity — add emojis or use new Date() to show current time.

## ■ Recap Discussion

- What is Node’s runtime responsible for?
- How does CLI input work in Node.js?
- What are 2 ways to debug Node.js code?
- What did you learn from the guessing game?