

OpenJS Node.js Application Developer (JSNAD)

This intermediate OpenJS Node.js Application Developer (JSNAD) training prepares software developers to build high-performance Node.js applications, integrate databases and test and debug apps for production readiness. This JSNAD exam prep outlines the topics you'll find on the OpenJS Node.js Application Developer exam. It will also give you lots of opportunities to implement buffer and streams, control flow, error handling, the module system, unit testing — as well as help you earn your JSNAD certification.

[CBT Nuggets course material](#) →

WEEK 1

Introduction to Node.js Buffers

157 min.

Creating Buffers	12
Writing Data to Buffers	9
Copying Buffers	10
Challenge & Solution: Printing an ASCII Chart	10

Manipulating Node.js Buffers

Slicing Buffers with "Subarray"	14
Combining Buffers with "Concatenate"	8
"Array-Like" Buffer Methods	9
Challenge & Solution: Buffer Operations Review	12

Node.js Buffer Integer Data Formats

Basics of Writing Numbers to Buffers	10
Writing Integers to Buffers	17
Reading Integers from Buffers	9
Challenge & Solution: Storing Multiple Numbers in a Buffer	10

Node.js Buffer Float Formats

How Floats Work	13
How Floats are Stored in Buffers	14

WEEK 2

154 min.

64 Bit "Doubles" and Little-Endian Floats	6
Challenge: Writing Multiple Floats	9

Node.js Buffer String Formats

What Are String Formats?	6
Working with String Formats in Node.js	16
Writing and Reading String Formats	6
Storing Arrays and Objects in Node.js Buffers	7
Challenge & Solution: Storing Objects	17

Basic Buffer Applications

Reading Files Into Buffers	9
Writing to Files from Buffers	8
Using Binary Operators with Buffers	12
Using Buffers to Create a Simple XOR Cipher	6
Challenge & Solution: Deciphering XOR Files	6

Manage Asynchronous Operations with Callbacks

Basics of Callbacks	7
Callbacks with Args	13
Callbacks Are Not Sequential!	13
Challenge & Solution: Back-to-Back Async Operations	13

WEEK 3

Intermediate Callback Operations

156 min.

Avoiding the Pyramid of Doom	1
Passing On Data Without Nesting	6
Creating Helper Functions with Callbacks	10
Executing Async Operations in Parallel	14
Challenge & Solution: Deleting In Parallel	6

Manage Asynchronous Operations with Promises

Basics of Promises	8
Basic Promise Syntax	6
Chaining Multiple Async Operations	9
Promises Are Sort-of Sequential	8
Handling Errors with Promises	8
Challenge: Rewriting Callbacks	8

Intermediate Promise Concepts

Creating Our Own Promises	12
Resolving and Rejecting Promises with Values	8
Creating a setTimeout Promise	9
Challenge & Solution: Promisifying Callback Functions	15

Basic Promise Concurrency Methods

The Promise.all Method	15
The Promise.allSettled Method	13

WEEK 4

156 min.

Mapping Asynchronously	10
Challenge & Solution: Creating a "Map Async" Function	5

Other Promise Concurrency Methods

The Promise.race Method	12
The Promise.any Method	4
The Promise.resolve and Promise.reject Methods	13
Challenge & Solution: Recreating the Concurrency Methods	14

Work with Async and Await in JavaScript

Introduction to Async/await Syntax	8
Using Async and Await	1
Converting Between Promises and Async/Await	9
Error Handling with Async/Await	2
A Closer Look At Async/Await Execution Flow	8
Challenge & Solution: Using Async/Await with Concurrency	8

Learn the Basics of Generator Functions

The Basic Idea of Generator Functions	11
Basic Generator Syntax and Usage	10
Yielding Dynamic Values	8
Creating Infinite Generators	10
Challenge & Solution: Generating Primes	6

Asynchronous Handling With the Async Library

Basics of the Async Library	1
How Async's Callbacks Work	8
Automatic Retries with the Async Library	8

WEEK 5

155 min.

Running Functions Based on Their Requirements	19
Challenge: Check Out Other Async Library Functions	1

Asynchronous Handling With the Bluebird Library

Promisifying With Bluebird	11
Bluebird Promise Syntax & Cancellation	9
Specific Error Handling with Bluebird	8

Setting Timeouts in Bluebird	6
Challenge & Solution: Canceling Promises in Practice	12

Intermediate Generator Concepts

Delegating Generators	15
Delegating to Iterable Objects	7
Passing Values Into Generators	1
Challenge & Solution: Collecting User Data	2

Managing Asynchronous Code with RxJS

Basics of the RxJS Library	8
Creating and Subscribing to Observables	16
Handling Errors and Completion	8
RxJS Operators	12
Challenge & Solution: User Input Observable	7

Advanced Generator Concepts

Can Generator Functions Have Return Values?	13
---	----

WEEK 6

154 min.

Mapping Generators	1
Filtering and Slicing Generators	10
Challenge & Solution: Generators with Promises	5

Manage CPU-Intensive Tasks with Child Processes

What are Child Processes?	10
Running a Simple Child Process	9
Passing Data Into Child Processes	1

Getting Data From Child Processes	8
Challenge & Solution: Running Multiple Child Processes	3

Intermediate Child Process Concepts

The Different Process Events	1
Combining Processes with Promises	14
Challenge & Solution: Combining Processes with RxJS	5

Run Shell Commands with Child Processes

Running Shell Commands with Exec	9
Other Shell Command Possibilities	13
Running Node Scripts with Exec	7
Using ExecFile	8
Challenge & Solution: Promisifying Shell Operations	12

Advanced Child Process Topics

The Spawn Function	9
Running Node Scripts with Spawn	10
Child Process Configuration Options	10
Other Inter-Process Communication Options with Spawn	9

WEEK 7

157 min.

Challenge & Solution: Convert Exec to Spawn	1
---	---

Using Node In The Terminal

Running JavaScript Files and Code	12
-----------------------------------	----

Setting Environment Variables	9
Command Line Arguments	1
Installing NPM Packages	7
Challenge & Solution: Greeting with Env Variables and Args	3

Basics of File I/O in Node.js

Reading Files	14
Writing and Appending Files	1
Deleting, Moving, and Copying Files	7
Challenge & Solution: Storing Program Data	11

Intermediate File Operations

The path.join Function	9
Other "path" Functions	16
Working with Directories	14
Challenge & Solution: Reading Multiple Files In a Directory	7

Advanced File I/O Operations

Getting File Stats	13
Working with File Handles	14
Watching Files	11
Challenge & Solution: Observing Files	7

WEEK 8

JavaScript Basics Review: Data Types and Functions 151 min.

JavaScript's Data Types: Strings	11
JavaScript's Data Types: Numbers and BigInts	6
JavaScript's Data Types: Booleans, Objects, and More	15

The Different Function Syntaxes	9
Challenge & Solution: Falsy and Truthy	5

JavaScript Basics Review: Inheritance

Prototypal Inheritance Examples	1
Prototypal vs. Classical Inheritance	14
Challenge & Solution: Shape Prototypes	10

Basics of Unit Testing Node.js Applications

What are Assertions?	5
The Node.js "assert" Function	12
More Assertion Examples	10
Assertion Utilities	5
Challenge & Solution: Writing Your Own Assertions	3

The Assert Module In-Depth

Different Ways to Check for Equality	14
Testing for Inequality	9
Testing for Failure	7
Using the Node.js Test Runner	11
Challenge & Solution: Writing A Test Suite	4

WEEK 9

Testing Asynchronous Code in Node.js

162 min.

Testing Callback Code	10
Testing Callback Errors	11
Testing Promises	10
Challenge & Solution: Writing Async Tests	3

Advanced Testing in Node.js

Skipping Tests	9
Running Groups of Tests	6
Filtering Tests By Name	5
Mocking Functions	16
Challenge & Solution: More Complex Mocking	9

Introduction to Node.js Web Sockets

Websocket Basics	7
Working with Websockets Using "ws"	9
Sending Messages Through Socket Connections	9
Handling Closed Connections	8
Challenge & Solution: Prefixed Messages	12

Node.js Websocket Libraries

The "Socket.io" Library	11
Other Socket.io Features	14
The SockJS Library	13

WEEK 10

160 min.

Challenge & Solution: SockJS Broadcasting	1
---	---

Node.js Testing Libraries: Mocha

Setting Up Mocha	15
Mocha's Hooks	13
The .Mocharc File	10
Challenge & Solution: Mocharc Changes	3

Node.js Testing Libraries: Chai

Making Basic Chai Assertions	12
An Easier Way to Import Chai	7
Making "Expect" Assertions	13
Making "Should" Assertions	5
Challenge & Solution: Exploring Chai Assertions	4

The Node.js Event System

The Observer Pattern	6
EventEmitter Basics	14
File Loading with EventEmitter	9
Progress Reporting with EventEmitter	8
Challenge & Solution: Building a Timer	3

Node.js Debugging Basics

Basic Debugging Tools	8
Debugging with the --inspect-brk Flag	9
Setting Breakpoints	10
Using the "Step" Buttons	10

WEEK 11

102 min.

Challenge & Solution: Debugging with node inspect	5
---	---

The Node.js Module System

Module Basics	7
The CommonJS Module System	14
The ES Module System	14
Challenge & Solution: When Are Modules Loaded?	2

Node.js Process Control

Node Process Basics	8
Listening for Process Events	11
Other Process Events	9
The OS Module	10
Challenge & Solution: Uptime Notifier	3

Learn the Basics of Node.js Streams

Basics of Streaming	14
Working with Duplex and Transform Streams	1
Streaming Files	1
Challenge & Solution: Reversing Words	3