



Javascript Roadmap



JAVASCRIPT MASTERY SERIES (100 Episodes)

FOUNDATION LEVEL (Episodes 1-25)

Episode 1-5: JavaScript Fundamentals

- What is JavaScript? History and evolution
- JavaScript engines (V8, SpiderMonkey)
- Setting up development environment
- Browser vs Node.js JavaScript
- JavaScript in HTML (script tags, external files)

Episode 6-10: Variables and Data Types

- Variables (var, let, const) - differences and scope
- Primitive data types (string, number, boolean, null, undefined, symbol, bigint)
- Reference types vs primitive types
- Type coercion and type conversion
- typeof operator and type checking

Episode 11-15: Operators and Expressions

- Arithmetic operators (+, -, *, /, %, **)
- Comparison operators (==, ===, !=, !==, <, >, <=, >=)
- Logical operators (&&, ||, !)
- Assignment operators (=, +=, -=, etc.)
- Bitwise operators
- Ternary operator
- Operator precedence and associativity

Episode 16-20: Control Flow

- Conditional statements (if, else if, else)
- Switch statements and fall-through behavior
- Loops (for, while, do-while)
- for...in vs for...of loops
- Break and continue statements

Episode 21-25: Functions Basics

- Function declarations vs expressions
- Arrow functions and their syntax
- Function parameters and arguments
- Default parameters
- Rest parameters (...args)
- Return statements
- Function scope and local variables

INTERMEDIATE LEVEL (Episodes 26-60)

Episode 26-35: Advanced Functions

- Higher-order functions

- Callbacks and callback hell
- Closures and lexical scoping
- IIFE (Immediately Invoked Function Expressions)
- Function currying and partial application
- Call, apply, and bind methods
- Pure functions and side effects
- Recursion and tail call optimization
- Generator functions and yield
- Function composition

Episode 36-45: Objects and Arrays

- Object creation methods (literal, constructor, Object.create)
- Object properties and methods
- Property descriptors and getters/setters
- Object destructuring assignment
- Array creation and initialization
- Array methods (push, pop, shift, unshift, splice, slice)
- Functional array methods (map, filter, reduce, forEach, find, some, every)
- Array destructuring
- Spread operator with arrays and objects

Episode 46-55: Asynchronous JavaScript

- Understanding the event loop
- Callbacks and callback patterns
- Promises (creation, chaining, error handling)
- Promise.all, Promise.race, Promise.allSettled
- Async/await syntax and error handling
- Fetch API and HTTP requests

- XMLHttpRequest (legacy approach)
- Handling JSON data
- Error handling in async code
- Microtasks vs macrotasks

Episode 56-60: ES6+ Features

- Template literals and tagged templates
- Symbols and their use cases
- Iterators and the iterator protocol
- Map and Set data structures
- WeakMap and WeakSet
- Proxy and Reflect APIs

ADVANCED LEVEL (Episodes 61-85)

Episode 61-70: Object-Oriented Programming

- Prototypes and prototype chain
- Constructor functions and new operator
- ES6 Classes and inheritance
- Static methods and properties
- Private fields and methods
- Mixins and composition
- Design patterns (Module, Singleton, Factory, Observer)
- Encapsulation, inheritance, and polymorphism
- instanceof and isPrototypeOf
- Object.create and Object.setPrototypeOf

Episode 71-80: Advanced Concepts

- Execution context and call stack

- Hoisting (variables and functions)
- Temporal dead zone
- Memory management and garbage collection
- Event delegation and event bubbling/capturing
- DOM manipulation best practices
- Web APIs (localStorage, sessionStorage, IndexedDB)
- Regular expressions and pattern matching
- Modules (CommonJS, AMD, ES6 modules)
- Package managers (npm, yarn)

Episode 81-85: Performance and Optimization

- JavaScript performance optimization
- Memory leaks and how to avoid them
- Debouncing and throttling
- Lazy loading and code splitting
- Web Workers for background processing
- Service Workers and caching strategies

EXPERT LEVEL (Episodes 86-100)

Episode 86-95: Advanced Topics

- Metaprogramming with Proxy
- Abstract Syntax Trees (AST)
- Babel and transpilation
- Webpack and build tools
- Testing (Jest, Mocha, Jasmine)
- TypeScript integration
- Functional programming concepts

- Immutability and immutable data structures
- State machines and finite automata
- Algorithm complexity (Big O notation)
- System design with JavaScript