

Advanced JavaScript with TypeScript Course

Course Duration: 24 Hours

Course Code : JS-TS201

1. Course Overview

- The **Advanced JavaScript with TypeScript Course** is designed for developers who already have a basic understanding of JavaScript and want to advance their skills by mastering modern JavaScript features and TypeScript. This course covers advanced concepts like asynchronous programming, closures, modules, object-oriented programming, and TypeScript fundamentals such as types, interfaces, generics, and decorators. By the end of the course, learners will be able to build robust, scalable, and maintainable applications using JavaScript and TypeScript.

2. What You'll Learn?

- Deep dive into **modern JavaScript (ES6+)** features.
- Work with **closures, prototypes, and object-oriented JavaScript**.
- Master **asynchronous programming** (Promises, async/await, fetch API).
- Explore **modules, bundlers, and package managers (NPM/Yarn)**.
- Introduction to **TypeScript** and its advantages over JavaScript.
- Work with **types, interfaces, enums, generics, and modules in TypeScript**.
- Implement **object-oriented programming in TypeScript**.



- Use **TypeScript with Node.js** and frameworks like **Angular/React**.
- Error handling, debugging, and best practices.
- Build **real-world projects** using JavaScript + TypeScript.

3. Target Audience

- Developers with a **basic understanding of JavaScript** who want to upgrade their skills.
- Front-end and back-end developers looking to transition to **TypeScript**.
- Professionals aiming for careers in **full-stack development**.
- Anyone preparing for **JavaScript/TypeScript technical interviews**.

4. Pre-Requisites

- Knowledge of **HTML & CSS**.
- Basic understanding of **JavaScript fundamentals**.

5. Course Content

- **Module 1: Advanced JavaScript Essentials**
- ES6+ features (let/const, arrow functions, template literals)
- DE structuring & spread/rest operators
- Modules and imports/exports
- Iterators & Generators
- **Module 2: Object-Oriented JavaScript**
- Prototypes and inheritance
- Classes and objects (ES6 OOP)



- Encapsulation, abstraction, and polymorphism
- **Module 3: Asynchronous JavaScript**
- Callbacks vs Promises vs async/await
- Fetch API & working with APIs
- Error handling in async code
- **Module 4: Advanced JavaScript Concepts**
- Closures and lexical scope
- Event loop, call stack & concurrency model
- Higher-order functions & functional programming concepts
- **Module 5: Introduction to TypeScript**
- Why TypeScript? Benefits over JavaScript
- Setting up TypeScript environment (tic, tsconfig.json)
- Basic types, arrays, tuples, and Enums
- Type inference and type assertions
- **Module 6: Functions & Object-Oriented TypeScript**
- Function types and optional/default parameters
- Interfaces and type aliases
- Classes, inheritance, access modifiers (public, private, protected)
- Abstract classes and polymorphism
- **Module 7: Advanced TypeScript Features**
- Generics and reusable code
- Modules and namespaces
- Decorators in TypeScript
- Working with third-party libraries and Definitely Typed
- **Module 8: TypeScript in Real Projects**



- Integrating TypeScript with Node.js
- Using TypeScript with React/Angular
- Configuring Webpack/Babel for TypeScript
- Debugging and linting TypeScript code
- **Module 9: Error Handling & Best Practices**
- Compile-time vs runtime errors
- Exception handling in TypeScript
- Writing clean, maintainable code
- **Module 10: Capstone Project**
- Building a **Full-Stack Application with JavaScript + TypeScript**
(Examples: Task Manager, E-commerce App, Blog API).

