



TECHNOCATION FREELANCING TRAINING INSTITUTE & SOFTWARE HOUSE

Professional TypeScript Course Outline

Module 1: Introduction to TypeScript

- What is TypeScript & Why Use It?
 - TypeScript vs. JavaScript
 - Installing & Setting Up TypeScript (Node.js, npm, tsc)
 - TypeScript Compiler (`tsc`), Compilation Targets & Configuration (`tsconfig.json`)
 - Running TypeScript Code (`ts-node`, `tsc --watch`)
-

Module 2: TypeScript Basics & Data Types

- Primitive Types (`string`, `number`, `boolean`, `null`, `undefined`, `symbol`, `bigint`)
 - Object Types (`arrays`, `tuples`, `enums`, `interfaces`, `types`)
 - Type Inference & Type Assertions (`as`, `<>` syntax)
 - Union & Intersection Types (`|`, `&`)
 - Literal & Template Literal Types
-

Module 3: Functions & Object-Oriented Programming (OOP)

- Function Types & Signatures
- Optional & Default Parameters
- Arrow Functions & `this` in TypeScript
- Classes & Object-Oriented Programming
 - Inheritance & Extending Classes
 - Access Modifiers (`public`, `private`, `protected`, `readonly`)
 - Static Members & Getters/Setters
- Abstract Classes & Interfaces

Module 4: Advanced TypeScript Features

- Generics (`Array<T>`, Generic Functions, Generic Constraints)
- Type Guards (`typeof`, `instanceof`, Custom Type Guards)
- Utility Types (`Partial`, `Pick`, `Omit`, `Readonly`, `Record`)
- Mapped Types & Conditional Types
- Decorators & Meta-Programming

Module 5: TypeScript with Modules & Namespaces

- Working with ES Modules & CommonJS
- Importing & Exporting Modules (`import`, `export`, `export default`)
- Namespaces & Module Resolution
- Using TypeScript with Webpack & Babel

Module 6: TypeScript with Frontend Frameworks

- Using TypeScript with React.js (`tsx`, Props & State)
- Using TypeScript with Angular (`decorators`, `services`, `interfaces`)
- TypeScript & Vue.js (`composition API`, `Vuex`)
- Handling Global State with TypeScript (`Redux`, `Zustand`, `Pinia`)

Module 7: TypeScript for Backend Development

- Using TypeScript with Node.js
- Setting Up Express with TypeScript
- TypeScript with GraphQL (Apollo Server, Type-GraphQL)
- Working with Databases (`TypeORM`, `Prisma`, `Mongoose`)
- Building REST APIs with TypeScript & Express

Module 8: Error Handling & Debugging in TypeScript

- Handling Runtime & Compile-Time Errors

- Debugging TypeScript with VS Code
 - Working with `try-catch` & Custom Error Classes
 - Logging & Monitoring TypeScript Applications
-

Module 9: Testing & Best Practices

- Unit Testing with Jest & TypeScript
 - Using Mocha & Chai for Backend Testing
 - End-to-End Testing with Cypress
 - Best Practices for Writing Scalable TypeScript Code
 - Code Linting & Formatting (ESLint, Prettier)
-

Module 10: Deploying TypeScript Applications

- Compiling TypeScript for Production (`tsconfig` optimization)
 - Deploying TypeScript Apps on AWS, Vercel, & Firebase
 - Using Docker with TypeScript
 - Working with CI/CD Pipelines (GitHub Actions, Jenkins)
-

Real-World Projects & Deployment

- Building a Full-Stack MERN/MEAN App with TypeScript
- Developing a Scalable E-Commerce Platform
- Creating a Real-Time Chat App with TypeScript & WebSockets
- Implementing a Secure Authentication System (JWT, OAuth)
- Deploying a Microservices Architecture with TypeScript & Docker