Master in TypeScript with NestJs

About
DevOpsSchool

DevOpsSchool is a unit of "Cotocus PVT ltd" and a leading platform which helps IT organizations and professionals to learn all the emerging technologies and trend which helps them to learn and embrace all the skills, intelligence, innovation and transformation which requires to achieve the end result, quickly and efficiently. We provide over 40 specialized programs on DevOps, Cloud, Containers, Security, AI, ML and on Big data that are focused on industry requirement and each curriculum is developed and delivered by leading experts in each domain and aligned with the industry standards.

About Course

Typescript is a superset of JavaScript language contains a single open-source compiler and it is mainly developed by a single vendor: Microsoft. TypeScript aims to help catch mistakes early and make JavaScript development more efficient through the type system.

NestJS is used for writing scalable, testable and loosely coupled applications. It brings scalable Node. js servers to a whole new level. It supports databases like PostgreSQL, MongoDB, MySQL.NestJS developers helps in creating highly scalable, modular and maintenance qualified server-side Web applications. It can use the MVC (Model-View-Controller) pattern, easy to implement and you take advantage of the Express framework to provide extensibility, as any third-party modules available for the Express.

This tutorial is designed for TypeScript OR NestJs who are completely unaware of TypeScript and NestJs concepts but have a basic understanding of computer programming. This course covers all the key concepts from basic level to advanced level. After the training completion, will provide you training completion certification. And also provide you learning materials like Class recordings, Notes Slides, etc.

Co-coordinator – Akanksha Kumari

Call/WhatsApp: - +91 1800 889 7977

Mail Address: -

contact@DevOpsSchool.com

Secondary contact – Patrick

Call/WhatsApp: - +91 7004 215 841

Mail Address: -contact@DevOpsSchool.com

Duration	30 Hours
Mode	Online (Instructor-led, live & Interactive)
Projects (Real time scenario based)	1

FEATURES	DEVOPSSCHOOL	OTHERS
Faculty Profile Check	✓	✗
Lifetime Technical Support	✓	✗
Lifetime LMS access	✓	✗
Top 25 Tools	✓	✗
Interviews Kit	✓	✗
Training Notes	✓	✗
Step by Step Web Based Tutorials	✓	✗
Training Slides	✓	✗
Training + Additional Videos	✓	✗

AGENDA OF THE MASTER IN TYPESCRIPT WITH NESTJS

Getting Started with TypeScript

- Course introduction
- Why use TypeScript
- TypeScript Features
- TypeScript Syntax, Keywords, and Code Hierarchy
- Tooling and Framework Options
- Tooling and Framework Options - TypeScript Playground
- Tooling and Framework Options - Visual Studio
- Tooling and Framework Options - Web Essentials
- Tooling and Framework Options - Sublime Text
- Tooling and Framework Options - TypeScript Compiler
- Tooling and Framework Options - NodeJS
- Hello World Example
- Hello World Example - Creating a Class

Typing, Variables, and Functions

- Overview
- Grammar, Declarations, and Annotations
- Type Inference
- Grammar
- Static and Dynamic Typing
- Compile Time or Run Time
- Ambient Declarations and Type Definition Files
- The Any Type and Primitives
- Applying Types
- Objects
- Functions
- Arrow Functions and Debugging
- Functions and Interfaces
- Static Typing Recap

Classes and Interfaces

- Introduction
- Defining Classes
- Demo: Defining Classes
- Demo: Property Limitations
- Casting and Type Definition Files
- Demo: Casting and Type Definition Files
- Extending Types
- Demo: Extending Types
- Using Interfaces
- Demo: Using Interfaces
- Extending an Interface
- Demo: Extending an Interface

Modules

- Overview
- Identifying a Module
- Creating an Internal Module
- Internal Module Accessibility and IIFE
- Named Modules
- Extending Modules and Importing Shortcuts
- Organizing Internal Modules
- Separating Internal Modules
- External Modules and Dependency Resolution
- Module Dependencies
- Importing External Modules Using AMD
- Importing 3rd Party Libraries Using AMD
- Modules Recap

Getting Started - NestJS

- Introduction to NestJS
- Installing the NestJS CLI (command-line interface)
- What’s inside a NestJS Application
- What we'll be building in this course
- Beginning your NestJS Journey

Creating a REST API application

- Install Insomnia
- Running NestJS in Development Mode
- Creating a Basic Controller
- Use Route Parameters
- Handling Request Body / Payload
- Response Status Codes
- Handling Update and Delete Requests
- Implement Pagination with Query Parameters
- Creating a Basic Service
- Send User-Friendly Error Messages
- Encompass Business-Domain in Modules
- Introduction to Data Transfer Objects
- Validate Input Data with Data Transfer Objects
- Handling Malicious Request Data
- Auto-transform Payloads to DTO instances

Add PostgreSQL with TypeORM

- Before we Get Started
- Install Docker
- Running PostgreSQL
- Introducing the TypeORM Module
- Creating a TypeORM Entity
- Use Repository to Access Database
- Create a Relation between two Entities
- Retrieve Entities with their Relations
- Using Cascading Inserts and Updates
- Adding Pagination
- Use Transactions
- Adding Indexes to Entities
- Setting up Migrations

Dependency Injection

- Understand Dependency Injection
- Control NestJS Module Encapsulation
- Diving Into Custom Providers
- Value-based Providers
- Non-class-based Provider Tokens
- Class Providers
- Factory Providers
- Leverage Async Providers
- Create a Dynamic Module
- Control Providers Scope
- Diving Deeper Into Request-Scoped Providers

Application Configuration

- Introducing the Config Module
- Custom Environment File Paths
- Schema Validation
- Using the Config Service
- Custom Configuration Files
- Configuration Namespaces and Partial Registration
- Asynchronously Configure Dynamic Modules

Other Building Blocks by Example

- Introducing More Building Blocks
- Understanding Binding Techniques
- Catch Exceptions with Filters
- Protect Routes with Guards
- Using Metadata to Build Generic Guards or Interceptors
- Add Pointcuts with Interceptors
- Handling Timeouts with Interceptors
- Creating Custom Pipes

Generating OpenAPI Specification

- Introducing the Swagger Module
- Enabling CLI Plugin
- Decorating Model Properties
- Adding Example Responses
- Using Tags to Group Resources

Testing

- Introduction to Jest
- Getting Started with Test Suites
- Adding Unit Tests
- Diving Into e2e Tests
- Creating our First e2e Test
- Implementing e2e Test Logic

Thank you!

Connect with us for more info

Call/WhatsApp: - +91 968 682 9970

Mail: -

contact@DevOpsSchool.com

www.DevOpsSchool.com