
GraphQL Course Content

Duration : 6 Half Days (4 Hours/day)

Prerequisites

Participants must have at least 2 years of working with Javascript. Nodejs experience is not required but would be handy. This course is for JavaScript developers looking to learn GraphQL. We would be integrating ApolloClient with React, React basics would be required.

Lab SetUp

- Visual Studio Code (Latest)
- Node JS (v16+)
- Chrome Browser
- Internet connection for installing npm packages
- Access to npmjs.com, <https://www.apollographql.com/docs/>

Agenda

Typescript

DAY 1:

- What is TypeScript and why to use it?
 - Conceptual comparison between JavaScript and TypeScript
 - Tooling and Framework Options
 - TypeScript Playground
 - Visual Studio Code
 - Installing TypeScript using NPM
 - Configure VS Code with Build
 - Understanding TypeScript Configuration
 - Understanding Transpilation
 - Typing, Declarations
 - Declaring Variables
 - Type Inferencing
 - Static vs Dynamic Declarations
-

- Basic Data Types
 - var vs let keyword
 - const
 - Type Assertions
 - Nullish Coalescing
- Functions
 - Understanding Functions
 - Function Parameters
 - Function Overloading
 - Arrow Functions
2. Basic Datatypes revisited
- Enum
 - Array
 - Tuple
 - De-structuring

DAY 2:

3. Interfaces
- Creating Object of Type Interface
 - Optional Properties
 - Read-only Properties
 - Functions Declarations
 - Using interface in
 - Function Parameter
 - Return
 - Extending Interface
 - Implementing interface on class
4. Classes
- Creating Class
 - Constructors
 - Fields
 - Access Modifiers
 - Private
 - Public
 - Protected
 - Inheritance
 - Accessors
 - Static Properties
 - Abstract Class
 - Optional Properties
 - Strict Check
5. Generics
- Using Generics with
 - Function

- Class
- Interface
- Generic Constraints

GraphQL

DAY 3:

Introduction

- What is GraphQL?
- GraphQL compared to REST Services
- Apollo compared to Flux, Redux, and Relay
- Benefits of GraphQL
- Benefits of Apollo Client, Server and Engine

GraphQL Types and Server Configuration

- Scheme, Object, Scalar, & List Types
- Interface Types
- Understanding Fields, Arguments and Resolve

GraphQL Queries

- Using the GraphQL web-based client for making queries and exploring documentation
- Querying connected nodes on the graph, and exploring how that works.
- Passing arguments into queries.
- Renaming a field with aliasing.
- Breaking a large query into fragments.
- Nesting many fragments.
- Exploring Apollo GraphQL Client Requests with Tooling

DAY 4 :

GraphQL Mutations

- Explore Mutation Queries and Input Types
- Execute Mutation Queries with Variables
- Structuring Mutation Query Results
- Configuring Mutations with Apollo Client

- Store Updates, Optimistic Responses
- Handling Error from Failed Mutations
- Create Task Mutation
- Modularize Resolvers
- Modularize Schema (Schema Stitching)
- Resolver Context
- Update/Modify Task Mutation
- Delete Task Mutation

DAY 5:

- Offset Limit Pagination (Task List)
- Offset limit Pagination Vs Cursor Based Pagination
- Cursor Based Pagination (Task List)
- Data Loaders- What and Why
- Implementing Data Loaders

Additional GraphQL Topics

- Enum, Union, Interface Types
- Inline Fragments
- Directives
- Introspection Queries
- Pre-fetching Data
- Subscriptions

DAY 6:

Connecting to React

- Using the web browser fetch API to connect to a GraphQL server
- Using Google Chrome extensions to examine GraphQL queries
- Migrate an existing React application from REST Services to GraphQL

Connecting to React using Apollo Client

- Setting up the Apollo Networking Environment
- Using the Apollo Provider to connect GraphQL to React
- Co-locating the GraphQL Query with the React component
- Containers with GraphQL function and decorator
- Container Config API
- Container with Query API

Conclusion