

# Full Stack Developer Training

Contact: pnaveenmscit@gmail.com, 9600276084

1. Angular 10 with Docker containerization
2. Dot Net Core with Docker containerization
3. Docker
4. Sql Server
5. CICD Pipeline
6. GitHub or Gitlab

## Angular -Practical oriented Classes

### Setup

Install of Angular, Node

Create Initial Application

### ES6

Module system

Classes

Arrow Functions

Template Strings

### Angular-CLI & project structure

Creating a new project

Project settings, bootstrapping

Building and serving

Component-based architecture

Angular building blocks overview

Generating project elements

Root Angular Module

### Angular Components

Component definition

User Input, Pipes

Component types

Component Interaction

Angular Elements

Template syntax

Data, property and event binding

Using directives and pipes

Inputs and Outputs (events) o Component style

## **Templates**

Binding Syntax

Property Binding

Attribute Classes and Style bindings

Two-way binding

Model classes

Directives

## **Providers, Dependency Injection, Observables**

Providers and Dependency Injection

Understanding the role of the Provider

Understanding the injector tree

Creating and using a class provider (service)

Other provider strategies

Configuring providers

Observables and RxJS o Subscription o RxJS Operators

Creating Subjects and Observables

HttpClient o Http requests (GET, PUT, and POST)

Configuring headers o Interceptors o Progress events

Making Http Requests

Handling Http Responses

## **Angular Forms**

Template-driven forms

Reactive forms

Form Builder

Form validation

Custom validators

Async validators

Angular Router

### **Setting up the router**

Navigation

Child routes

Routing prams

Lazy loading

Guards and hooks

### **Testing**

Introduction to Angular testing

Testing using Karma

Code Coverage

E2E Testing

### **Advanced Concepts**

Integrating the third party components

Payment Integration few

Prime Faces

Angular Material

Real Time Project Training

State management with ngRx

Understanding Redux architecture

NgRx Store

Actions & Reducers

JWT Authentication

Encryption and Decryption

Deployment CICD

Docker

Types of Storage

Interview Questions

### **Docker-Practical oriented Classes**

1. Windows Containers

2. Linux Container

Docker objects  
Docker Images  
Docker Container  
Docker commands  
Docker File  
Docker Compose File  
The base image from Docker  
Copying source code  
Exposing a port  
Building Docker images  
Running a Docker image

## **Dot Net Core-Practical oriented Classes**

Install Dot Net Core  
Create ASP.NET Core Application  
ASP.NET Core - Project Structure  
Advantages of ASP.NET Core  
ASP.NET Core Features  
ASP.NET Core - Dependency Injection  
ASP.NET Core - Environment Variable  
ASP.NET Core - Exception Handling  
ASP.NET Core Environment Setup  
ASP .NET Core First Application  
Project Layout  
Understanding Life Cycle of ASP.Net Core Request  
Authentication and Authorization  
Implementing Security using ASP.NET Core Identity  
CRUD Operations using Entity Framework  
JWT Authentication  
Docker Containerization

Code First Approach

Database First Approach

OOPS Concepts

Class Library

Swagger Implementation

### **Rest Methods**

GET, PUT, POST, DELETE

Implement Repository, Unit of Work and Dependency Injection Design Pattern.

Ado.Net

### **Using EF Approaches Code First, Model**

First, Database first

Implementing a Repository and Entity

Framework Data Model

Define Dependency Injection

Using Dependency Injection

Accessing a Repository in controller

Docker

CICD Pipeline

Create Docker Files

Compose Docker File

### **Sql server -Practical oriented Classes**

- Database Architecture
- Data Manipulation Language (DML)
- Data Definition Language (DDL)
- Manipulation of Data (SQL Command)
- Stored Procedure
- Function
- Trigger
- Views
- Cursor

- SQL - Databases
- SQL - Syntax
- SQL - Data Types
- SQL - Operators
- SQL - Expressions
- SQL - Create Database
- SQL - Drop Database
- SQL - Select Database
- SQL - Create Table
- SQL - Drop Table
- SQL - Insert Query
- SQL - Select Query
- SQL - Where Clause
- SQL - AND & OR Clauses
- SQL - Update Query
- SQL - Delete Query
- SQL - Like Clause
- SQL - Top Clause
- SQL - Order By
- SQL - Group By
- SQL - Distinct Keyword
- SQL - Sorting Results
- Advanced SQL
- SQL - Constraints
- SQL - Using Joins
- SQL - Unions Clause
- SQL - NULL Values
- SQL - Alias Syntax
- SQL - Indexes
- SQL - Alter Command
- SQL - Truncate Table
- SQL - Using Views
- SQL - Having Clause
- SQL - Transactions

- SQL - Wildcards
- SQL - Date Functions
- SQL - Temporary Tables
- SQL - Clone Tables
- SQL - Sub Queries
- SQL - Using Sequences
- Sub queries
- Nested sub queries
- Correlated sub queries
- Derived tables
- Joins
- Introduction to Joins
- Inner join
- Outer join
- Cross joins
- Unions
- Working with Indexes
- Introduction to indexes
- Creating, dropping indexes
- Complex indexes
- Clustered & non clustered indexes
- Implementing Views
- Introduction & advantages of views
- Creating views
- Altering, dropping views
- Cursors
- Views

## **GitLab and GitHub -Practical oriented Classes**

- Installing GitLab on Windows
- Important Git Commands
- SSH Setup
- Forking a Project

- Branching
- Adding Files