Creating Object-oriented TypeScript Code

INTRODUCTION TO OBJECT-ORIENTED PROGRAMMING IN TYPESCRIPT

Dan Wahlin
WAHLIN CONSULTING

@danwahlin www.codewithdan.com

Course Overview

Introduction to Object-oriented TypeScript

Classes and Interfaces

Classes and Objects

Putting It All Together

Inheritance and Abstraction

Target Audience

Developers looking to enhance their knowledge of TypeScript

Course Prereqs

Prior experience using HTML, JavaScript, and TypeScript

Introduction

Module Overview

The Role of Objects

Object Creation Techniques

Object-oriented Concepts

The Role of Objects

Source Code Link

https://github.com/DanWahlin/Angular-Architecture

https://github.com/DanWahlin/Angular-JumpStart

Software Requirements

Node.js

https://nodejs.org

Angular CLI

https://cli.angular.io

VS Code

https://code.visualstudio.com
(any editor can be used)

Object-oriented Concepts

Course Overview

Object-oriented Programming (OOP)

Programs are composed of objects which communicate with each other, which may be arranged into hierarchies, and which can be combined to form additional objects.

OOP Benefits

- Code reuse
- Faster development time frames
- Real-world mapping of objects
- Modular architecture
- More maintainable code base

Principles of Object-oriented Programming

Abstraction

Encapsulation

Inheritance

Polymorphism

Summary

JavaScript/TypeScript provides several different ways to create objects

JavaScript is a type of object-oriented language based on prototype-based inheritance

Key object-oriented concepts include:

- Abstraction
- Encapsulation
- Inheritance
- Polymorphism