

SEMESTER-1

Strengthening Foundations & TypeScript Mastery

(November 2024 - February 2025)

Part 1: First Project (Learning)

Timeline: November 17 - December 10, 2024

Learning Focus: TypeScript Basics

1. TypeScript Basics:

- a. **Introduction to TypeScript:** Understanding the benefits of TypeScript over JavaScript.
- b. **Installing and Setting Up TypeScript:** Installing TypeScript and configuring it in your project.
- c. **Types:**
 - i. Basic types (string, number, boolean, array, tuple, enum).
 - ii. Type inference and type annotations.
- d. **Variables:** Declaring variables using `let`, `const`, and `var`.
- e. **Interfaces:** Defining simple interfaces for object types, arrays, and functions.
- f. **Classes and Objects:** Basic classes, constructors, and methods.
- g. **Function Types:** Understanding how to type functions and return types.
- h. **Type Aliases:** Defining custom types with `type` keyword.

Project 1: User Management CRUD Application

- Build a simple CRUD app that allows users to add, update, view, and delete user data.
- This project will involve implementing:
 - **Create:** Add new users (using TypeScript classes).
 - **Read:** View existing users (using arrays and objects).
 - **Update:** Modify user data (via TypeScript methods).
 - **Delete:** Remove a user from the list.

Outcome: A fully functioning **User Management CRUD app** using TypeScript basics, such as classes, interfaces, types, and functions.

Part 2: Midterm Project (Implementation Exam)

Timeline: December 11 - December 15, 2024

Focus: Implement everything learned in Part 1.

- **Objective:** Build a project that shows understanding of TypeScript basics and CRUD functionality.
- **Project:** Implement a **CRUD app (User Management)**, showcasing TypeScript basics and CRUD operations.
 - **Create:** Implement a form to add users.
 - **Read:** Display the list of users.
 - **Update:** Allow users to edit their data.
 - **Delete:** Allow users to be removed from the list.

Goal: Demonstrate mastery of **TypeScript basics** (types, interfaces, functions, and classes) and CRUD functionality.

Part 3: Second Project (Learning)

Timeline: December 16 - December 25, 2024

Learning Focus: Advanced TypeScript Concepts

1. **Generics:**
 - a. Creating reusable components with generics (classes, functions, and interfaces).
 - b. Understanding when to use generics to make types flexible.
 - c. Practical examples like generic functions for CRUD operations.
2. **Enums:**
 - a. How and when to use enums for a set of related constants (e.g., user roles).
 - b. Define enums to represent different states or categories.
3. **Decorators:**
 - a. Introduction to decorators (function decorators, class decorators).
 - b. How to use decorators for logging, validation, etc.
4. **Type Assertions:**

- a. Using `as` keyword to override TypeScript's inferred types.

5. **Modules and Namespaces:**

- a. Structuring a TypeScript project with modules.
- b. How to organize and export types, interfaces, and functions.

Project 2: Post Management CRUD Application

- Build a **Post Management CRUD app** that allows you to create, read, update, and delete posts.
- This project will involve implementing:
 - **Create:** Add posts with type-safe data.
 - **Read:** Display all posts.
 - **Update:** Edit post content.
 - **Delete:** Remove posts.
 - **Advanced TypeScript:** Use generics for flexible post operations, implement enums for post categories or status, and apply decorators for post validation or logging.

Outcome: A fully functioning **Post Management CRUD app** using advanced TypeScript features like generics, enums, and decorators.

Part 4: Final Project (Implementation Exam)

Timeline: December 26 - December 31, 2024

Focus: Combine **TypeScript basics** and **advanced TypeScript** into a comprehensive CRUD application.

- **Project:** Build a **Full CRUD app** for both **User and Post Management** with TypeScript.
 - Integrate both **User Management CRUD** and **Post Management CRUD** in one project.
 - **Advanced Features:** Use **generics** to handle user and post data in a flexible, reusable way.
 - Implement **enums** for user roles (admin, user) and post status (published, draft).
 - Apply **decorators** for validation and logging.

- Make sure all CRUD operations are working seamlessly for both users and posts.

Goal: Build a **final project** that reflects everything learned during the semester, with full CRUD functionality, TypeScript basics and advanced features implemented in real-time.

Summary for Phase 1 (November 17 - December 31, 2024)

1. First Project (Learning):

- a. **Focus:** TypeScript basics (types, classes, interfaces, and functions)
- b. **Project:** User Management CRUD app
- c. **Goal:** Learn TypeScript basics and implement basic CRUD functionality.

2. Midterm Project (Implementation Exam):

- a. **Focus:** Implement TypeScript basics and CRUD functionality
- b. **Project:** CRUD app (User Management)
- c. **Goal:** Demonstrate mastery of TypeScript basics and CRUD.

3. Second Project (Learning):

- a. **Focus:** Advanced TypeScript (generics, enums, decorators, type assertions)
- b. **Project:** Post Management CRUD app
- c. **Goal:** Learn advanced TypeScript features and apply them in a CRUD app.

4. Final Project (Implementation Exam):

- a. **Focus:** Combine TypeScript basics and advanced features with CRUD functionality
- b. **Project:** Full CRUD app (User and Post Management)
- c. **Goal:** Showcase full mastery of TypeScript and CRUD operations.