

Week 2 Internship Report

Name: Akalya Ravichandran

Role: Frontend Developer Intern

Week: Week 2

Duration: Monday – Friday

1. Objective of the Week

The primary objective for Week 2 was to enhance proficiency in modern JavaScript (ES6+), understand asynchronous programming patterns, gain experience with API integration, and establish foundational knowledge in TypeScript for type-safe development.

2. ES6+ Features

Topics Covered

- Arrow functions
- Destructuring (arrays and objects)
- Spread and rest operators
- ES6 modules and import/export syntax

Practical Exercises

- Converted traditional functions to arrow functions
 - Used destructuring to extract values from objects and arrays
 - Applied spread and rest operators in functions and arrays
-

3. Array Methods

Topics Covered

- `map` for transforming arrays
- `filter` for conditional selection
- `reduce` for aggregation
- `find` for locating elements

Practical Exercises

- Solved 15+ JavaScript practice problems focusing on ES6+ and array methods
-

4. Asynchronous JavaScript

Topics Covered

- Promises
- Async/Await syntax
- Error handling with try/catch

Practical Exercises

- Built an asynchronous mini-project that fetches data from a public API
 - Handled successful responses and error cases gracefully
-

5. Working with APIs

Topics Covered

- `fetch` API for network requests
- Parsing JSON data

Practical Work

- Integrated external API endpoints into mini-projects
 - Displayed fetched data dynamically on the web page
-

6. TypeScript Basics

Topics Covered

- Basic types (string, number, boolean, array, tuple, any)
- Interfaces and type aliases
- Type inference and type checking
- Functions with typed parameters and return types
- Objects with typed properties

Practical Exercises

- Converted existing JavaScript mini-project to TypeScript
 - Ensured type safety and resolved type errors
-

7. Deliverables Summary

- 15+ JavaScript ES6+ practice problems solved
 - Asynchronous mini-project completed using fetch and API integration
 - Existing JavaScript project successfully converted to TypeScript
 - TypeScript basics assessment passed
-

8. Challenges Faced

- Adapting to asynchronous patterns and handling errors initially
- Understanding TypeScript type system for objects and functions

Resolution: Overcame challenges through tutorials, documentation reference, and hands-on practice.

9. Learnings & Takeaways

- ES6+ features significantly simplify JavaScript coding patterns
- Async/Await provides clean, readable asynchronous code
- TypeScript improves code reliability and maintainability