# 🛠️ Tech Stack

- JavaScript (ES6+)
- Node.js
- NestJS
- PostgreSQL
- Prisma ORM
- Swagger (OpenAPI)
- Git/Github
- Docker Basics
- Redis Basics

# 📅 Week 1: JavaScript + Node.js + Git/GitHub Basics

## 🎯 Goal

Build a strong foundation in modern JavaScript, understand how Node.js works, and get comfortable with Git and GitHub version control.

---

## 📘 Topics Covered

### 1. JavaScript Fundamentals (ES6+)

#### ✅ Concepts

- Variables (`let`, `const`, `var`)
- Data Types: Strings, Numbers, Booleans, Null, Undefined, Objects, Arrays
- Conditional Statements (`if`, `else`, `switch`)
- Loops (`for`, `while`, `forEach`, `map`)
- Functions: Function declarations, expressions, arrow functions
- Objects and Arrays: Creation, access, mutation
- Destructuring, Spread/Rest Operators
- Template literals

#### 📚 Study Material

- [JavaScript.info - The Modern JavaScript Tutorial](#)
- [Eloquent JavaScript (Free Book)](#)

## 2. Asynchronous JavaScript

### ✅ Concepts

- Callback functions
- Promises
- `async` / `await`

### 📚 Study Material

- [MDN: Promises](#)
- [JavaScript.info - Async](#)

## 3. Node.js Basics

### ✅ Concepts

- Introduction to Node.js & Event Loop
- Global objects (`__dirname`, `__filename`, `process`, etc.)
- File System Module (`fs`)
- HTTP Module: Creating a basic server
- Path module
- NPM: `init`, installing packages, `package.json`
- Creating and running custom scripts with `npm run`

### 📚 Study Material

- [Node.js Docs](#)
- [Node.js Crash Course - Traversy Media](#)

## 4. Git & GitHub Basics

### ✅ Concepts

- Version control concepts
- Git Commands: `git init`, `clone`, `add`, `commit`, `status`, `log`, `push`, `pull`
- Branching: `checkout`, `merge`
- Working with GitHub: creating repo, pushing code, README basics
- Git Ignore and Git Logs

### 📚 Study Material

- [Git Handbook - GitHub Docs](#)
- [Git Crash Course - Traversy Media](#)

---

# 🧪 Week 1 Assignments & Activities

## ✅ Practice Tasks

- Build a CLI tool using Node.js (e.g., a TODO app that saves data in a JSON file)
- Create a GitHub repository and push your code with proper commits
- Try branching and merging in Git

## ✅ Mini Project

**Title:** Simple Note-Taking API (no DB)

- Use `fs` module to save notes in a JSON file
- Implement `create`, `read`, `delete` note functionality
- Create a GitHub repo, write a README, and push your project