

Training Day 11 Daily Dairy

Day 11 – 3rd July 2025

TOPICS COVERED – Node.js, React, Vite Setup & API Revision

1. Installed Node.js

- Node.js is a JavaScript runtime environment that allows JS to run outside the browser.
- Required for running React and other modern frontend tools.
- Confirmed installation using:

```
node -v  
npm -v
```

2. React Introduction

- React is a JavaScript library for building user interfaces.
- Works using **components**, **JSX**, and **state**.
- Useful for creating **dynamic single-page applications (SPA)**.

3. Project Setup using Vite

- Vite is a fast modern build tool to scaffold React projects.
- Steps followed:

```
npm create vite@latest
```

- Project folder created with:
 - index.html
 - src/ (source code)
 - public/ (static assets)

- node_modules/ (installed dependencies)
- vite.config.js (Vite configuration)

4. File Structure Explained

- node_modules/ – All npm packages are stored here.
- public/ – Static files like images, favicon.
- src/ – Source folder where App.jsx and components are written.
- index.html – Root HTML file where React mounts.
- package.json – Lists dependencies and scripts.
- vite.config.js – Custom configurations for Vite.
- .gitignore – Lists files to ignore in git.

5. Running the Project

```
npm install  
npm run dev
```

- Runs the Vite dev server.
- Opens the React app in the browser at localhost:5173.

6. API Revision

- Revisited API concepts using fetch():

```
fetch('https://jsonplaceholder.typicode.com/posts')  
  .then(res => res.json())  
  .then(data => console.log(data));
```

- Also revised:
 - async/await syntax
 - Handling .catch() for errors

- Displaying API data in UI (briefly touched with React)

TASK FOR TOMORROW

- Learn and implement **React component creation**.
- Understand and use **JSX, Props, and State**.
- Build a simple component-based app (like a todo or counter).