

# React.js Syllabus

**Objective:** Learn the basics of React to build interactive, component-based user interfaces.

## Module 1: Introduction

- What is React.js and why use it
- Understanding **Single Page Applications (SPA)**
- Setting up development environment:
  - Node.js & npm
  - Create React App / Vite
  - Code editors (VS Code)

## Module 2: JSX & Components

- JSX syntax (HTML in JS)
- Functional vs class components
- Props (passing data to components)
- Rendering lists using `.map()`
- Handling nested components

## Module 3: State Management

- Using `useState` hook
- Updating state and re-rendering
- Conditional rendering
- Event handling (`onClick`, `onChange`)

## Module 4: Component Lifecycle & Effects

- `useEffect` hook basics
- Side effects: API calls, timers
- Cleanup functions

## Module 5: Forms & User Input

- Controlled components
- Handling form submission
- Basic validation

## Module 6: Routing

- React Router basics
- Navigating between pages

- Passing parameters in routes

## **Module 7: Project: Simple React App**

- Example: To-Do List
  - Focus on:
    - Components
    - State
    - Props
    - Event handling
-

# ☐ Node.js & Express.js Syllabus

**Objective:** Learn backend fundamentals using Node.js and Express to build APIs and handle server-side logic.

## Module 1: Introduction to Node.js & Express

- What is Node.js
- Understanding backend vs frontend
- Installing Node.js and npm
- Creating your first Express server

## Module 2: Routing & HTTP Methods

- Understanding GET, POST, PUT, DELETE
- Creating routes in Express
- Sending responses (`res.send()`, `res.json()`)

## Module 3: Middleware

- What is middleware
- Using built-in middleware (`express.json()`, `express.urlencoded()`)
- Third-party middleware (like `cors`, `morgan`)

## Module 4: Handling Data

- Reading data from request body
- Query parameters and URL parameters
- Simple in-memory data storage (array/object)

## Module 5: Working with Files / Static Content

- Serving HTML, CSS, JS files
- Serving images and assets

## Module 6: Connecting React & Express

- Making API calls from React to Express
- Fetching data and displaying it in React
- Handling POST requests from forms

## Module 7: Mini Project

- Example: To-Do List API

- Features:
  - React frontend
  - Express backend
  - Add, view, delete items via API