# ReactJS Training - Part 1

## React Components

A component is a reusable, independent block of code that controls part of the UI.

Types of Components:

- Functional Component (Modern)

- Class Component (Legacy)

## Creating an Application Using Create React App

Steps:

1. Install Node.js from https://nodejs.org

2. Run in terminal:

npx create-react-app my-first-app

cd my-first-app

npm start

This will create a React boilerplate and start it at http://localhost:3000

## Hello World Program

In src/App.js:

import React from 'react';

function App() {

return (

<div>

<h1>Hello World from React!</h1>

</div>

);

}

export default App;

## Folder Structure

my-first-app/

├── node\_modules/ // All npm packages

├── public/ // Static files (favicon, index.html)

├── src/ // Your actual React code

│ ├── App.js // Main component

│ ├── index.js // Entry point, renders <App />

│ └── ...

├── package.json // Project config + dependencies

## Components

Functional Component:

function Welcome() {

return <h2>Welcome Functional Component</h2>;

}

Arrow function:

const Welcome = () => <h2>Welcome Functional Component</h2>;

Class Component:

import React, { Component } from 'react';

class WelcomeClass extends Component {

render() {

return <h2>Welcome Class Component</h2>;

}

}

## Understanding JSX

JSX = JavaScript XML

Allows HTML in JS and uses expressions inside {}

Example:

const name = "Manav";

return <h2>Hello, {name}</h2>;

JSX Rules:

- One parent element (use <div> or <> fragment)

- Use className instead of class

- Close all tags