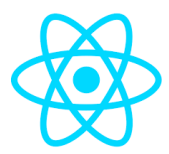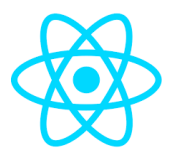# React Training
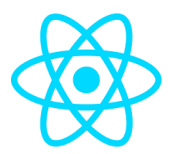
Presented By

Bibhuranjan Mohanty

SESSION-01

Objectives
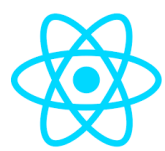
👉 **Introduction to React**

👉 **Features of React**

👉 **Set Up a React Development Environment**

👉 **Build and Run a Basic React App**

React is used for building most of the user interfaces today
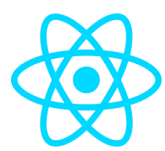
Introduction to React

What is React?

At this point, let's cite the React documentation that has summarized it nicely:

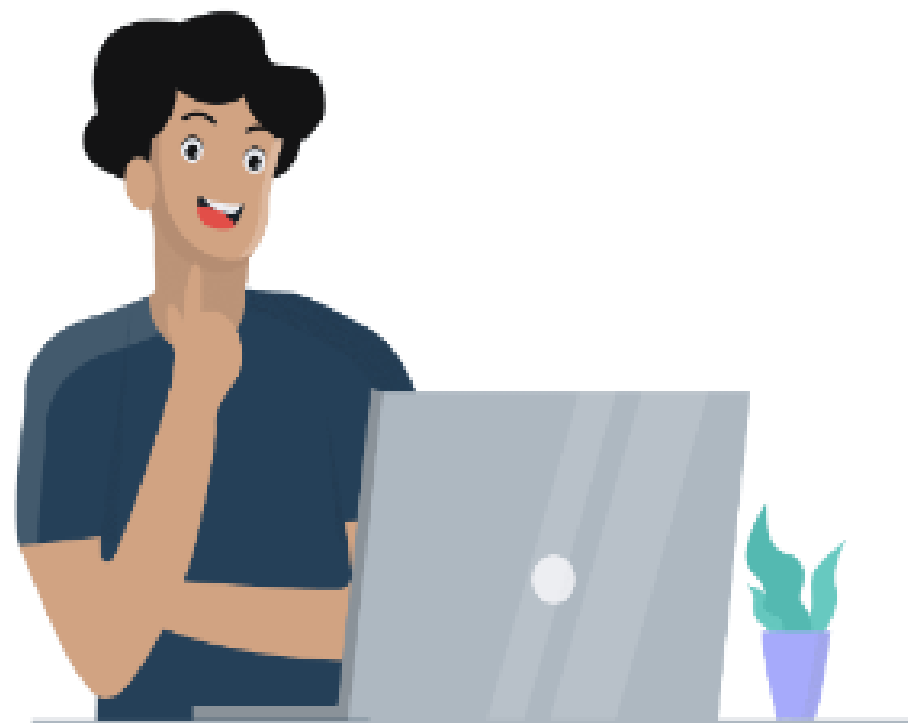> "React is a library for building user interfaces."

👉 React is just a library that makes it easy to develop user interfaces.

👉 There are no services or methods to make API calls ,built-in models, or ORM.

👉 React is only concerned with user interfaces.

When Should I Use React and When Is It Best to Avoid It?

When to Use React: ✓

👉 Building Interactive User Interfaces

👉 Component-Based Development

👉 Handling Complex State Management

👉 Large-Scale Applications

👉 Cross-Platform Development

When to Avoid React: ✗

Should we develop everything in React now?

👉 Simple Static Websites

👉 SEO-Heavy Applications

👉 Limited JavaScript Expertise

👉 Simple Static Websites

👉 Performance-Sensitive Applications with Minimal JavaScript

Key Features of React

👉 Component-Based Architecture

👉 Virtual DOM

👉 Declarative UI

👉 Unidirectional Data Flow
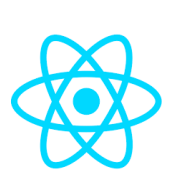
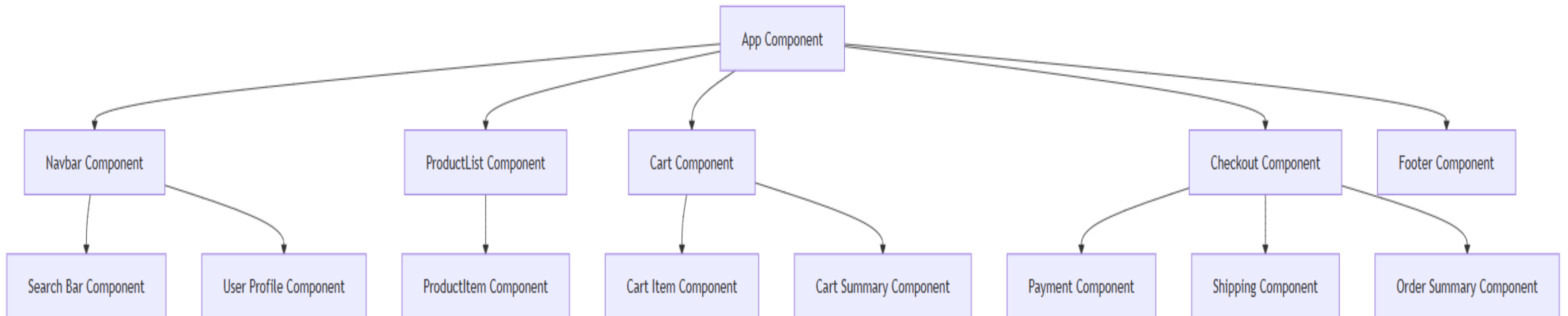👉 JSX - JavaScript XML

👉 Rich Ecosystem

👉 Fast Learning Curve

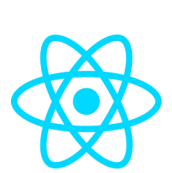👉 Community Support

Key Features of React

👉 **Component-Based Architecture**

Key Features of React

👉 **JSX - JavaScript XML**


```
<div id="app">
<p>A paragraph in JSX</p>
<p>Another paragraph</p>
</div>
```
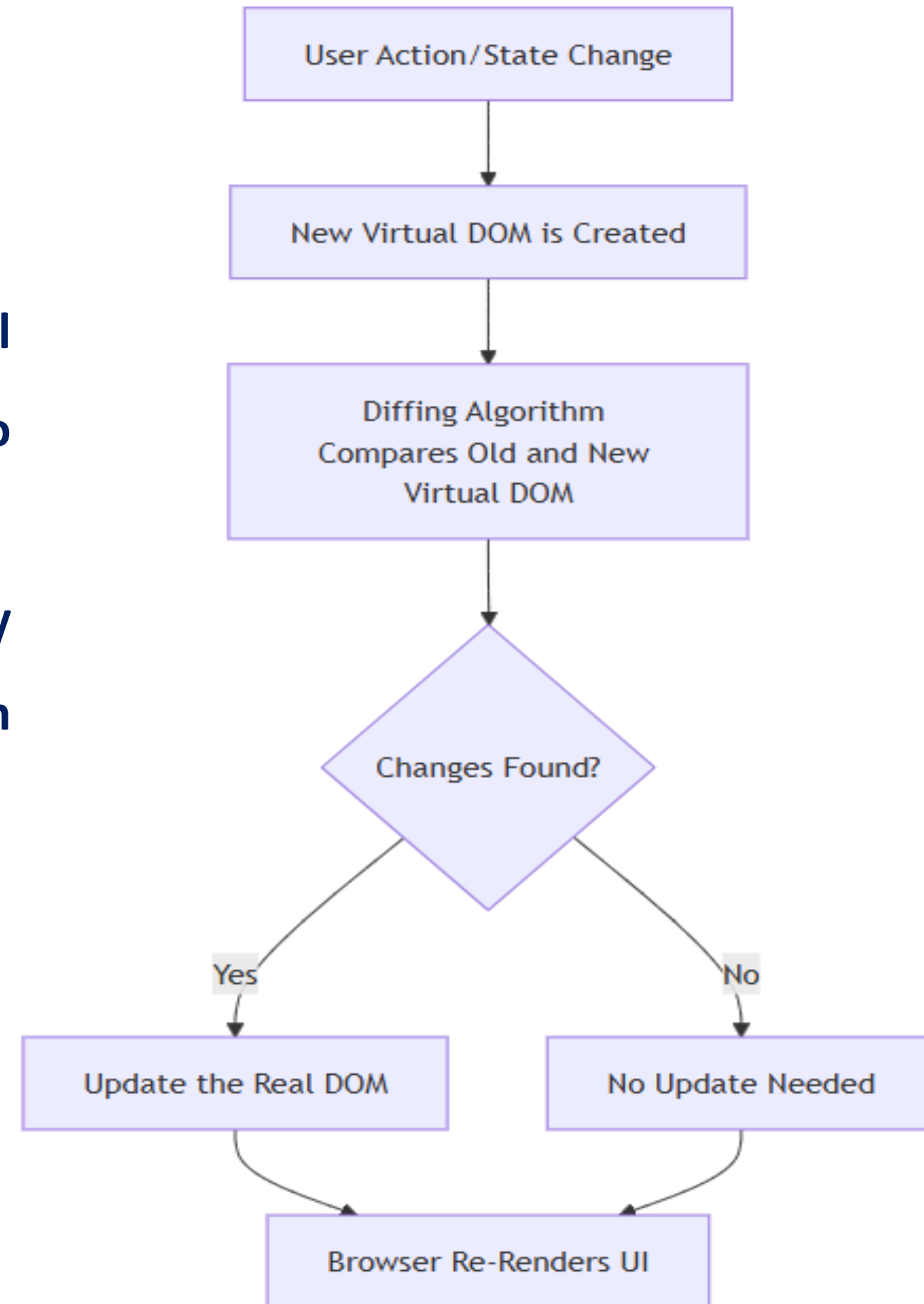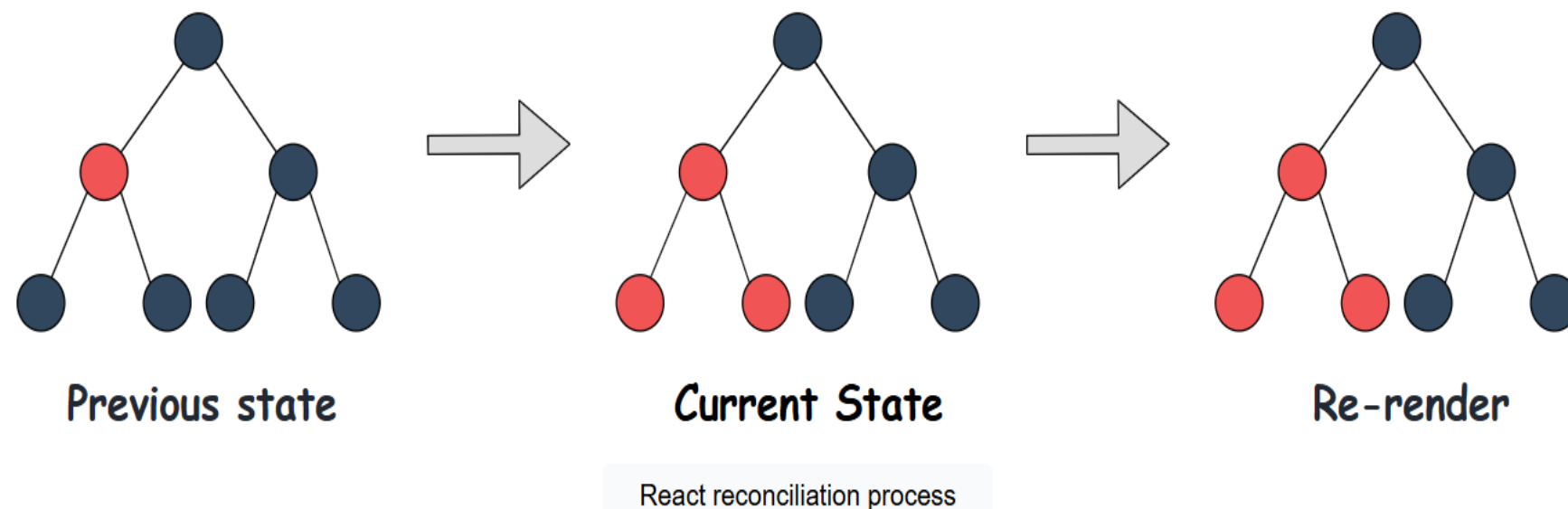
⬇ **Babel Compiler**

```
React.createElement(
'div', { id: 'app' },
React.createElement('p', null, 'A paragraph in JSX'),
React.createElement('p', null, 'Another paragraph')
);
```

Key Features of React

👉 **Virtual DOM**

❑ **The Virtual DOM is a lightweight copy of the actual DOM (Document Object Model) that React uses to optimize updates and rendering.**

❑ **It improves the performance of React applications by reducing direct manipulations of the real DOM, which can be slow.**

Previous state   Current State   Re-render

React reconciliation process

User Action/State Change

New Virtual DOM is Created

Diffing Algorithm Compares Old and New Virtual DOM

Changes Found?

Yes → Update the Real DOM

No → No Update Needed

Browser Re-Renders UI
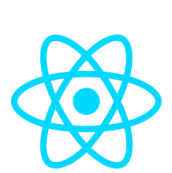
Key Features of React

👉 **Rich Ecosystem**

JSX Syntax

React Hooks

React Core Library

SSR

Next.js

Navigation

React Router

DevTools

React Developer Tool

ESLint

Prettier

Build Tools

Webpack

Babel

Vite

Testing

Jest

Cypress

React Testing Library

Data Fetching

GraphQL

Axios

SWR

Styling

Styled Components

Tailwind CSS

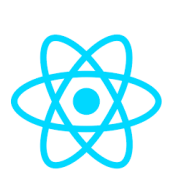CSS Modules

State Management

Redux

ContextAPI

Recoil

Advanced

In-Built

Modern

State

Key Features of React
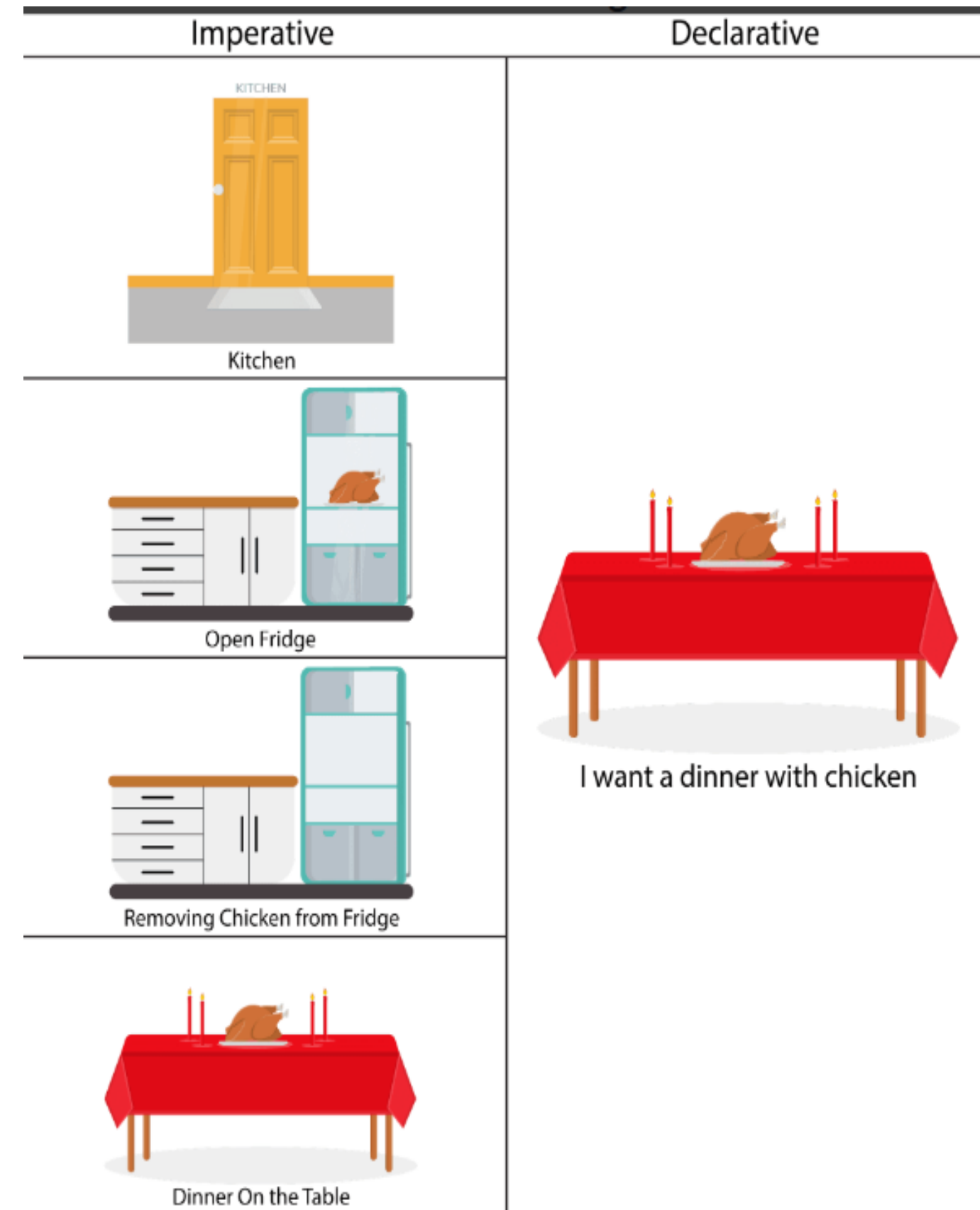
👉 **Declarative UI**

List = [1, 2, 3, 4, 5]

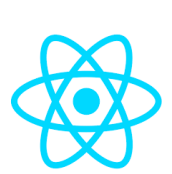Requirement: Access its last element.

Imperative

- Loop through this list: [1, 2, 3, 4, 5]
- Access the element at index list.length - 1
- Return the element

Declarative

- I have this list: [1, 2, 3, 4, 5]
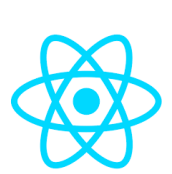- I want the last element of this list

Key Features of React

👉 **Fast Learning Curve**

❑Simple and Intuitive Syntax

❑Extensive Documentation

❑Component-Based Structure
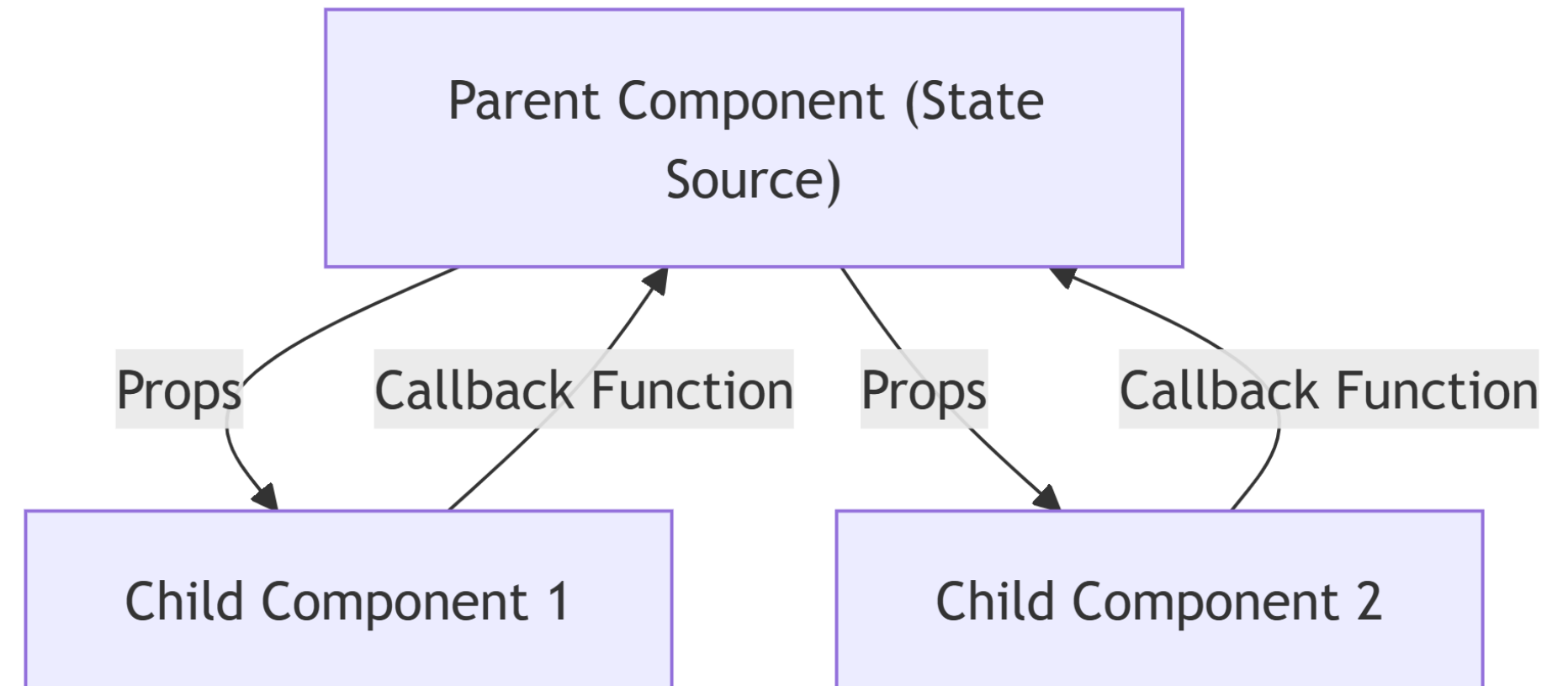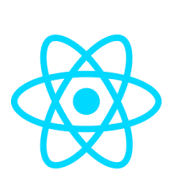
❑Rich Developer Tools

Key Features of React

👉 Unidirectional Data Flow

❑ In React, data flows in a single direction: from parent components to child components.

❑ The parent component manages the state and passes data to child components via props.

❑ This flow simplifies data management and helps prevent unexpected changes in the application state.
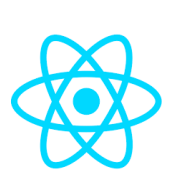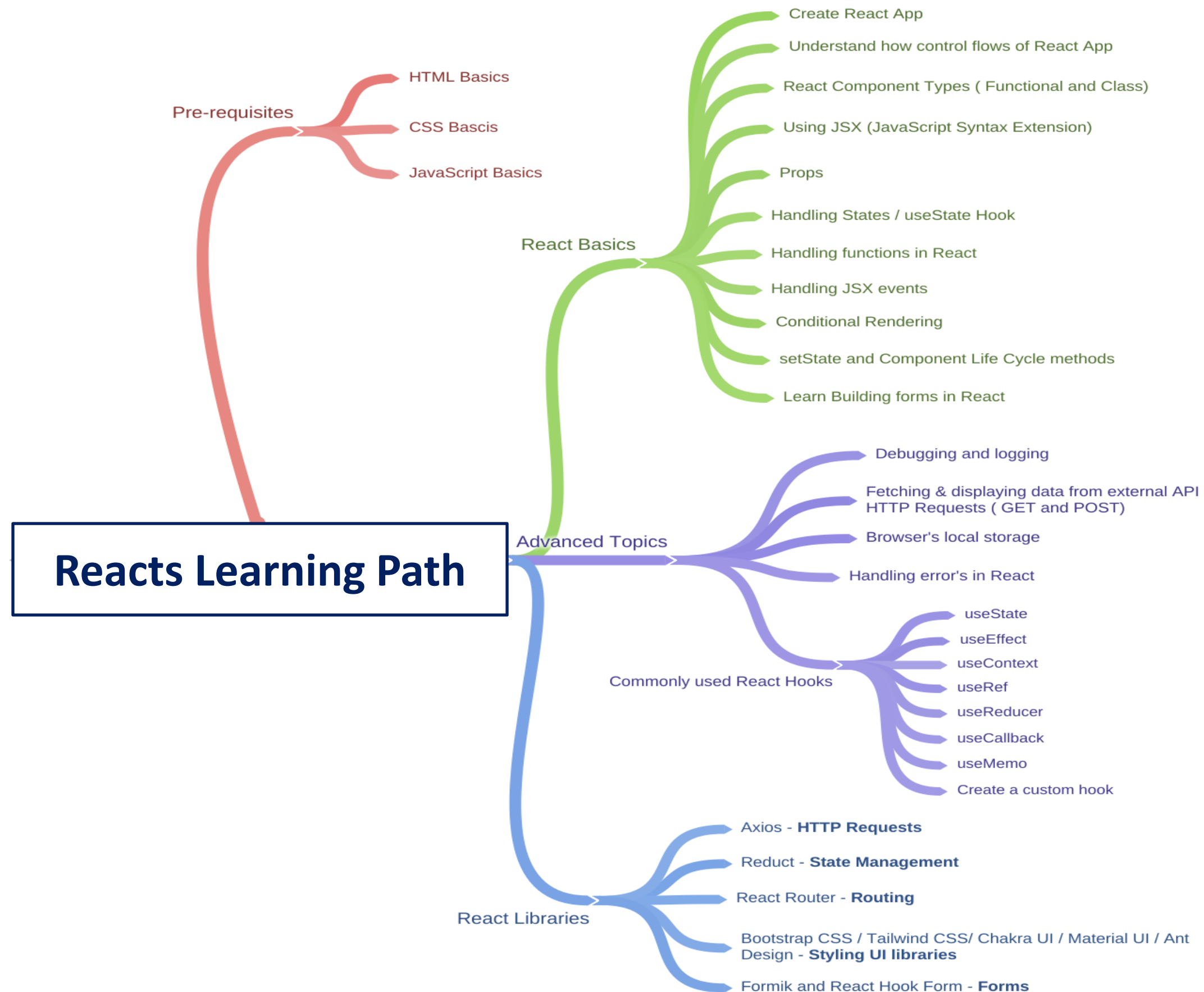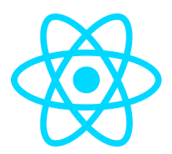
Key Features of React

Community Support

1. **Official Documentation**

2. **GitHub**

3. **Stack Overflow**

4. **Reddit**

5. **Reactiflux Discord Community**

6. **Meetups & Conferences**

Learning Path

Reacts Learning Path

Pre-requisites
- HTML Basics
- CSS Bascis
- JavaScript Basics

React Basics
- Create React App
- Understand how control flows of React App
- React Component Types (Functional and Class)
- Using JSX (JavaScript Syntax Extension)
- Props
- Handling States / useState Hook
- Handling functions in React
- Handling JSX events
- Conditional Rendering
- setState and Component Life Cycle methods
- Learn Building forms in React

Advanced Topics
- Debugging and logging
- Fetching & displaying data from external API HTTP Requests (GET and POST)
- Browser's local storage
- Handling error's in React

Commonly used React Hooks
- useState
- useEffect
- useContext
- useRef
- useReducer
- useCallback
- useMemo
- Create a custom hook

React Libraries
- Axios - **HTTP Requests**
- Reduct - **State Management**
- React Router - **Routing**
- Bootstrap CSS / Tailwind CSS/ Chakra UI / Material UI / Ant Design - **Styling UI libraries**
- Formik and React Hook Form - **Forms**

Setting Up the Development Environment

Tools and Setup

1. Node.js and npm

2. Yarn

3. Babel

4. Webpack

5. ESLint

6. Prettier
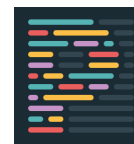
7. Editor
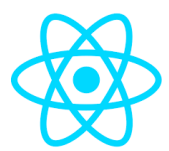
npm vs yarn

Feature	npm	Yarn
Installation Command	`npm install`	`yarn install`
Lock File	`package-lock.json`	`yarn.lock`
Performance	Slower (earlier versions), faster in npm v7+	Generally faster, parallel installation
Security	Audit command (`npm audit`)	Built-in checks, selective resolutions
Workspaces Support	Supported (npm v7+)	Built-in, better support
Offline Cache	Partial (with cache directory)	Full offline cache
Installation Method	Node.js installation (`npm i npm@latest -g`)	`npm i yarn -g` or via corepack
Deterministic Installs	With lock file (npm v5+)	Always deterministic
Project Setup Command	`npm init`	`yarn init`
Script Execution	`npm run <script>`	`yarn <script>`
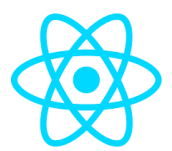
npm vs yarn

| Task	npm Command	Yarn Command
Initialize Project	`npm init`	`yarn init`
Install Dependencies	`npm install` or `npm i`	`yarn`
Add a Package	`npm install <package>`	`yarn add <package>`
Remove a Package	`npm uninstall <package>`	`yarn remove <package>`
Update a Package	`npm update <package>`	`yarn upgrade <package>`
Install Dev Dependency	`npm install <package> -D`	`yarn add <package> -D`
Run a Script	`npm run <script>`	`yarn <script>`
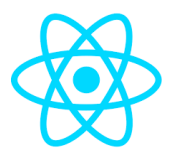
Different Ways to Create React App

1. Using Create React App (CRA)

❑ Recommended for beginners and quick prototypes.

❑ It is the simplest way to start a new React project without manual configuration.

❑ **Pros:** Easy setup, pre-configured with best practices, includes build scripts, hot reloading, and linting.

❑**Command**

```
npx create-react-app my-app
cd my-app
npm start
```

❑Create React App Documentation
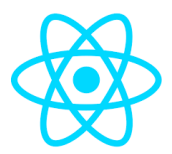
Different Ways to Create React App

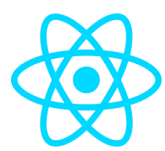## 2. Using Vite

❑ For faster builds and a more modern development experience.

❑ Vite is a build tool that is much faster than CRA, providing a more efficient developer experience.

❑ **Pros:** Fast startup, lightweight, optimized for development with instant hot module replacement.

❑ **Command**

```
npm create vite@latest my-app --template react

cd my-app

npm install

npm run dev
```

❑Vite Documentation

| Method	Use Case	Pros	Example Command
Create React App (CRA)	Beginners, quick start	Easy setup, no config needed	`npx create-react-app my-app`
Vite	Fast development	Fast builds, HMR	`npm create vite@latest my-app --template react`
Next.js	Full-stack, SSR, SEO-friendly	Server-side rendering, routing	`npx create-next-app my-app`
Remix	Enhanced user experience	Efficient data handling	`npx create-remix@latest`
Parcel	Zero-config setup	Fast and simple	`npx parcel index.html`
Manual Setup	Full control, custom config	Customizable	N/A (custom steps)
Expo	Mobile app development	Cross-platform support	`npx create-expo-app my-app`
Online Editors	Quick testing, sharing	No local setup required	N/A (web-based)
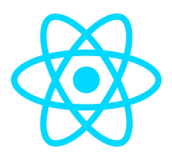
Let's Check Our Understanding So far !
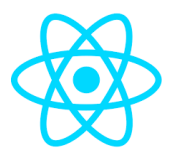
Quiz

1. React is mainly used for building _____.

A) Database

B) Connectivity
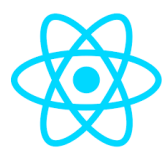
C) User interfaces

D) Design platform
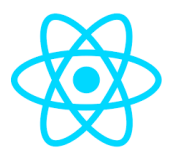
Quiz

2. Which of the following is used to handle data in React ?

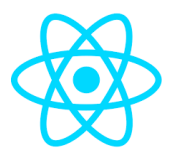**A)**Component

B)State

C)Model

D)None of the above

Quiz

3. Which command is used to install create-react-app?

A) npm install -g create-react-app
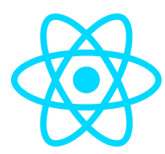
B) npm create-react-app

C) install -g create-react-app

D) None of the above

Quiz

4. Everything in React is a _____.

A) Module
B) Package
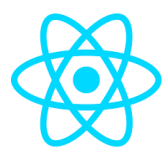**C)** Class
D) Component

Quiz

5. The _____ is an officially supported way to create a React application.

A) create

B) react-app

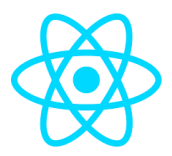**C) create-react-app**

D) create react app

Quiz

6. What function allows you to render React content on an HTML page?

A) ReactDOM.render()

B) React.render()
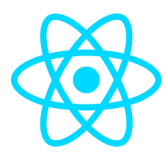
C) React.mount()

**D) ReactDOM.start()**

Quiz

7. What is Reconciliation?

A) It is a process to set the state

B) The process through which React deletes the DOM

C) The process through which React updates the DOM

D) None of the above

Quiz

8. JSX allows us to write HTML elements in JavaScript and place them in the DOM without using the _____ method(s).
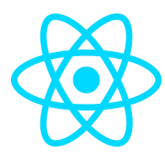
A) createElement

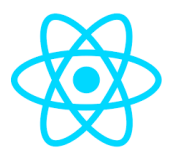B) getElementById

C) All of the above

D) None of the above
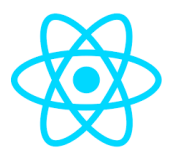
Quiz

9. What command copies build dependencies, configuration files, and scripts into the app directory?

A) yarn build

B) yarn eject

C) yarn start

D) yarn copy

Q & A

Thank You