ONE PAGE SUMMARY

IT & Software › Frontend Development › React

Introduction to React

Unlock the power of dynamic UIs with React!

DEFINITION

> React is a JavaScript library for building user interfaces, especially single-page applications. It allows developers to create reusable UI components that manage their own state.

KEY CONCEPTS

Components

Components are the building blocks of a React application, encapsulating UI and behavior.

JSX

JSX is a syntax extension that allows mixing HTML with JavaScript, making code more readable.

State

State is a built-in object that allows components to manage dynamic data and re-render when it changes.

Props

Props are inputs to components, allowing data to be passed from parent to child components.

Lifecycle Methods

Lifecycle methods are hooks that allow you to run code at specific points in a component's life.

EXAMPLES

→ Creating a button component that changes color on click.

→ Building a form that submits user input and displays a confirmation message.

→ Rendering a list of items from an array using map function.

MEMORY TIPS

★ Remember 'CJSPL' for Components, JSX, State, Props, Lifecycle methods.

★ Think of components as Lego blocks: reusable and customizable.

★ Visualize state as a container that holds data, changing over time.

COMMON MISTAKES

✗ Forgetting to use 'key' prop in lists, leading to rendering issues.

✗ Mutating state directly instead of using setState.

✗ Not properly managing component lifecycle, causing memory leaks.

QUICK RECAP

React is essential for building interactive UIs with reusable components. Focus on understanding state and props for effective data management. Remember the lifecycle methods to optimize component performance.