

React Lecture - 5

Props and Conditional Rendering

More on Props

Pass JSX to a Component as Props

It is common to nest built-in browser tags:

Sometimes you'll want to nest your own components the same way:

```
<div>  
  <img />  
</div>
```

```
<Card>  
  <Avatar />  
</Card>
```

When you nest content inside a JSX tag, the parent component will receive that content in a prop called children.

Example

```
function Card({ children, title }) {  
  return (  
    <div className="card">  
      <h2>{title}</h2>  
      <div>{children}</div>  
    </div>  
  );  
}  
  
// Usage  
<Card title="My Content">  
  <p>This is the content inside the card.</p>  
</Card>
```

Pass JSX to a Component as Props

You can think of a component with a `children` prop as having a “hole” that can be “filled in” by its parent components with arbitrary JSX. You will often use the `children` prop for visual wrappers: panels, grids, etc.

Props

Props are read-only snapshots in time: every render receives a new version of props.

You can't change props. When you need interactivity, you'll need to set state.

Recap

- To pass props, add them to the JSX, just like you would with HTML attributes.
- To read props, use the function `Navgurukul({ person, size })` destructuring syntax.
- You can specify a default value like `size = 100`, which is used for missing and undefined props.
- You can forward all props with `<Avatar {...props} />` JSX spread syntax, but don't overuse it!
- Nested JSX like `<Card><Avatar /></Card>` will appear as Card component's children prop.
- Props are read-only snapshots in time: every render receives a new version of props.
- You can't change props. When you need interactivity, you'll need to set state.

Conditional Rendering

Conditional Rendering

Your components will often need to display different things depending on different conditions. In React, you can conditionally render JSX using JavaScript syntax like if statements, `&&`, and `? :` operators.

Conditionally returning a JSX

We can use `if else`, `&&` and `?:` to conditionally return JSX from a component to render on the page.

If you don't want to render anything you can return `null` from the component and it will not render anything on the page.