

React - Conditional

Topics : [React JS](#)

Written on [January 02, 2024](#)

Conditional rendering is a fundamental concept in React that allows you to render different content based on certain conditions. You can use JavaScript expressions and statements within your JSX to conditionally include or exclude elements. Here are some common patterns for conditional rendering in React:

1. If Statements:

You can use regular JavaScript `if` statements to conditionally render content.

```
import React from 'react';

const ConditionalComponent = ({ condition }) => {
  if (condition) {
    return <p>Content when condition is true.</p>;
  } else {
    return <p>Content when condition is false.</p>;
  }
};

export default ConditionalComponent;
```

2. Ternary Operator:

The ternary operator (`? :`) provides a concise way to express simple conditional statements.

```
import React from 'react';

const ConditionalComponent = ({ condition }) => (
  <div>
    {condition ? <p>Content when condition is true.</p> : <p>Content when condition is false.</p>}
  </div>
);

export default ConditionalComponent;
```

3. Logical && Operator:

The logical `&&` operator can be used for conditional rendering if you want to render content only when a certain condition is true.

```
import React from 'react';

const ConditionalComponent = ({ condition }) => (
  <div>
    {condition && <p>Content when condition is true.</p>}
  </div>
);

export default ConditionalComponent;
```

4. Conditional Rendering with Functions:

You can encapsulate your conditional logic in functions for cleaner code.

```
import React from 'react';

const RenderContent = ({ condition }) => {
  if (condition) {
    return <p>Content when condition is true.</p>;
  } else {
    return <p>Content when condition is false.</p>;
  }
};

const ConditionalComponent = ({ condition }) => (
  <div>
    <RenderContent condition={condition} />
  </div>
);

export default ConditionalComponent;
```

5. Switch Statements:

For more complex conditions, you might use a switch statement.

```
import React from 'react';

const ConditionalComponent = ({ option }) => {
  switch (option) {
    case 'A':
      return <p>Content for option A.</p>;
    case 'B':
      return <p>Content for option B.</p>;
    default:
      return <p>Default content.</p>;
  }
};

export default ConditionalComponent;
```

6. Using Variables:

You can use variables or functions to store the content you want to render conditionally.

```
import React from 'react';

const ContentA = () => <p>Content for option A.</p>;
const ContentB = () => <p>Content for option B.</p>;

const ConditionalComponent = ({ option }) => {
  let contentComponent;

  if (option === 'A') {
    contentComponent = <ContentA />;
  } else if (option === 'B') {
    contentComponent = <ContentB />;
  } else {
    contentComponent = <p>Default content.</p>;
  }

  return <div>{contentComponent}</div>;
};

export default ConditionalComponent;
```

© Copyright **Aryatechno**. All Rights Reserved. Written tutorials and materials by [Aryatechno](#)