

I am passing props data from parent to child components in react

From my parent component

```
<Hero title="Test Title" subtitle="This is the subtitle" />
```

How to receive the props data in child component?



To receive props data in a child component in React, you need to define the child component in such a way that it accepts the props passed down from the parent. Here's how you can do it:

Parent Component

In your parent component, you pass the props to the child component like this:

```
jsx                                     Copy code

import React from 'react';
import Hero from './Hero';

const ParentComponent = () => {
  return (
    <div>
      <Hero title="Test Title" subtitle="This is the subtitle" />
    </div>
  );
};

export default ParentComponent;
```

Child Component

In the child component, you need to declare the props as parameters in the component function. You can then use these props within the component. Here is how you can define the 'Hero' component to receive and use the 'title' and 'subtitle' props:

```
jsx                                     Copy code

import React from 'react';

const Hero = ({ title, subtitle }) => {
  return (
    <div>
      <h1>{title}</h1>
      <p>{subtitle}</p>
    </div>
  );
};

export default Hero;
```

Explanation

- **Parent Component:** This component imports the 'Hero' component and passes the 'title' and 'subtitle' as props.
- **Child Component ('Hero'):** This component receives the props by destructuring the props object in the function parameters. It then uses these props to render the title and subtitle.

Complete Example

Here is the complete example with both components together:

```
jsx                                     Copy code

// ParentComponent.js
import React from 'react';
import Hero from './Hero';

const ParentComponent = () => {
  return (
    <div>
      <Hero title="Test Title" subtitle="This is the subtitle" />
    </div>
  );
};

export default ParentComponent;
```

```
jsx                                     Copy code

// Hero.js
import React from 'react';

const Hero = ({ title, subtitle }) => {
  return (
    <div>
      <h1>{title}</h1>
      <p>{subtitle}</p>
    </div>
  );
};

export default Hero;
```

When you run this code, the 'ParentComponent' will render the 'Hero' component, passing the 'title' and 'subtitle' props. The 'Hero' component will receive these props and display them accordingly.

And now suppose I want to send some data from child to parent component, how to do it



To send data from a child component to a parent component in React, you typically use a callback function. The parent component defines a function that can handle the data, and this function is passed as a prop to the child component. The child component then calls this function, passing the data as an argument.

Here's an example to illustrate how this can be done:

Parent Component

1. Define a function in the parent component that will handle the data received from the child.
2. Pass this function as a prop to the child component.

```
jsx Copy code

import React, { useState } from 'react';
import ChildComponent from './ChildComponent';

const ParentComponent = () => {
  const [childData, setChildData] = useState('');

  const handleChildData = (data) => {
    setChildData(data);
    console.log("Data received from child:", data);
  };

  return (
    <div>
      <h1>Parent Component</h1>
      <p>Data from child: {childData}</p>
      <ChildComponent onData={handleChildData} />
    </div>
  );
};

export default ParentComponent;
```

Child Component

1. Accept the function as a prop.
2. Call this function when you want to send data to the parent, passing the data as an argument.

```
jsx Copy code

import React, { useState } from 'react';

const ChildComponent = ({ onData }) => {
  const [inputValue, setInputValue] = useState('');

  const handleChange = (e) => {
    setInputValue(e.target.value);
  };

  const handleSubmit = () => {
    onData(inputValue);
  };

  return (
    <div>
      <h2>Child Component</h2>
      <input type="text" value={inputValue} onChange={handleChange} />
      <button onClick={handleSubmit}>Send to Parent</button>
    </div>
  );
};

export default ChildComponent;
```

Explanation

- **Parent Component:**
 - `handleChildData` is the function that will handle the data received from the child component. It sets the received data to the state `childData` and logs it.
 - This function is passed to the `ChildComponent` via the `onData` prop.
- **Child Component:**
 - The `onData` function is received as a prop.
 - When the user types in the input field, the `handleChange` function updates the `inputValue` state.
 - When the user clicks the "Send to Parent" button, the `handleSubmit` function calls the `onData` function, passing the `inputValue` as an argument.

Complete Example

Here is the complete example with both components together:

```
jsx Copy code

// ParentComponent.js
import React, { useState } from 'react';
import ChildComponent from './ChildComponent';

const ParentComponent = () => {
  const [childData, setChildData] = useState('');

  const handleChildData = (data) => {
    setChildData(data);
  };

  return (
    <div>
      <h1>Parent Component</h1>
      <p>Data from child: {childData}</p>
      <ChildComponent onData={handleChildData} />
    </div>
  );
};

export default ParentComponent;
```

```
    console.log("Data received from child:", data);
  });

  return (
    <div>
      <h1>Parent Component</h1>
      <p>Data from child: {childData}</p>
      <ChildComponent onData={handleChildData} />
    </div>
  );
};

export default ParentComponent;
```

jsx

Copy code

```
// ChildComponent.js
import React, { useState } from 'react';

const ChildComponent = ({ onData }) => {
  const [inputValue, setInputValue] = useState('');

  const handleChange = (e) => {
    setInputValue(e.target.value);
  };

  const handleSubmit = () => {
    onData(inputValue);
  };

  return (
    <div>
      <h2>Child Component</h2>
      <input type="text" value={inputValue} onChange={handleChange} />
      <button onClick={handleSubmit}>Send to Parent</button>
    </div>
  );
};

export default ChildComponent;
```

This example shows how to pass data from a child component to a parent component in React using a callback function. When you enter text in the child component's input field and click the button, the text is sent to the parent component and displayed there.

📄 ↻ 🗑️ ⚙️

Message ChatGPT

ChatGPT can make mistakes. Check important info.

?