

❓ What Are React Props?

Props (short for **properties**) are how **data is passed from one component to another** in React.

They allow a **parent component** to send **information** to a **child component**, which then uses that information to render something.

Think of Props Like Function Parameters

If a component is a **function**, then **props are like its arguments**.

Think of Props Like Function Parameters

If a component is a **function**, then **props are like its arguments**.

Example:

```
function greet(name) {  
  console.log("Hello, " + name);  
}  
  
greet("Amit"); // Output: Hello, Amit
```

In React, it works the same way:

```
function Greeting(props) {  
  return <h1>Hello, {props.name}</h1>;  
}
```

Now if we call it like this:

```
<Greeting name="Amit" />
```

It will render:

```
<h1>Hello, Amit</h1>
```

❑ Why Are Props Useful?

- ❑ **Reusable components:** You can use the same component with different data.
- ❑ **Data flow:** They allow data to flow from **parent** → **child**.
- ❑ **Customization:** Let you customize components with different values.

Example: A Simple Card Component Using Props

Create React App

1. Run this command:

```
npx create-react-app simple-card
```

2. Go into your project folder:

```
cd simple-card
```

3. Start the development server:

```
npm start
```

Step 1: Create the Card Component

```
// Card.jsx
import React from 'react';

const Card = (props) => {
  return (
    <div style={{ border: "1px solid black", padding: "10px", margin: "10px" }}>
      <h2>{props.title}</h2>
      <p>{props.content}</p>
    </div>
  );
};

export default Card;
```

Step 2: Use the Card Component with Props

```
// App.jsx
import React from 'react';
import Card from './Card';

const App = () => {
  return (
    <div>
      <Card title="React Props" content="Props are used to pass data between components." />
      <Card title="React State" content="State is used to manage data within a component." />
    </div>
  );
};
```

```
export default App;
```

Start the development server:

```
npm start
```

□ What You'll See on Screen:

React Props

Props are used to pass data between components.

React State

State is used to manage data within a component.

🔗 Step 1: Set up your React App — step by step.

I'll give you **two options** to create a React project:

- **Option A:** Using **Vite** (faster and modern — recommended)
 - **Option B:** Using **Create React App** (still widely used)
-

□ Step 1: Set Up Your React App

□ Option A: Using Vite (Recommended)

□ 1. Open your terminal and run:

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

□ **2. Go into your project folder:**

```
cd my-blog-app
```

□ **3. Install dependencies:**

```
npm install
```

□ **4. Start the development server:**

```
npm run dev
```

Or:

```
yarn dev
```

□ Vite will show a local server URL like `http://localhost:5173/` — open it in your browser.

Option B: Using Create React App

□ **1. Run this command:**

```
npx create-react-app my-blog-app
```

□ **2. Go into your project folder:**

```
cd my-blog-app
```

□ **3. Start the development server:**

```
npm start
```

This will open `http://localhost:3000/` in your browser.

2 Step 2: Create some dummy data

We'll use a simple array of blog posts `blogData.js`.

```
// blogData.js
const blog = [
  {
    title: "React Basics",
    content: "Learn about components, props, and state."
  },
  {
    title: "Using useEffect",
    content: "Understand how side effects work in React."
  },
  {
    title: "React Router",
    content: "Learn routing with React Router."
  }
];

export default blog;
```

2 Step 3: Create the `ProductItems.jsx` Component

This component will receive the blog item as props and display it.

```
// ProductItems.jsx

import React from 'react';

const ProductItems = ({ pitems }) => {
  return (
    <div style={{ border: '1px solid gray', padding: '10px', marginBottom: '10px' }}>
      <h2>{pitems.title}</h2>
      <p>{pitems.content}</p>
    </div>
  );
};

export default ProductItems;
```

Step 4: Use `.map()` in the Parent Component(`App.jsx`)

This is where the main logic from your question happens.

```
// App.jsx
import React from 'react';
import blog from './blogData'; // Importing the blog array
import ProductItems from './ProductItems'; // Importing the child component

const App = () => {
  return (
    <div style={{ width: '60%', margin: '0 auto' }}>
      <h1>Blog Posts</h1>

      {/* Here's the map function you're teaching */}
      {blog.map((v, i) => {
        return (
          <ProductItems pitems={v} key={i} />
        );
      })}
    </div>
  );
};

export default App;
```

- `blog.map((v, i) => { ... })` loops through each blog item.
- `v` is the current blog object.
- `i` is the index (used as `key` — unique identifier).
- Each `v` is passed as `pitems` prop to `<ProductItems />`.