

useContext

Children

Another way of getting input in component is children
if you get you input in middle of you component like this

```
<Cart>3</Cart>
```

you can get number 3 in porps like this:

```
const Cart = (props) => {  
  return <h1>{props.children + 3}</h1>;  
};
```

It means all the data you pass in the middle of tags you can have that in porps.children

Context

React Context is a way to manage state globally.
You can pass your state so easy to you nested components.

Global access state: When we want to have access a state in which is not in this component we need to use context

Props drilling: Passing states downward

Create Context

At first you need to create context with this code

```
export const MyContext = React.createContext({});
```

Now you need create MyContext.Provider and wrap your Dom with Provider and all the components in that DOM can use the value you pass to MyContext.Provider

all the code in App

```
import React from "react";
import Cart from "../components/cart";
export const MyContext = React.createContext({});

const App = (props) => {
  const [state, setState] = React.useState({
    name: "Ali"
  });
  return (
    <MyContext.Provider value={state}>
      <Cart>
        <h1>Hello</h1>
      </Cart>
    </MyContext.Provider>
  );
};

export default App;
```

Other Component child of App

```
import React from "react";
import Product from "../Product";
const Cart = (props) => {
  return (
    <>
      <div>
        <Product/>
      </div>
    </>
  );
};

export default Cart;
```

Product Component child of Cart

In this code you can use useContext to get the value

```
import React from "react";
import {MyContext} from "../App";

const Product = (props) => {
  const con = React.useContext(MyContext);
  console.log(con)
  return (
    <div className="product">
      {con.name}
    </div>
  )
}
```

```
export default Product;
```

If you want to move your Context to another file you can do this
create another file in context folder and do this

```
import React, {Context} from "react";

const MyContext = React.createContext();
const ErfanContext = (props) => {
  const [state, setState] = React.useState({
    name: "Wes",
    age: 100,
    cool: true
  });
  return (
    <MyContext.Provider value={state}>
      {props.children}
    </MyContext.Provider>
  )
}

export default ErfanContext;
```

this code this is same first code it means props.children get components and pass to MyContext.Provider

```
<MyContext.Provider value={state}>
  {props.children}
</MyContext.Provider>
```

then your code in app will be:

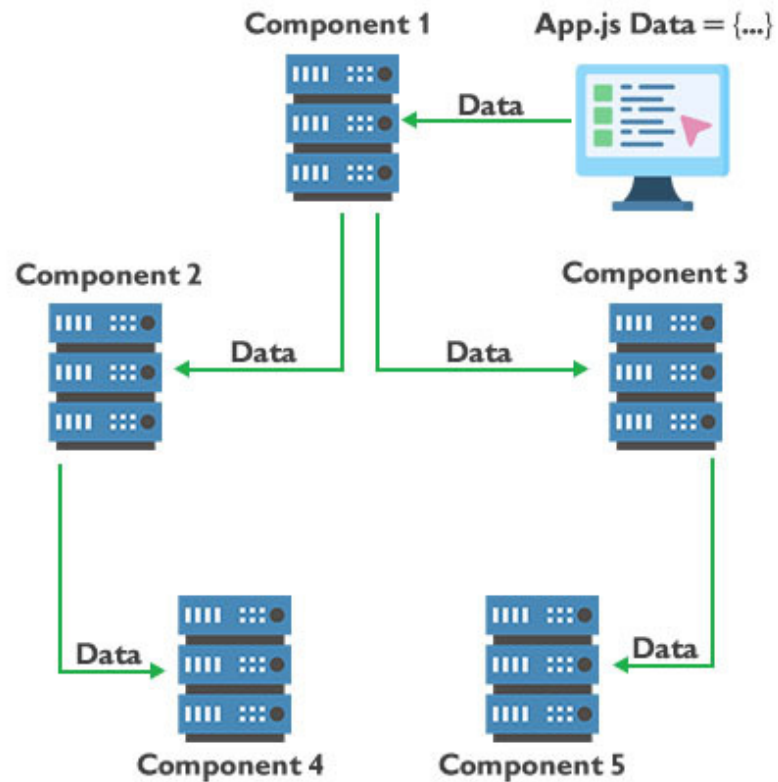
```
import React from "react";
import Cart from "../components/cart";
import { ErfanContext } from "../components/context";

const App = (props) => {

  return (
    <ErfanContext>
      <Cart>
        <h1>Hello</h1>
      </Cart>
    </ErfanContext>
  );
};

export default App;
```

sending data without context



sending data with context

