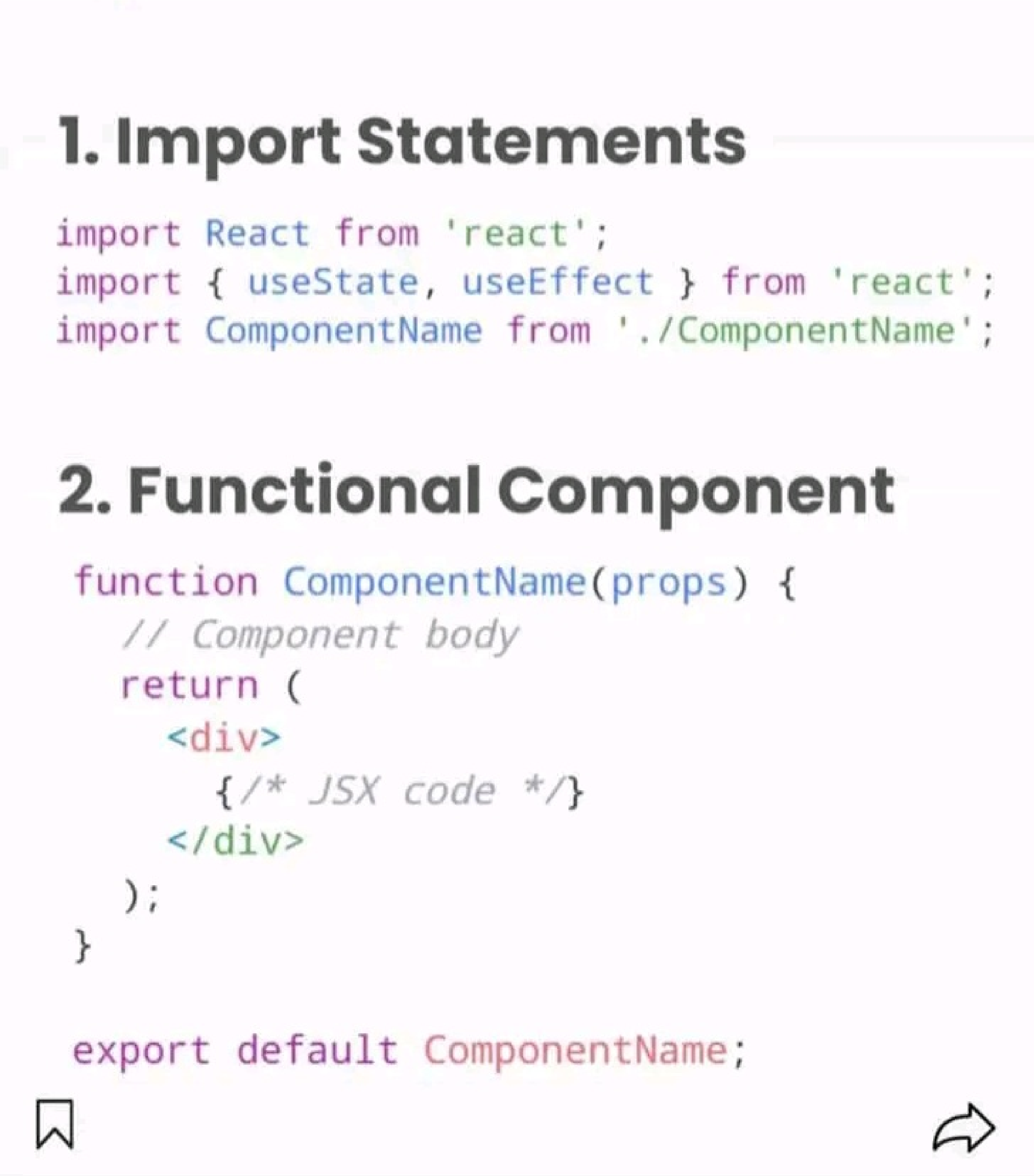
React
Cheat Sheet

Swipe



1. Import Statements

```
import React from 'react';
import { useState, useEffect } from 'react';
import ComponentName from './ComponentName';
```

2. Functional Component

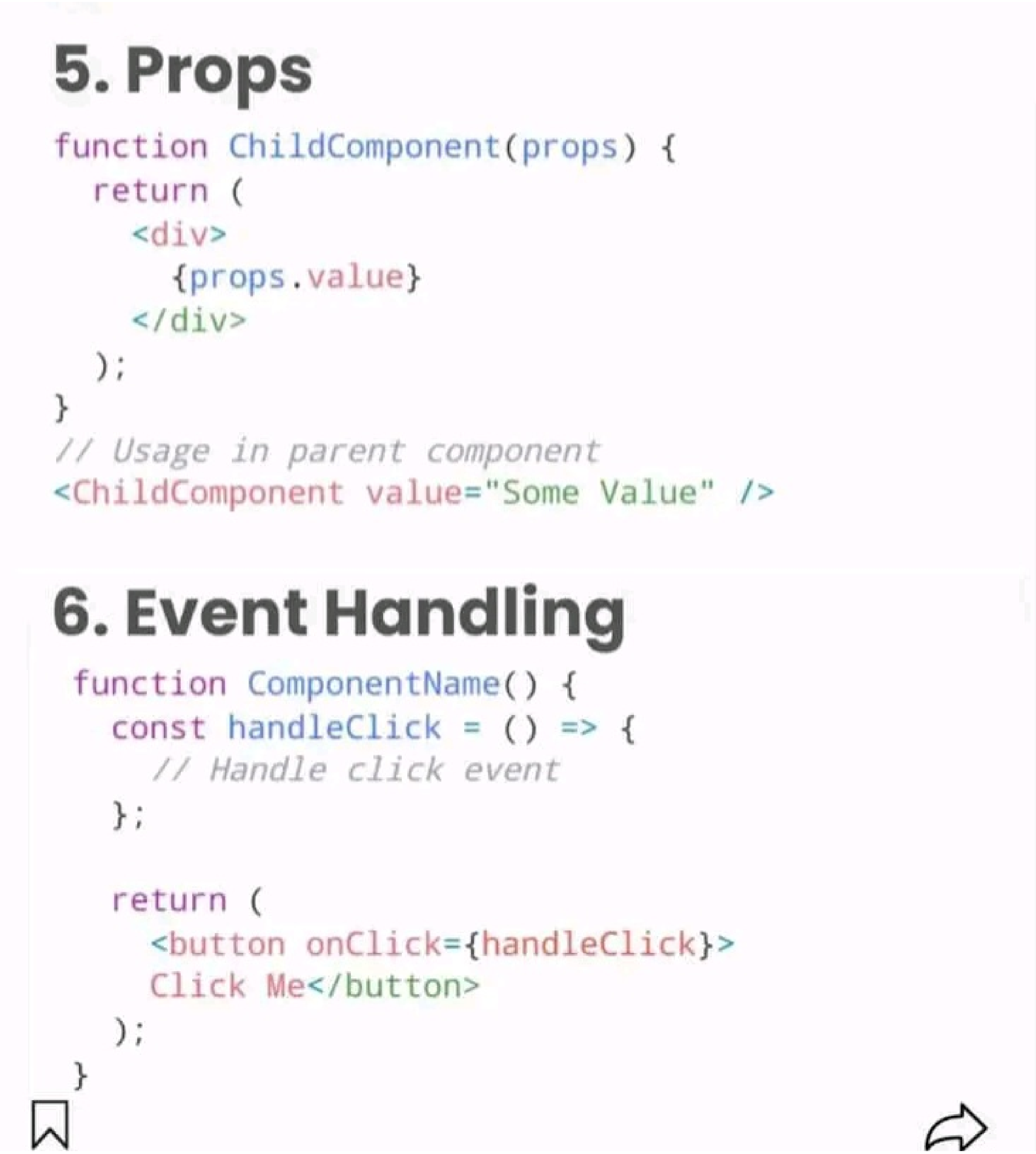
```
function ComponentName(props) {
  // Component body
  return (
    <div>
      {/* JSX code */}
    </div>
  );
}

export default ComponentName;
```

3. Class Component

```
import React, { Component } from 'react';
class ComponentName extends Component {
  render() {
    return (
      <div>
        {/* JSX code */}
      </div>
    );
  }
}
export default ComponentName;
```

4. JSX Syntax

```
 return (
   <div>
     <h1>{title}</h1>
     <p>{description}</p>
   </div>
 );
```

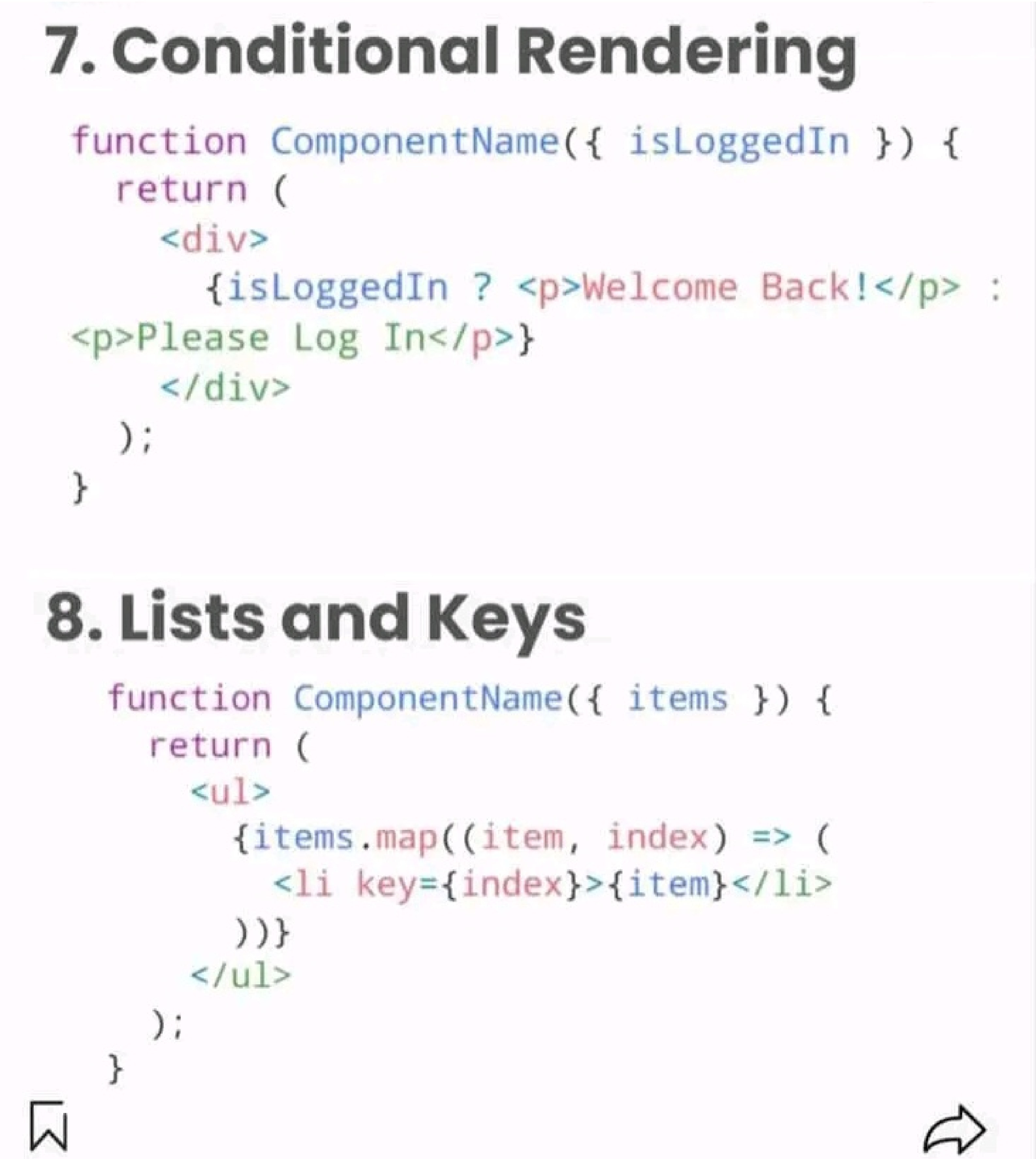


5. Props

```
function ChildComponent(props) {
  return (
    <div>
      {props.value}
    </div>
  );
}
// Usage in parent component
<ChildComponent value="Some Value" />
```

6. Event Handling

```
 function ComponentName() {
   const handleClick = () => {
     // Handle click event
   };

   return (
     <button onClick={handleClick}>
     Click Me</button>
   );
 }
```

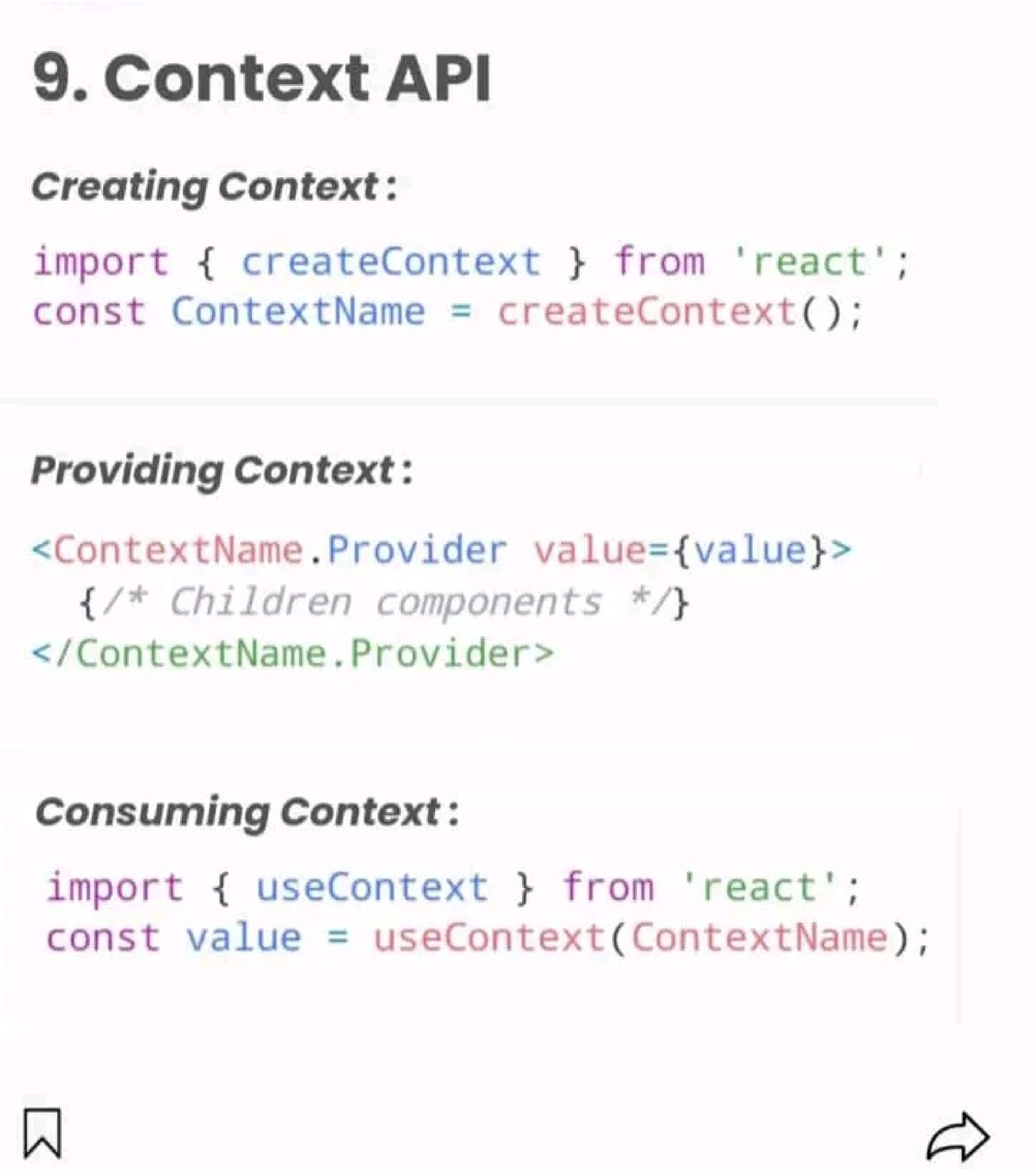


7. Conditional Rendering

```
function ComponentName({ isLoggedIn }) {
  return (
    <div>
      {isLoggedIn ? <p>Welcome Back!</p> :
<p>Please Log In</p>}
    </div>
  );
}
```

8. Lists and Keys

```
function ComponentName({ items }) {
  return (
    <ul>
      {items.map((item, index) => (
        <li key={index}>{item}</li>
      ))}
    </ul>
  );
}
```



9. Context API

Creating Context :

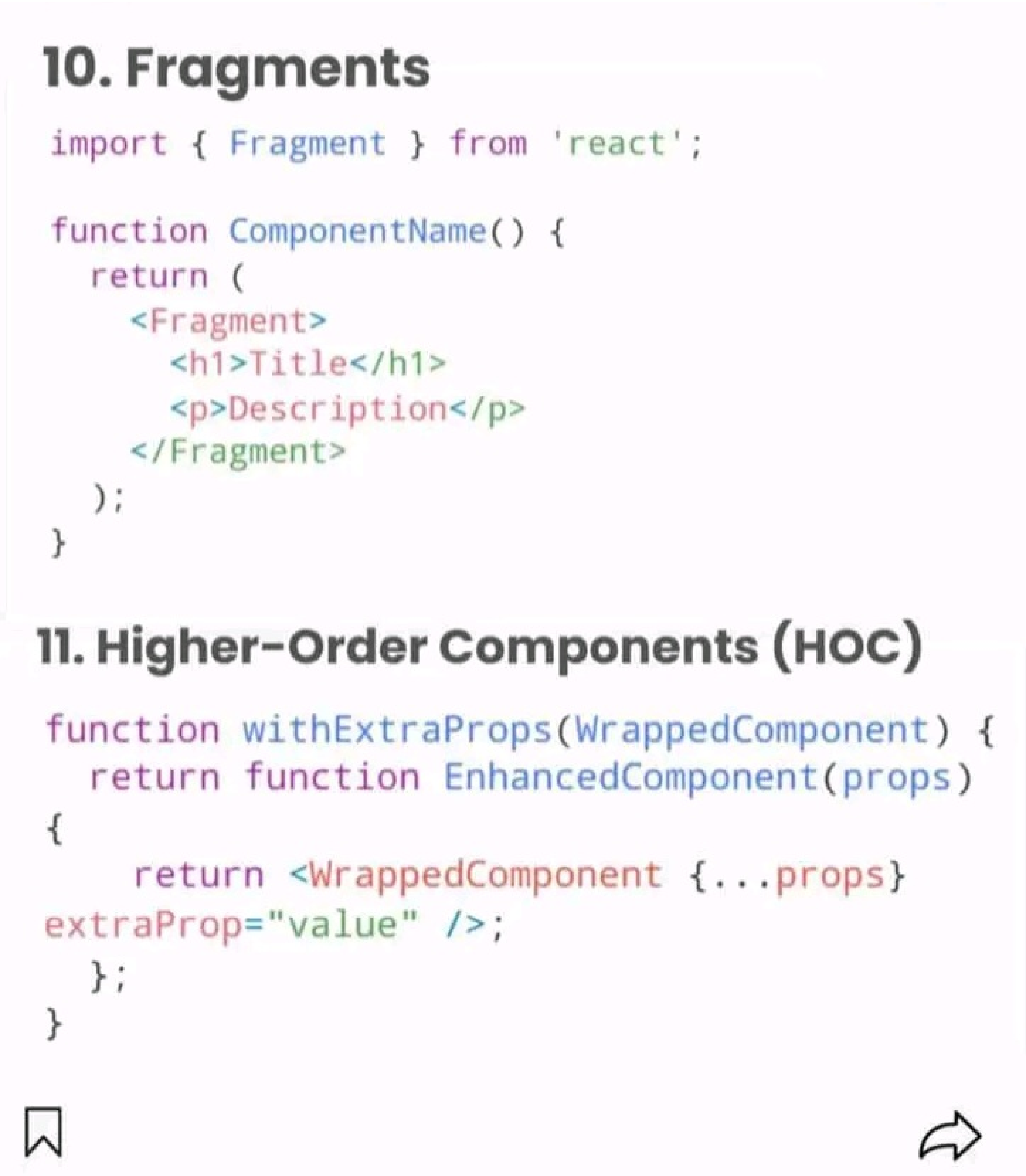
```
import { createContext } from 'react';
const ContextName = createContext();
```

Providing Context :

```
<ContextName.Provider value={value}>
  {/* Children components */}
</ContextName.Provider>
```

Consuming Context :

```
import { useContext } from 'react';
const value = useContext(ContextName);
```

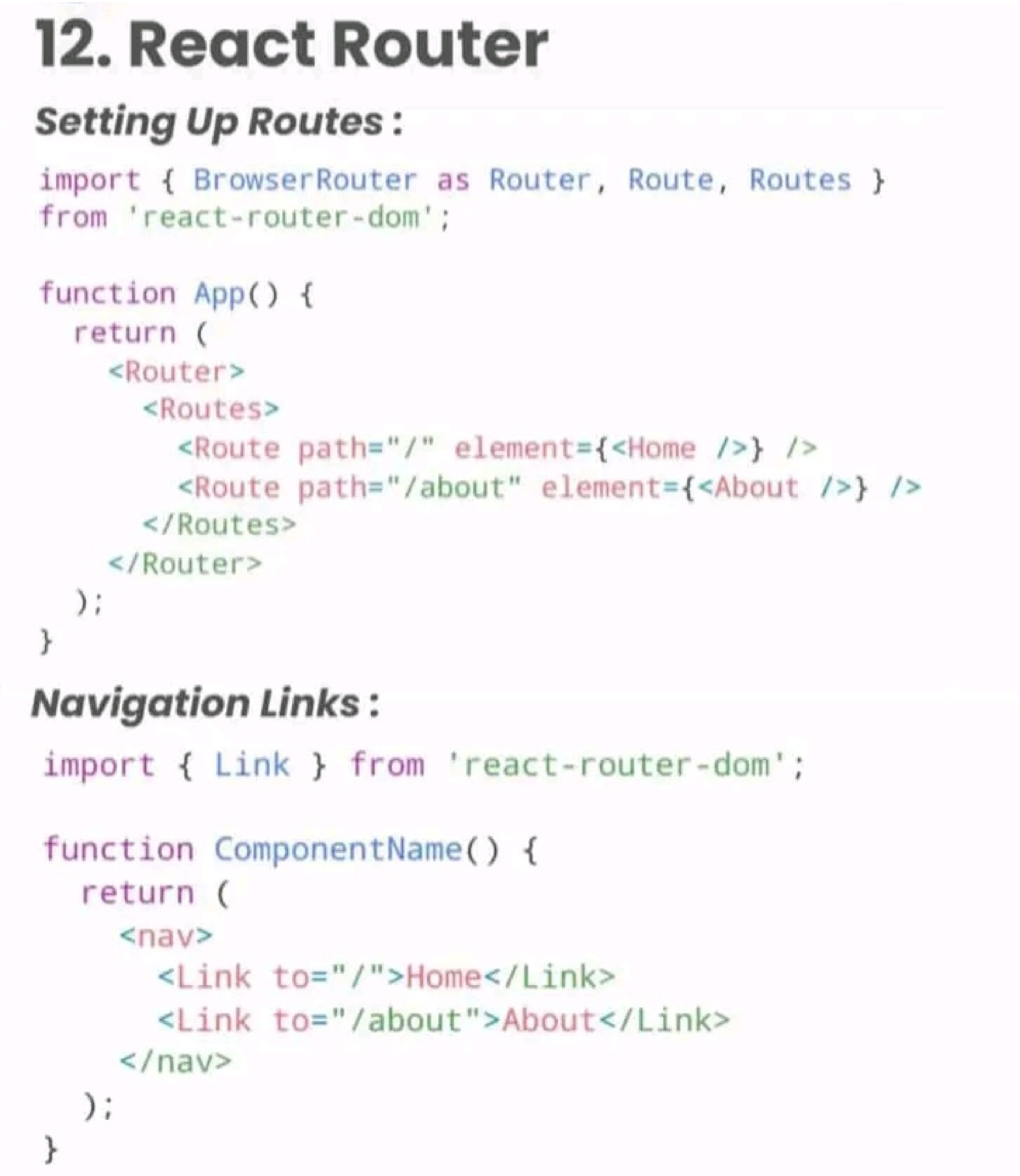


10. Fragments

```
import { Fragment } from 'react';

function ComponentName() {
  return (
    <Fragment>
      <h1>Title</h1>
      <p>Description</p>
    </Fragment>
  );
}
```

11. Higher-Order Components (HOC)

```
function withExtraProps(WrappedComponent) {
  return function EnhancedComponent(props)
{
    return <WrappedComponent {...props}
extraProp="value" />;
  };
}
```



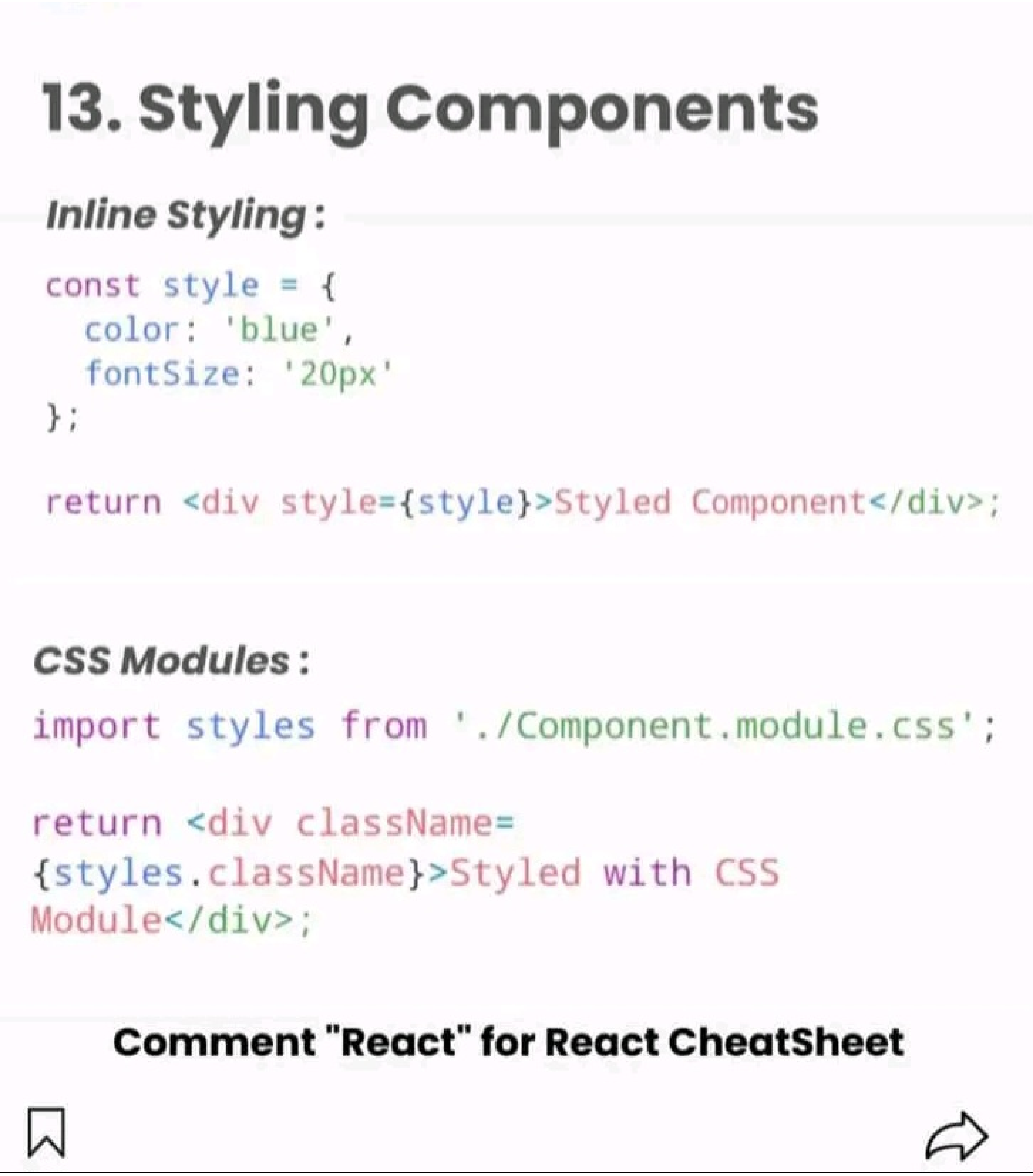
12. React Router

Setting Up Routes :

```
import { BrowserRouter as Router, Route, Routes }
from 'react-router-dom';

function App() {
  return (
    <Router>
      <Routes>
        <Route path="/" element={<Home />} />
        <Route path="/about" element={<About />} />
      </Routes>
    </Router>
  );
}
```

Navigation Links :

```
import { Link } from 'react-router-dom';

function ComponentName() {
  return (
    <nav>
      <Link to="/">Home</Link>
      <Link to="/about">About</Link>
    </nav>
  );
}
```



13. Styling Components

Inline Styling :

```
const style = {
  color: 'blue',
  fontSize: '20px'
};

return <div style={style}>Styled Component</div>;
```

CSS Modules :

```
import styles from './Component.module.css';

return <div className=
{styles.className}>Styled with CSS
Module</div>;
```

Comment "React" for React CheatSheet