

تمرین سری 15 (سوال دوم)

React State Management.1

مدیریت استیت روشی برای ایجاد ارتباط و نقل و انتقال داده است . مدیریت استیت یک ساختار داده ای می سازد که استیت های برنامه ما را نشان میدهد که می توانیم از روی این ساختار داده ای بخوانیم و یا روی آن بنویسیم .به صورت کلی استیت یک شی جاوا اسکریپت است و هر گاه کاربر عملی را در برنامه ریکت انجام میدهد تغییری در استیت ها اتفاق می افتد و اگر برنامه بزرگ شود بررسی تمام وابستگی ها می تواند مشکل ساز باشد .مدیریت استیت یکی از جنبه ها هر برنامه است که نشان میدهد که یک برنامه چگونه دیده شود و یا دیتا ها چگونه ذخیره شوند ریکت کتابخانه هایی دارد که کار مدیریت استیت ها را توسط انها انجام می دهد .

برخی از کتابخانه ها که در ریکت می توانیم استفاده کنیم عبارتند از :

Recoil-Jotai-Redux-Rematch-Zustand-Hookstate

[1.3],[1.2],[1.1]

Functional Components.2

یکی از روش های تعریف کامپوننت ها استفاده از فانکشن های جاوا اسکریپت است که در صورت لزوم و برای انتقال اطلاعات بین چند کامپوننت از **props** ها را به عنوان ورودی به آن ها می دهیم . و در واقع تنها **state** هایی که فانکشن کامپوننت ها به آنها دسترسی دارند هم از همین طریق باید به آنها داده شود .

همانند قبل می توانیم از فانکشن کامپوننت ها (به صورت کامپوننت) هم داخل یکدیگر استفاده کنیم . یکسری از خواص که در کلاس کامپوننت ها وجود دارد به صورت پیش فرض در فانکشن کامپوننت ها وجود نداشت و این ویژگی ها توسط هوک ها به فانکشن کامپوننت ها اضافه شده است. **[2.1],[2.2]**

Hook in react.3

هوک ها به ما اجازه می دهند که از **state** بدون نوشتن کلاس استفاده کنیم . (سابق بر این فانکشن کامپوننت ها **stateless** بودند ولی بعد از هوک ها فانکشن ها هم **Stateful** شدند) هوک ها به خودی خودی نوع خاصی از فانکشن هستند که ویژگی های مربوط به ریخت را به فانکشنال کامپوننت ها اضافه می کنند . هوک ها داخل کلاس ها استفاده نمی شوند

در استفاده از هوک باید دو شرط را رعایت کنیم

1. نباید هوک ها را داخل حلقه ها ، شروط و یا توابع تو در تو استفاده کنیم . همواره باید هوک ها را در بالاترین سطح از تابع ریکت استفاده کنیم . این باعث می شود که هوک ها بر اساس یک ترتیب در هر بار رندر شدن کامپوننت صدا زده شوند .

2. نمی توانیم هوک ها را داخل فانکشن ها معمولی جاوا اسکریپت صدا بزنیم .

3. هوک ها نمی توانند شرطی باشند

State ها در هوک به وسیله `useState` فراهم می شوند که کار مقدار دهی اولیه و یا تغییر آن به وسیله متدی که تعریف میکنیم را فراهم می کنند . به وسیله هوکی به نام `useEffect` می توانیم `lifecycle` های مربوطه را پیاده سازی کنیم . همین طور می توانیم برای یک منطق تکراری یک هوک سفارشی برای خودمان تعریف کنیم [\[3.1\]](#) , [\[3.2\]](#)

منابع

[1.1]<https://blog.openreplay.com/top-6-react-state-management-libraries-for-2022>

[1.2]<https://dev.to/workshub/state-management-battle-in-react-2021-hooks-redux-and-recoil-2am0>

[1.3]https://www.tutorialspoint.com/reactjs/reactjs_state.htm

[2.1]<https://reactjs.org/docs/components-and-props.html>

[2.2]<https://www.geeksforgeeks.org/reactjs-functional-components/>

[3.1]<https://reactjs.org/docs/hooks-overview.html>

[3.2]https://www.w3schools.com/react/react_hooks.asp