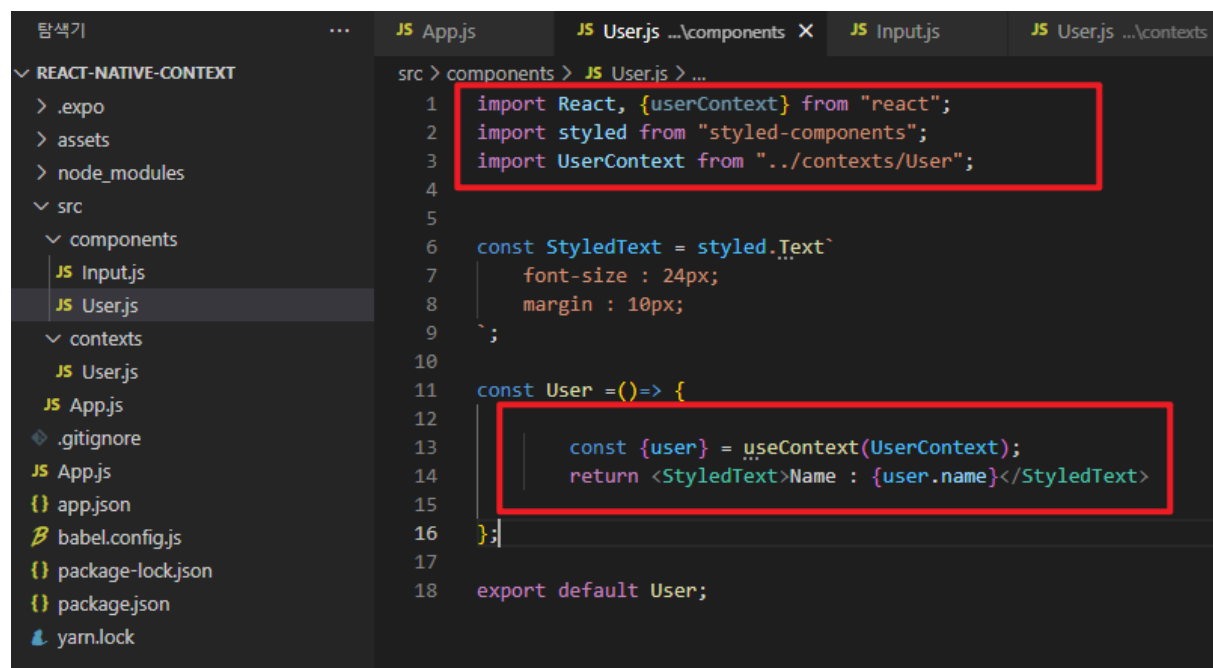


day49-rm-useContext

☰ 태그	
📅 날짜	@2022년 12월 8일

Consumer 컴포넌트의 자식 함수로 전달되던 값과 동일한 데이터를 반환하므로 Consumer 컴포넌트를 사용하지 않고 Context의 내용을 사용할 수 있게 한다.



```
src > components > JS User.js > ...
1  import React, {useContext} from "react";
2  import styled from "styled-components";
3  import UserContext from "../contexts/User";
4
5
6  const StyledText = styled.Text`
7    font-size : 24px;
8    margin : 10px;
9  `;
10
11  const User = () => {
12
13    const {user} = useContext(UserContext);
14    return <StyledText>Name : {user.name}</StyledText>
15  };
16
17  export default User;
```

Consumer 컴포넌트를 사용할 때는 Consumer 컴포넌트의 자식으로 반드시 리액트 컴포넌트를 반환하는 함수를 넣어야 하지만, useContext를 이용하면 Consumer 컴포넌트를 사용했을 때보다 사용법이 훨씬 간편하고 코드의 가독성도 좋아진다.

이제 Input 컴포넌트도 useContext를 이용하도록 수정해보자.

```
src > components > JS Input.js > [0] Input
1  import React, {useState, useContext} from "react";
2  import styled from "styled-components/native";
3  import UserContext from "../contexts/User";
4
5  const StyledInput = styled.TextInput`
6    border : 1px solid #606060;
7    width : 250px;
8    padding : 10px 15px;
9    margin: 10px;
10   font-size : 24px;
11 `;
12
13 const Input =()=>{
14   const [name, setName] = useState('');
15   const {dispatch} = useContext(UserContext);
16
17   return(
18     <StyledInput
19       value={name}
20       onChangeText={text=>setName(text)}
21       onSubmitEditing={()=>{
22         dispatch(name);
23         setName('');
24       }}
25       placeholder="Enter a name..."
26       autoCapitalize="none"
27       autoComplete={false}
28       returnKeyType="done"
29     />
30   );
31 };
32 export default Input;
```

useContext를 사용한 코드도 문제 없이 잘 동작하는 것을 확인할 수 있다.



Name : Lee Hong si

