



component-hook-factories

Validates against higher order functions defining nested components or hooks. Components and hooks should be defined at the module level.

Rule Details

Defining components or hooks inside other functions creates new instances on every call. React treats each as a completely different component, destroying and recreating the entire component tree, losing all state, and causing performance problems.

Invalid

Examples of incorrect code for this rule:

```
// ❌ Factory function creating components
function createComponent(defaultValue) {
  return function Component() {
    // ...
  };
}
```

```
// ❌ Component defined inside component
function Parent() {
  function Child() {
    // ...
  }

  return <Child />;
}
```

```
// ❌ Hook factory function
function createCustomHook(endpoint) {
  return function useData() {
```

```
// ...  
};  
}
```

Valid

Examples of correct code for this rule:

```
// ✅ Component defined at module level  
function Component({ defaultValue }) {  
  // ...  
}  
  
// ✅ Custom hook at module level  
function useData(endpoint) {  
  // ...  
}
```

Troubleshooting

I need dynamic component behavior

You might think you need a factory to create customized components:

```
// ❌ Wrong: Factory pattern  
function makeButton(color) {  
  return function Button({children}) {  
    return (  
      <button style={{backgroundColor: color}}>  
        {children}  
      </button>  
    );  
  };  
}  
  
const RedButton = makeButton('red');
```

Pass **JSX as children** instead:

```
// ✅ Better: Pass JSX as children
function Button({color, children}) {
  return (
    <button style={{backgroundColor: color}}>
      {children}
    </button>
  );
}

function App() {
  return (
    <>
      <Button color="red">Red</Button>
      <Button color="blue">Blue</Button>
    </>
  );
}
```

< [이전](#)
[rules-of-hooks](#)

[다음](#) >
[config](#)

 Meta Open Source

Copyright © Meta Platforms, Inc

uwu?

React 학습하기

[빠르게 시작하기](#)

[설치하기](#)

[UI 표현하기](#)

[상호작용성 더하기](#)

API 참고서

[React APIs](#)

[React DOM APIs](#)

State 관리하기

탈출구

커뮤니티

행동 강령

팀 소개

문서 기여자

감사의 말

더 보기

블로그

React Native

개인 정보 보호

약관

