



purity

Validates that [components/hooks are pure](#) by checking that they do not call known-impure functions.

Rule Details

React components must be pure functions - given the same props, they should always return the same JSX. When components use functions like `Math.random()` or `Date.now()` during render, they produce different output each time, breaking React's assumptions and causing bugs like hydration mismatches, incorrect memoization, and unpredictable behavior.

Common Violations

In general, any API that returns a different value for the same inputs violates this rule. Usual examples include:

- `Math.random()`
- `Date.now()` / `new Date()`
- `crypto.randomUUID()`
- `performance.now()`

Invalid

Examples of incorrect code for this rule:

```
// ❌ Math.random() in render
function Component() {
  const id = Math.random(); // Different every render
  return <div key={id}>Content</div>;
}
```

```
}  
  
// ❌ Date.now() for values  
function Component() {  
  const timestamp = Date.now(); // Changes every render  
  return <div>Created at: {timestamp}</div>;  
}
```

Valid

Examples of correct code for this rule:

```
// ✅ Stable IDs from initial state  
function Component() {  
  const [id] = useState(() => crypto.randomUUID());  
  return <div key={id}>Content</div>;  
}
```

Troubleshooting

I need to show the current time

Calling `Date.now()` during render makes your component impure:

```
// ❌ Wrong: Time changes every render  
function Clock() {  
  return <div>Current time: {Date.now()}</div>;  
}
```

Instead, [move the impure function outside of render](#):

```
function Clock() {  
  const [time, setTime] = useState(() => Date.now());  
  
  useEffect(() => {
```

```
const interval = setInterval(() => {
  setTime(Date.now());
}, 1000);

return () => clearInterval(interval);
}, []);

return <div>Current time: {time}</div>;
}
```

< [이전](#)
[preserve-manual-memoization](#)

[다음](#) >
[refs](#)

 Meta Open Source

Copyright © Meta Platforms, Inc

uwu?

React 학습하기

[빠르게 시작하기](#)

[설치하기](#)

[UI 표현하기](#)

[상호작용성 더하기](#)

[State 관리하기](#)

[탈출구](#)

커뮤니티

[행동 강령](#)

[팀 소개](#)

[문서 기여자](#)

[감사의 말](#)

API 참고서

[React APIs](#)

[React DOM APIs](#)

더 보기

[블로그](#)

[React Native](#)

[개인 정보 보호](#)

[약관](#)

