



v19.2



검색

Ctrl K

[API 참고서](#) > [LINTS](#) >

# use-memo

Validates that the `useMemo` hook is used with a return value. See [useMemo docs](#) for more information.

## Rule Details

`useMemo` is for computing and caching expensive values, not for side effects. Without a return value, `useMemo` returns `undefined`, which defeats its purpose and likely indicates you're using the wrong hook.

## Invalid

Examples of incorrect code for this rule:

```
// ❌ No return value
function Component({ data }) {
  const processed = useMemo(() => {
    data.forEach(item => console.log(item));
    // Missing return!
  }, [data]);

  return <div>{processed}</div>; // Always undefined
}
```

## Valid

Examples of correct code for this rule:

```
// ✅ Returns computed value
function Component({ data }) {
```

```
const processed = useMemo(() => {
  return data.map(item => item * 2);
}, [data]);

return <div>{processed}</div>;
}
```

## Troubleshooting

### I need to run side effects when dependencies change

You might try to use `useMemo` for side effects:

```
// ❌ Wrong: Side effects in useMemo
function Component({user}) {
  // No return value, just side effect
  useMemo(() => {
    analytics.track('UserViewed', {userId: user.id});
  }, [user.id]);

  // Not assigned to a variable
  useMemo(() => {
    return analytics.track('UserViewed', {userId: user.id});
  }, [user.id]);
}
```

If the side effect needs to happen in response to user interaction, it's best to colocate the side effect with the event:

```
// ✅ Good: Side effects in event handlers
function Component({user}) {
  const handleClick = () => {
    analytics.track('ButtonClicked', {userId: user.id});
    // Other click logic...
  };

  return <button onClick={handleClick}>Click me</button>;
}
```

```
}
```

If the side effect synchronizes React state with some external state (or vice versa), use `useEffect`:

```
// ✅ Good: Synchronization in useEffect
function Component({theme}) {
  useEffect(() => {
    localStorage.setItem('preferredTheme', theme);
    document.body.className = theme;
  }, [theme]);

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

이전



[unsupported-syntax](#)

 Meta Open Source

Copyright © Meta Platforms, Inc

uwu?

## React 학습하기

빠르게 시작하기

설치하기

UI 표현하기

상호작용성 더하기

State 관리하기

탈출구

## API 참고서

React APIs

React DOM APIs

커뮤니티

더 보기

[행동 강령](#)

[팀 소개](#)

[문서 기여자](#)

[감사의 말](#)

[블로그](#)

[React Native](#)

[개인 정보 보호](#)

[약관](#)

