

REACT

7. forduló



A kategória támogatója: TCS - Tata Consultancy
Services

Ismertető a feladathoz

Felhasznált idő: 00:00/20:00

Elért pontszám: 0/10

1. feladat 0/5 pont

Adott az alábbi komponens:

```
import { atom, useAtom } from 'jotai'

const countAtom = atom(0)

export function Counter() {
  const [count, setCount] = useAtom(countAtom)
  return (
    <h1>
      <p>{count}</p>
      <button onClick={() => setCount((c) => c + 1)}>one up</button>
    </h1>
  )
}
```

és a hozzá tartozó unit teszt:

```
import React from 'react'
import { render, screen, fireEvent } from '@testing-library/react'
import { Counter } from './Counter'

test('should increment counter', () => {
  render(<Counter />)

  const counter = screen.getByText('0')
  const incrementButton = screen.getByText('one up')

  fireEvent.click(incrementButton)
```

```
expect(counter.textContent).toEqual('1')
}))
```

Melyik cserékre fut le a fenti teszt, ha kicseréljük a komponens-ben ezt a sort:

```
`import { atom, useAtom } from 'jotai`
```

Válasz



```
import { useAtom, atom } from 'jotai'
```

Ez a válasz helyes, de nem jelölted meg.



```
import { atom, useAtom } from 'recoil'
```



```
import { atom, atom as useAtom } from 'recoil'
```

Magyarázat

A `"import { useAtom, atom } from 'jotai'"` válasz **helyes**, mivel az importálásnál nem számít a nevek sorrendje.

A `"import { atom, useAtom } from 'recoil'"` válasz **helytelen**, mivel a recoil nem exportál useAtom nevű függvényt.

A `"import { atom, atom as useAtom } from 'recoil'"` válasz **helytelen**, mivel a recoil ugyan exportál egy atom nevű függvényt, a kód nem helyesen használná azt:

```
const countAtom = atom(0)
//...
const [count, setCount] = useAtom(countAtom)
```

További információt, segítséget a feladat helyes megoldásához az alábbi dokumentációs linken találhattok:

<https://jotai.org/docs/guides/testing>

2. feladat 0/5 pont

Zustand forráskódját olvasva az alábbi error-t dobja:

```
const useStoreApi = () => {
  const store = useContext(ZustandContext)
  if (!store) {
    throw new Error(
      'Seems like you have not used zustand provider as an ancestor.'
    )
  }
}
```

Ez mikor történhet?

Válaszok

```
import ReactFlow, { ReactFlowProvider } from 'react-flow-renderer';

function Flow(props) {
  const reactFlowInstance = useReactFlow();

  return <ReactFlow {...props} />;
}

function FlowWithExternalProvider(props) {
  return (
    <ReactFlowProvider>
    <Flow {...props} />
    </ReactFlowProvider>
  );
}

export default FlowWithExternalProvider;
```



```
import ReactFlow from 'react-flow-renderer';

function FlowWithoutProvider(props) {
  const reactFlowInstance = useReactFlow();

  return <ReactFlow {...props} />;
}

export default FlowWithoutProvider;
```

Ez a válasz helyes, de nem jelölted meg.



```
import ReactFlow, { ReactFlowProvider } from 'react-flow-renderer';

function FlowWithProvider(props) {
  const reactFlowInstance = useReactFlow();

  return (
    <ReactFlowProvider>
    <ReactFlow {...props} />
    </ReactFlowProvider>
  );
}

export default FlowWithProvider;
```

Ez a válasz helyes, de nem jelölted meg.

Magyarázat

Ahhoz, hogy a `useReactFlow()` horgot használj tudjuk egy komponensben és ne kapjuk hibát, már a szülő komponensben szükség van a `ReactFlowProvider` használatára.

Ezért a `FlowWithExternalProvider`-t tartalmazó válasz helytelen, mivel a komponens jól működik nem dob hibát.

A `FlowWithoutProvider`-t tartalmazó válasz helyes, mivel nincs használva a `ReactFlowProvider`, ezért a `useReactFlow` használta a kérdésben leírt hibát dobja.

A `FlowWithProvider`-t tartalmazó válasz helyes, mivel hiába van használva a `ReactFlowProvider`, az csak a gyermek komponensekben lesz használható ezért a `useReactFlow` használta itt is hibát dob.

További információt, segítséget a feladat helyes megoldásához az alábbi dokumentációs linken találhattok:
<https://github.com/pmndrs/zustand> illetve <https://reactflow.dev/docs/guides/troubleshooting/>



[Legfontosabb tudnivalók](#)

[Kapcsolat](#)

[Versenyszabályzat](#)

[Adatvédelem](#)

© 2023 Human Priority Kft.

KÉSZÍTETTE **cone**

Megjelenés

 Világos 