

# REACT.JS

6. forduló

  
**accenture**

A kategória támogatója: Accenture

RENDELKEZÉSRE ÁLLÓ IDŐ:

15:00

## Ismertető a feladathoz

Felhasznált idő: 01:49/15:00

Elért pontszám: 0/16

### 1. feladat 0/2 pont

Melyik az a Jest CLI paraméter, amivel a teszteteket egymás után futtatja a Jest és nem párhuzamosan?

A megoldást a kötőjelek nélkül kell megadni. (pl. cliOption)

#### Válaszok

A helyes válasz:

runinband  
runInBand

#### Magyarázat

<https://jestjs.io/docs/cli#--runinband>

## 2. feladat 0/4 pont

Melyik Jest konfigurációs opciót kell testre szabni, ha olyan npm library-t akarunk használni, ami nem közvetlenül futtatható JavaScript kódot tartalmaz és ezért azt még külön fel kell dolgozni (pl. ES modules, TypeScript, stb.)?

### Válasz

- ☒ transformIgnorePatterns
- ☐ unmockedModulePathPatterns
- ☐ testPathIgnorePatterns
- ☐ modulePathIgnorePatterns

### Magyarázat

Az **unmockedModulePathPatterns** válasz helytelen: ennek a beállításnak a mock-oláshoz van köze.

A **testPathIgnorePatterns** válasz helytelen: ennek a beállításnak a test-ek megtalálásához van köze.

A **modulePathIgnorePatterns** válasz helytelen: ennek a beállításnak köze van a modulok megtalálásához és betöltéséhez, de nincs köze azok feldolgozásához.

## 3. feladat 0/6 pont

Melyik Enzyme metódus hívások adják vissza a MyComponent komponenst?

```
const wrapper = mount(  
  <div>  
    <MyComponent value="21" />  
  </div>  
>);
```

### Válaszok

- ☐ wrapper.find("MyComponent[value=21]");
- ☒ wrapper.find("MyComponent[value='21']");
- ☒ wrapper.find("MyComponent[value^='2']");
- ☐ wrapper.find({ value: 21 });
- ☒ wrapper.find({ value: "21" });
- ☐ wrapper.find({ value: "2\*" });
- ☐ wrapper.find({ value: "2" }, { comparer: "contains" });

## Magyarázat

A `"wrapper.find("MyComponent[value=21]);"` megoldás helytelen: a `value` tulajdonság értéke a `"21"` karakterlánc és nem a `21` egész szám.

A `"wrapper.find({ value: 21 });"` megoldás helytelen: a `value` tulajdonság értéke a `"21"` karakterlánc és nem a `21` egész szám.

A `"wrapper.find({ value: "2*" });"` megoldás helytelen: a csillag és a részleges keresés ebben az esetben nem támogatott.

A `"wrapper.find({ value: "2" }, { comparer: "contains" });"` megoldás helytelen: nincsen `comparer` opciója a `find`-nak

## 4. feladat 0/2 pont

Mi annak a React Testing Library metódusnak neve, amivel aktualizálhatjuk az éppen tesztelt komponenst miután annak a props-ját megváltoztattuk egy tesztben?

### Válaszok

A helyes válasz:

`render`

## Magyarázat

<https://testing-library.com/docs/react-testing-library/api#render>

## 5. feladat 0/2 pont

Melyik React Testing Library lekérdező metódus család várakozik automatikusan a keresett DOM element-re?

### Válasz

- ☐ `getBy...`
- ☒ `findBy...`
- ☐ `queryBy...`
- ☐ mindhárom (`getBy...`, `findBy...`, `queryBy...`)
- ☐ egyik sem

## Magyarázat

A `getBy` és a `queryBy` metódusok szinkron térnek vissza, míg a `findBy` metódusok egy `Promisal`-al térnek vissza és alaphelyzetben 1000ms-ot várnak a kívánt element-re.

[Legfontosabb tudnivalók](#)

[Kapcsolat](#)

[Versenyszabályzat](#)

[Adatvédelem](#)

© 2022 Human Priority Kft.

KÉSZÍTETTE

Megjelenés

 Világos 