

✔ Congratulations! You passed!

Grade received 100% Latest Submission Grade 100% To pass 80% or higher

[Go to next item](#)

1. How is array destructuring relevant to hooks in React?

1 / 1 point

- ☐ It makes it possible to handle click events.
- ☐ It makes it possible to reassign state objects.
- ☒ It is a way to get individual items from an array of items, and save those individual items as separate components.
- ☐ It makes the Virtual DOM possible.



Correct

Correct. It *is* a way to get individual items from an array of items, and save those individual items as separate components.

2. Is the following paragraph correct?

1 / 1 point

With array destructuring, you are free to give any variable name to the items that you destructure from an array. Contrary to that, when destructuring objects, you have to destructure a property of an object using that exact property's name as the name of the destructured variable.

- ☒ Yes
- ☐ No



Correct

That's right, the paragraph is correct.

3. The `useEffect` hook is a way to:

1 / 1 point

- ☒ handle a *side effect*.
- ☐ handle visual effects (animations and transitions) in React
- ☐ handle one-way data flows



Correct

That's correct! The name of the `useEffect` hook is closely related to the concept of an *effect*, or more precisely, of a *side effect*.

4. Which answer is correct about the following code snippet?

1 / 1 point

```
1  useEffect( () => {  
2    |  if (data !== '') {  
3    |    setData('test data')  
4    |  }  
5  })
```

- ☒ This code is not breaking the rules of hooks
- ☐ This code is breaking the rules of hooks
- ☐ This code is ok, except the fact that you should replace the `if` statement with a ternary operator.



Correct

Correct. The code snippet is valid.

5. Choose an example of a side-effect with which you'd need to use a `useEffect` hook:

1 / 1 point

- ☐ Render some prop values on the screen.

- ☒ Run a Fetch API call in React.
- ☐ Update the value of the state variable in a child component.

✓ **Correct**

Correct. The call to the Fetch API is a side-effect, and you should use the `useEffect` hook to handle it.

6. Complete the sentence:

1 / 1 point

The `useState` hook starts with an initial state, but...

- ☐ the `useReducer` hook cannot be used to track the state at all.
- ☐ the `useReducer` hook must be used when the initial state is an object.
- ☒ the `useReducer` hook gets a reducer function in addition to the initial state.

✓ **Correct**

Correct. The `useState` hook starts with an initial state, but the `useReducer` hook gets a reducer function in addition to the initial state.

7. True or false: `useRef` is a custom hook in React.

1 / 1 point

- ☐ Yes.
- ☒ No

✓ **Correct**

Correct. The `useRef` hook is a built-in hook, just like some other hooks are, such as the `useState` or `useReducer`.

8. JavaScript is a single-threaded language, meaning...

1 / 1 point

- ☒ ...it can only do a single thing at any given time.
- ☐ ...you can use it with React only when this single thread is passed to the `useState` variable.
- ☐ ...you can use it with React only when this single thread is used with the `useEffect` hook.

✓ **Correct**

Correct. This is a true statement.

9. Which statement is correct about the following code snippet:

1 / 1 point

```
1 import { useEffect } from "react";
2
3 function useConsoleLog(varName) {
4   useEffect(() => {
5     console.log(varName);
6   });
7 }
8
9 export default useConsoleLog;
```

Choose the correct statement below.

- ☒ This is an example of a custom hook.
- ☐ This code is an example of an implicit state-updating function.
- ☐ This code is an example of an explicit state-updating function.

✓ **Correct**

Correct. This is an example of a custom hook.

10. Find the error in this code:

1 / 1 point

```

1  import {useState} from "react";
2
3  export default function App() {
4    const [restaurantName, setRestaurantName] = useState("Lemon");
5
6    function updateRestaurantName() {
7      useRestaurantName("Little Lemon");
8    };
9
10   return (
11     <div>
12       <h1>{restaurantName}</h1>
13       <button onClick={updateRestaurantName}>
14         Update restaurant name
15       </button>
16     </div>
17   );
18 };

```

- ☒ The code inside the `updateRestaurantName()` function definition should not invoke `useRestaurantName("Little Lemon")`
- ☐ The state-setting function's variable name in the array destructuring should not be `setRestaurantName`.
- ☐ The `onClick` event should not be this:

`onClick={updateRestaurantName}`



Correct

Correct. The code inside the `updateRestaurantName()` function should invoke `setRestaurantName("Little Lemon")`.