

Started on	Thursday, 16 September 2021, 4:59 PM
State	Finished
Completed on	Thursday, 16 September 2021, 5:01 PM
Time taken	2 mins 23 secs
Grade	6.00 out of 6.00 (100%)

Question 1

Correct

Mark 2.00 out of 2.00

Match the utilities from Redux and React-Redux to their purpose.

createStore()

Create the Redux store as the repository for application state.



<Provider>

Makes the Redux store available to nested components that are wrapped in the connect() function.



connect()

Subscribe components to the Redux store.



mapStateToProps()

Selects the part of the state that a component needs from the Redux store and pass it to the component as props.



Your answer is correct.

The correct answer is: createStore() → Create the Redux store as the repository for application state., <Provider> → Makes the Redux store available to nested components that are wrapped in the connect() function., connect() → Subscribe components to the Redux store., mapStateToProps() → Selects the part of the state that a component needs from the Redux store and pass it to the component as props.

Question 2

Correct

Mark 1.00 out of 1.00

Given the following code:

```
const today = "Friday";  
const matinee = true;
```

What is the value of ticketPrice after this ternary operation?:

```
const ticketPrice = (today === "Friday") && (!matinee) ? 10 : 7;
```

Select one:

- ☒ a. 7
- ☐ b. NaN
- ☐ c. 10

✓ Correct. By the logic presented, all matinees are set to a ticketprice of 7.

Your answer is correct.

The correct answer is: 7

Question 3

Correct

Mark 1.00 out of 1.00

When you bind a method in a class component using **bind()**, where in the class component should this binding occur?

Select one:

- ☐ a. in the render method
- ☒ b. in the constructor method
- ☐ c. in the export statement
- ☐ d. in the method definition of the method to be bound
- ☐ e. in the return statement



Correct. The binding of class component methods using `bind()` should occur inside the class constructor.

Your answer is correct.

The correct answer is: in the constructor method

Question 4

Correct

Mark 1.00 out of 1.00

Which of these are NOT a principle of Redux? (More than one answer.)

Select one or more:

☒ a. Separation of concerns.

✓ Correct. While Redux does separate concerns, this is not one of its three core principles.

☐ b. State is read-only.

☐ c. Changes are made with pure functions.

☐ d. Single source of truth.

☒ e. Unidirectional data flow.

✓ Correct. This is one of the principles of the Flow architecture, which Redux is influenced by. While it is not technically one of the core principles of Redux, Redux *does* have unidirectional data flow.

Your answer is correct.

The correct answers are: Unidirectional data flow. , Separation of concerns.

Question 5

Correct

Mark 1.00 out of 1.00

What is the value of **iceCream2** after the following code has executed?

```
const iceCream1 = {  
  flavor: "mint chocolate",  
  scoops: 2  
};  
  
const iceCream2 = {...iceCream1, flavor: "pistachio"};
```

Select one:

- ☐ a. {flavor: "pistachio"}
- ☐ b. {flavor: "mint chocolate", scoops: 2}
- ☐ c. {{iceCream1}, flavor: "pistachio"}
- ☒ d. {flavor: "pistachio", scoops: 2}

✓ Correct! The flavor property would have been updated.

Your answer is correct.

The correct answer is: {flavor: "pistachio", scoops: 2}

