

Lab 6: About Redux, Redux Thunk and Redux Toolkit

To build a ReactJS application with a user interface that requires the user to input a list of questions and answer options for each question, and then display the questions and answer options to the user while checking whether the user has selected the correct or incorrect answer, you can utilize Redux, Redux Thunk, and Redux Toolkit.

1. Setting up the Redux Store:

- Install the necessary dependencies: `redux`, `react-redux`, `redux-thunk`, and `@reduxjs/toolkit`.

```
"dependencies": {  
  "@reduxjs/toolkit": "^2.0.1",  
  "react-redux": "^8.1.3",  
  "redux": "^4.2.1",  
  "redux-thunk": "^2.4.2",  
}
```

- Create a Redux store using Redux Toolkit's `configureStore` function.
- Define a Redux slice using `createSlice` to manage the state for the quiz questions and answers.
- Export the slice's actions and reducer.

2. Creating the Quiz Component:

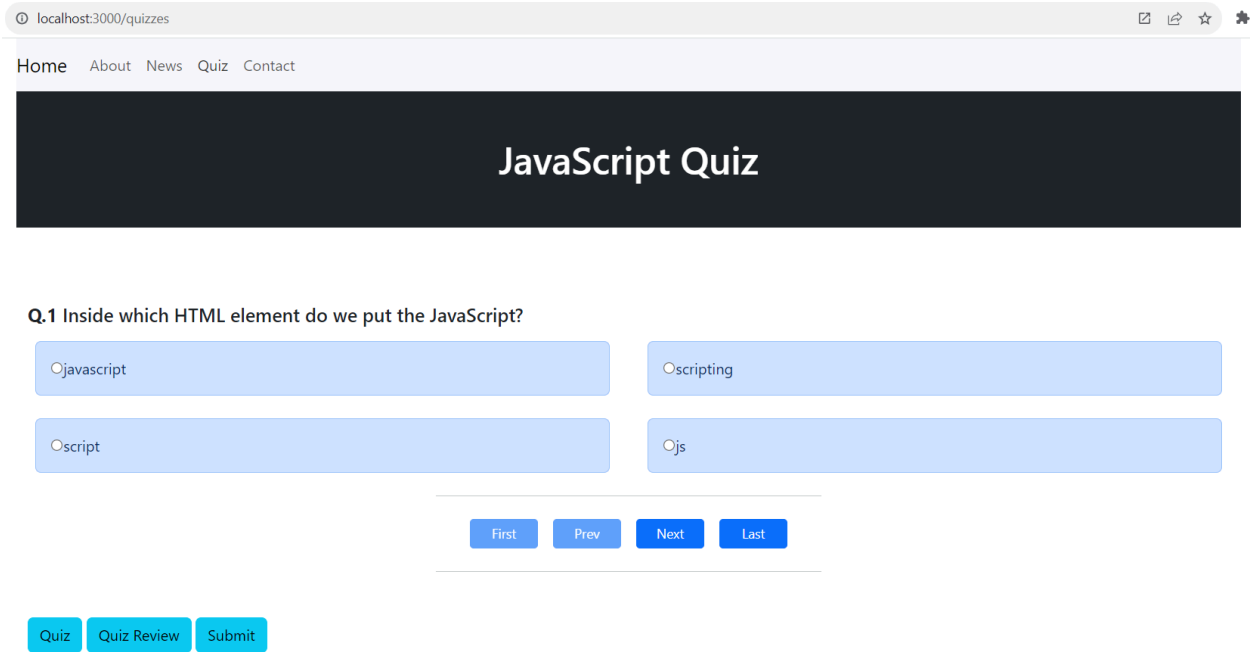
- Import the necessary dependencies: `react-redux` and the Redux slice actions.
- Create a functional component called `Quiz` that will render the quiz questions and answer options.
- Use `createSlice` is a function provided by the Redux Toolkit that combines both creating reducers and actions for a "slice" of application state.
- Render the quiz questions and answer options using JSX.

3. Displaying Quiz Questions and Answer Options:

- Use the `map` function to iterate over the quiz questions and answer options in the Redux store.
- Render each question and its corresponding answer options using JSX.
- Add event handlers to track the user's selection of answer options.
- Highlight the selected answer option and update the Redux store with the user's selection.

4. Checking User Answers:

- Add a button to the `Quiz` component to allow the user to check their answers.
- Create a function to compare the user's selected answer options with the correct answer options.
- Display a message to the user indicating whether their answers are correct or incorrect.



The screenshot shows a web browser at localhost:3000/quizzes. The page has a navigation bar with links: Home, About, News, Quiz, and Contact. Below the navigation bar is a dark header with the text "JavaScript Quiz". The main content area displays a question: "Q.1 Inside which HTML element do we put the JavaScript?". There are four radio button options: ☐ javascript, ☐ scripting, ☐ script, and ☐ js. Below the options is a set of navigation buttons: First, Prev, Next, and Last. At the bottom of the page, there are three buttons: Quiz, Quiz Review, and Submit.

localhost:3000/quiz/result

Home About News Quiz Contact

Quiz Review

Q1. Inside which HTML element do we put the JavaScript?

- ☒ javascript
- ☐ scripting
- ☐ script
- ☐ js

Right answer is: **script**

Q2. What are variables used for in JavaScript Programs?

- ☒ Storing numbers, dates, or other values
- ☐ Varying randomly
- ☐ Causing high-school algebra flashbacks
- ☐ None of these

Right answer is: **Storing numbers, dates, or other values**

Q3. Which of the following can't be done with client-side JavaScript?

- ☐ Validating a form
- ☐ Sending a form's contents by email

localhost:3000/quiz/review

Home About News Quiz Contact

Quiz Review

<u>Question No</u> 1	<u>Question No</u> 2	<u>Question No</u> 3	<u>Question No</u> 4	<u>Question No</u> 5	<u>Question No</u> 6	<u>Question No</u> 7	<u>Question No</u> 8	<u>Question No</u> 9	<u>Question No</u> 10
Answered	Answered	Answered	Answered	Answered	Answered	Answered	Answered	Answered	Answered