

Quiz Section 2: React

In this section, we will create a simple Trivia React App that will count the number of answers that the user gets correct. To get started, check out the starter code using the command

```
git clone https://gitlab.cs.washington.edu/cse331-25sp/materials/sec02.git
```

To install the modules `cd sec02` and run `npm install --no-audit`. Then, start it with `npm run start`. Point your browser at `http://localhost:8080`. You should see an Application titled “Animal Trivia”.

Take a moment to look at the starter code given. The Trivia question along with the answers to the questions are stored in an array in `src/trivia.ts`. You *should not* modify this file. The main logic of the application is in `App.tsx`. The App component has a state that keeps track of the current question, the user’s score, and whether the user has answered all questions correctly. The App component also has methods to check if the user’s answer is correct, to update the score, and to get a new question.

Navigate to <https://comfy.cs.washington.edu/service/hw2-practice> to access the Debugging Log for this section. You will be using this log to record the bugs you find and how you fix them. For each of the tasks below, you will be asked to find and fix a bug in the code. You will also be asked to add a log entry to the Debugging Log with the bug you found and how you fixed it.

Task 1 – React Bug

Currently, when you submit your answer, nothing changes in the UI.

- (a) Add a debugger statement at the top of the `doCheckAnswerClick` method in `App.tsx`. Confirm that the method is being called.
- (b) The `doCheckAnswerClick` method updates the state to include the field called “correct”, but this information is not currently being used in render.

Update render to show some HTML (either a congratulatory message or one saying that the answer is incorrect) when `this.state.correct` is not undefined.

Verify that the HTML now appears on the screen when you submit an answer.

- (c) Add a log entry to the Debugging Log with the bug you found and how you fixed it

Task 2 – Incorrect

Currently, when you submit the correct answer, the app does not recognize it as correct.

- (a) Look at the `doCheckAnswerClick` method in `App.tsx` and add a debugger statement check what values are being compared. Are they what you expect?
- (b) Fix the bug so that the correct answer is recognized as correct.
- (c) Add a log entry to the Debugging Log with the bug you found and how you fixed it.

Task 3 – State Bug

Currently, when you submit the correct answer, the score does not increase.

- (a) Look at the `doCheckAnswerClick` method in `App.tsx` and add a debugger statement to check if the `score` field in the state is being updated when the user submits the correct answer.
- (b) How is the `score` field being updated in the `doCheckAnswerClick` method? Is the `score` field being updated correctly? Fix the bug so that the score increases when the user submits the correct answer.
- (c) Add a log entry to the Debugging Log with the bug you found and how you fixed it.

Task 4 – Score Again

Currently, when you submit the correct answer to a question multiple times, the number of correct answers increases each time you submit the answer. The score should only increase the first time you submit the correct answer.

- (a) Look at the `doCheckAnswerClick` method in `App.tsx` and add a debugger statement to see why the score is increasing each time you submit the correct answer.
- (b) Notice that we have an Array of string `questionsAnswered` in our app state that keep track of the answers the user has answered the question correctly. Is this field being set correctly?
- (c) Fix the bug so that the score only increases the first time the user submits the correct answer.
- (d) Add a log entry to the Debugging Log with the bug you found and how you fixed it. Was bug was due to mutating something that should not have been mutated?