React – AJAX

In this lab, you will modify a React app and add AJAX calls to the server.

# Objectives

In this lab, you will

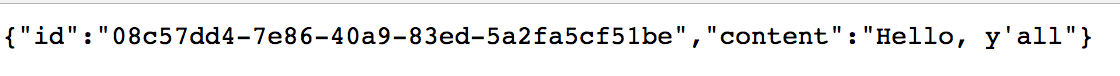
* Examine the server REST interface for the Todos
* Modify the todos controller to use the REST interface
* Run the app

# Run the solution verify correctness

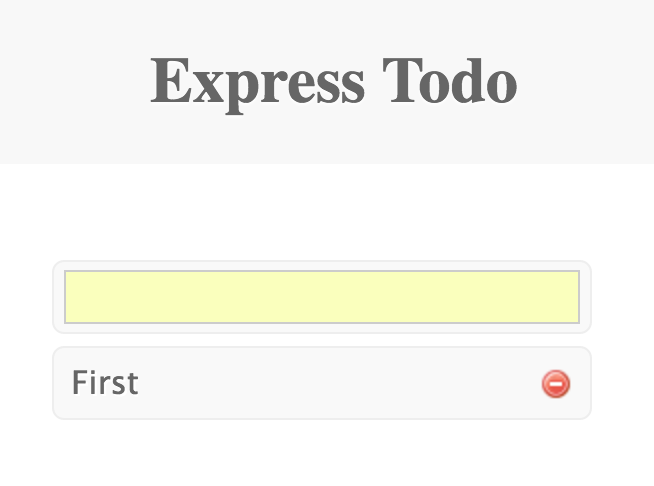
1. There are two servers in this lab, react-app and restful-server. The restful-server is an Express app that services the todos with the following API:

|  |  |  |
| --- | --- | --- |
| Method | URI | Results |
| GET | /v1/todos | Returns an array of all TODO items |
| GET | /v1/todos/1234 | Returns a single TODO item |
| POST | /v1/todos | Creates a TODO item. Body contains a single property: content |
| POST | /v1/todos/1234 | Updates a TODO item. Body contains two properties: id and content |
| DELETE | /v1/todos/1234 | Delete a single TODO item |

1. In a terminal or command window, go to the solution/restful-server folder. Install and start the server with yarn install and yarn start.
2. Open the browser with <http://localhost:3080/v1/todos> and you should see the list of todos similar to the following:



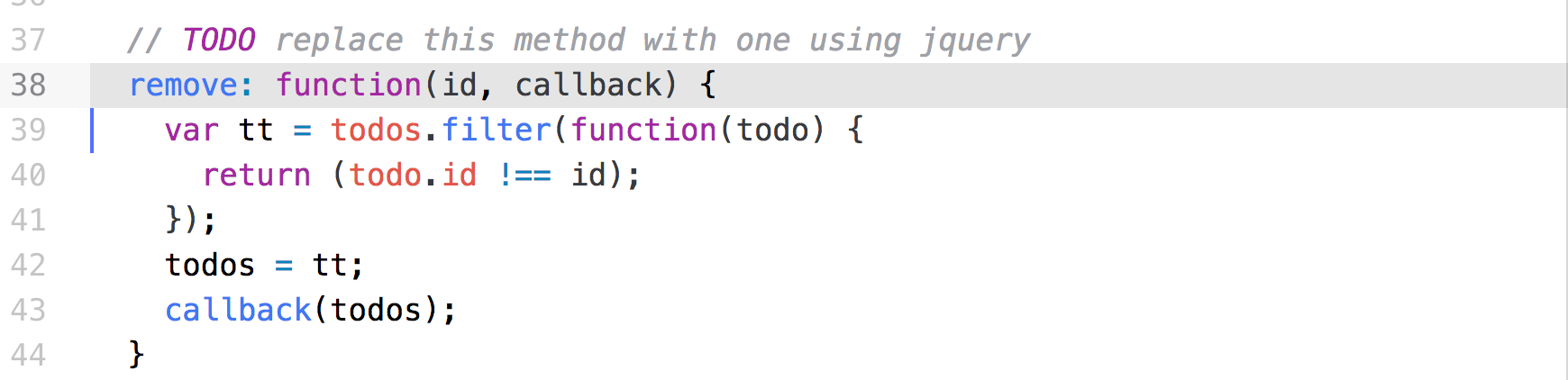
1. In a different terminal or command window, go to the solution/react-app folder. Install and start the server with yarn install and yarn start.
2. Open the browser with <http://localhost:3000> and you should see the following:



1. Stop the react-app server but leave the restful-server running. We use the restful-server to access the todos in the exercise.

# Test the Lab App

1. Go to the lab/react-app folder and install and start the server with yarn install and yarn start.
2. Open the browser with <http://localhost:3000> and notice the page is the same as the solution. Currently, the lab uses a version of todo-controller.js that uses an in-memory copy of the todo list.
3. Let’s refactor the src/components/todo-controller.js to use AJAX to access the RESTful interface on the server.
4. The remove() method looks like this:



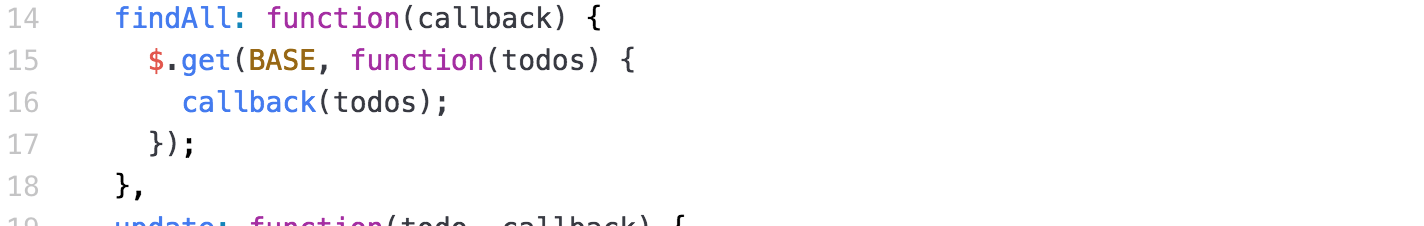
1. AJAX makes calls to the URL. The BASE of the URL points to the RESTful server started earlier. It is defined like this:



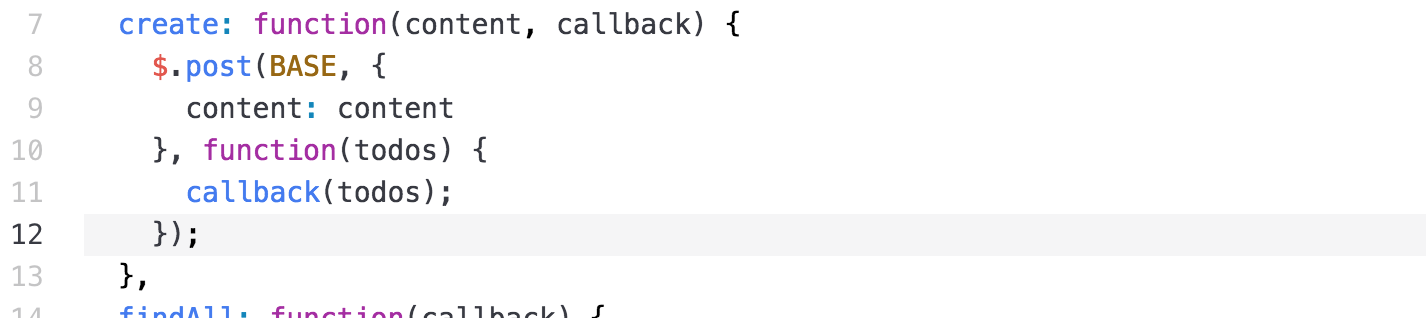
1. The AJAX remove() method should look like this:



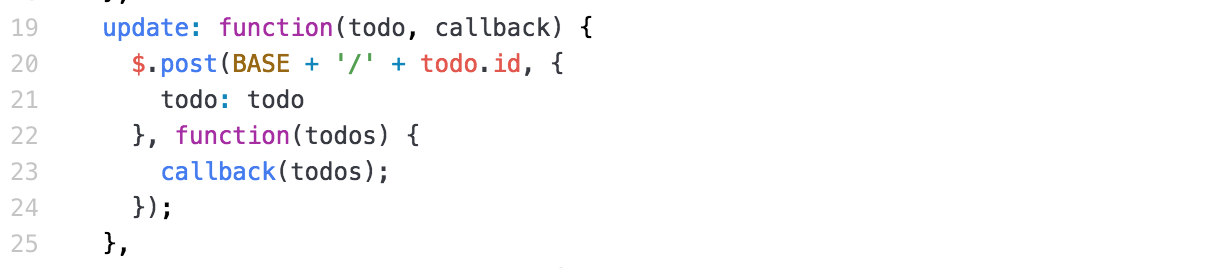
1. The AJAX call uses jQuery ($.ajax()) at the url, /v1/todos/:id .
2. The findAll() method should use the following AJAX call:



1. The create() method should use:



1. Notice it sends the content field in the body.
2. The update() method should use:



1. Notice it sends the id as part of the URL.
2. Now, import the jQuery library using:

**var *$*** = ***require***(**'jquery'**);

1. Restart the server and refresh the page. Add and delete some TODO items.

# Extra Credit

1. Instead of using jQuery for the AJAX calls, use the library, ‘axios’ or ‘isomorphic-fetch’ instead.

Congratulations. You have completed this lab.