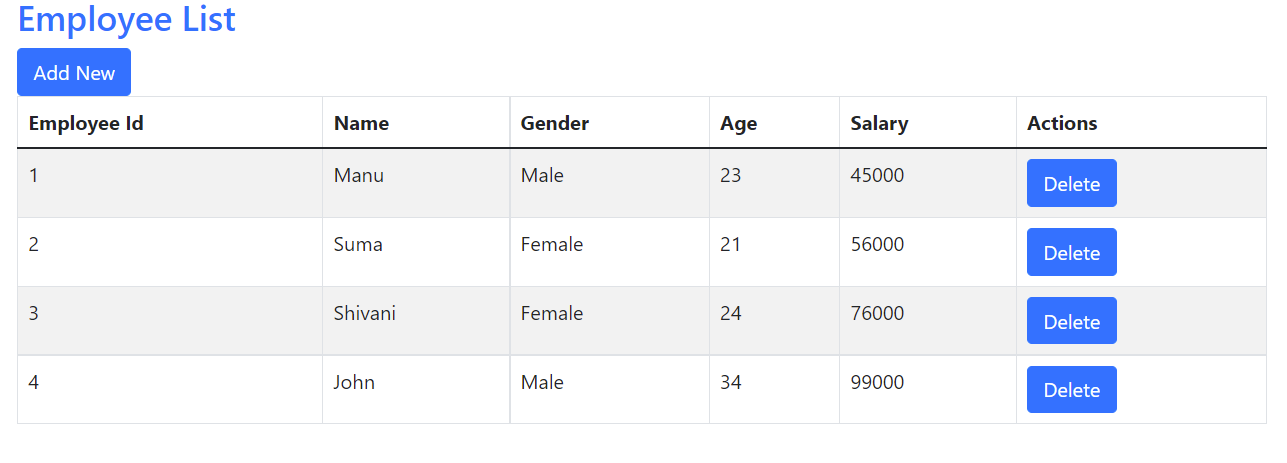
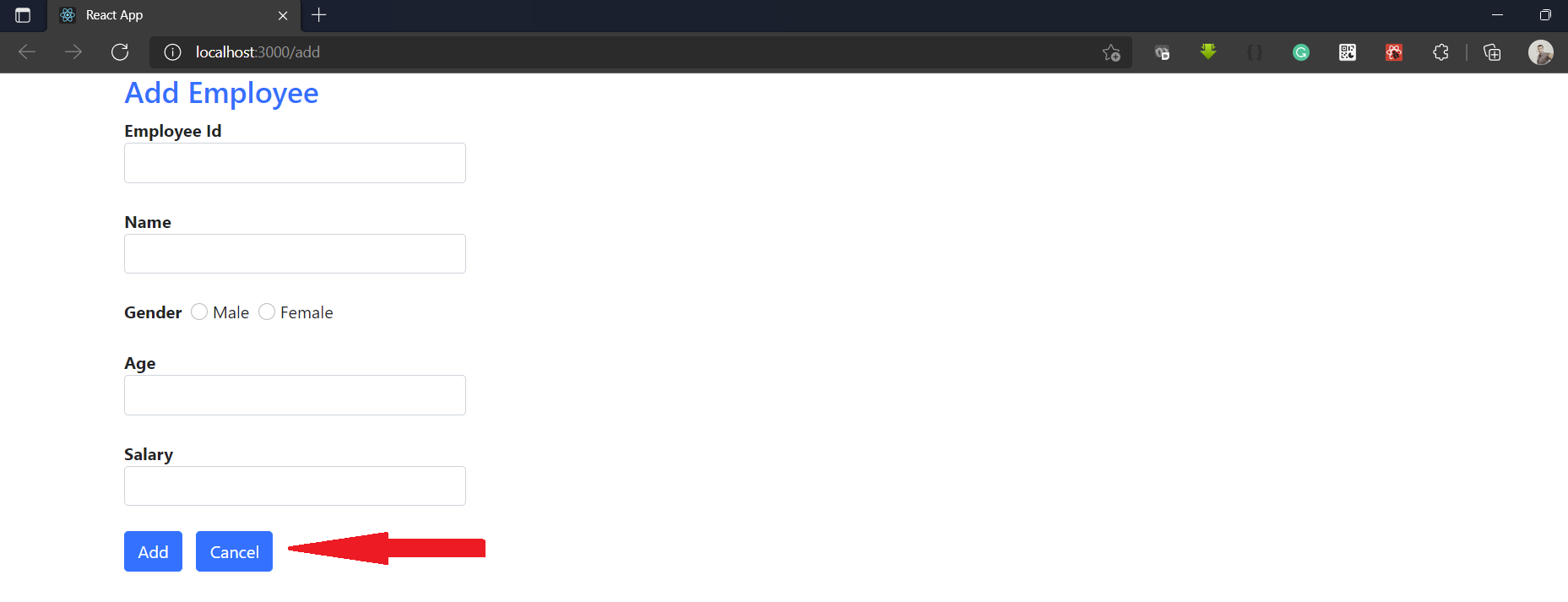
Create a React Application which consumes the data exposed by the spring boot REST API.

**Specifications.**

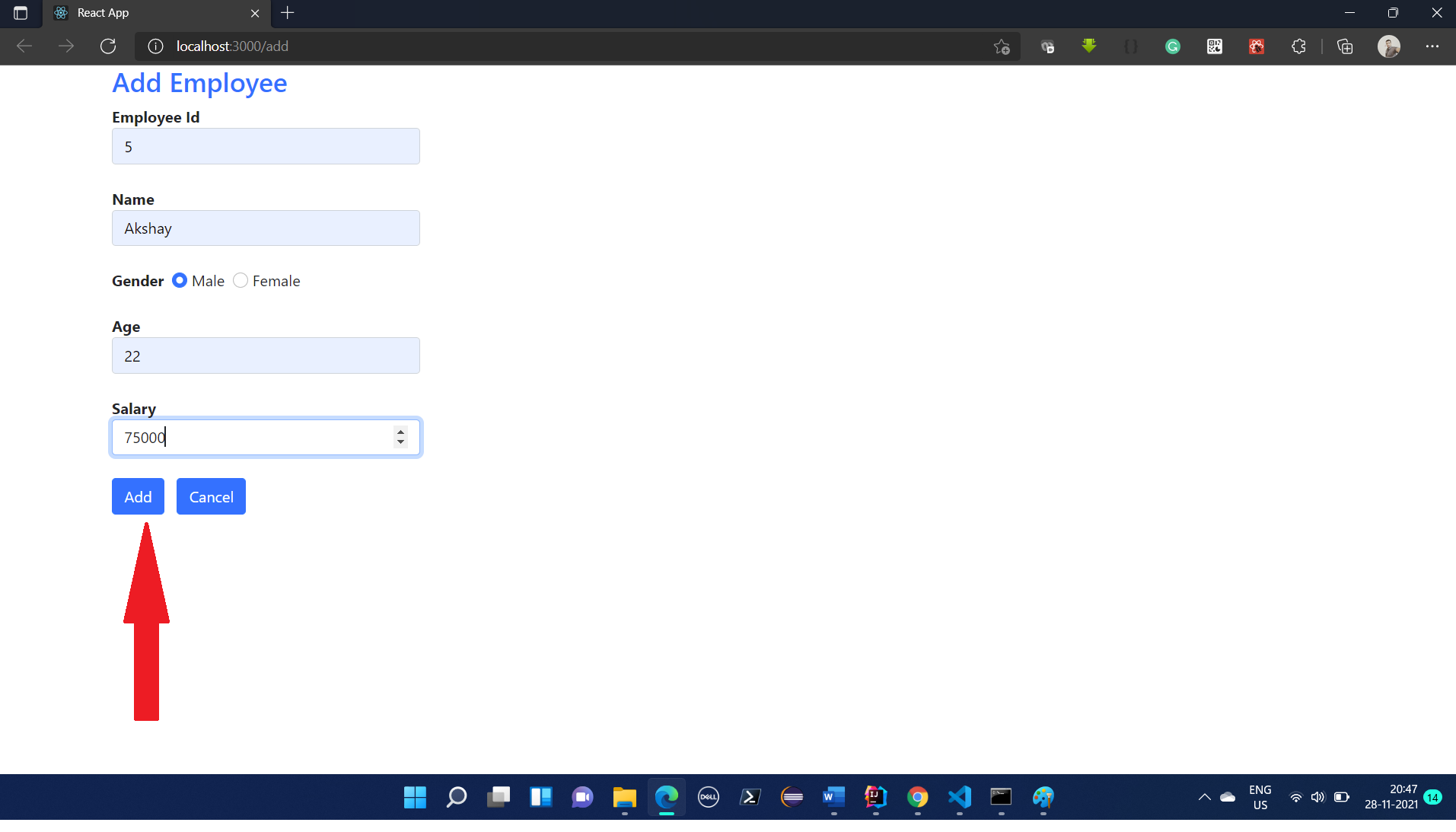
1. Focus of this hands-on is consuming Spring REST API from React application.
2. Not focusing much on exception handling.
3. Use H2 database for backend datastore.
4. Spring REST service should be deployed in AWS Elastic Beanstalk.
5. React application when started should display the following screen which enables you to perform all list, add and delete operations.



1. On clicking “Add New” button, the “Add Employee” screen should be displayed as shown below.

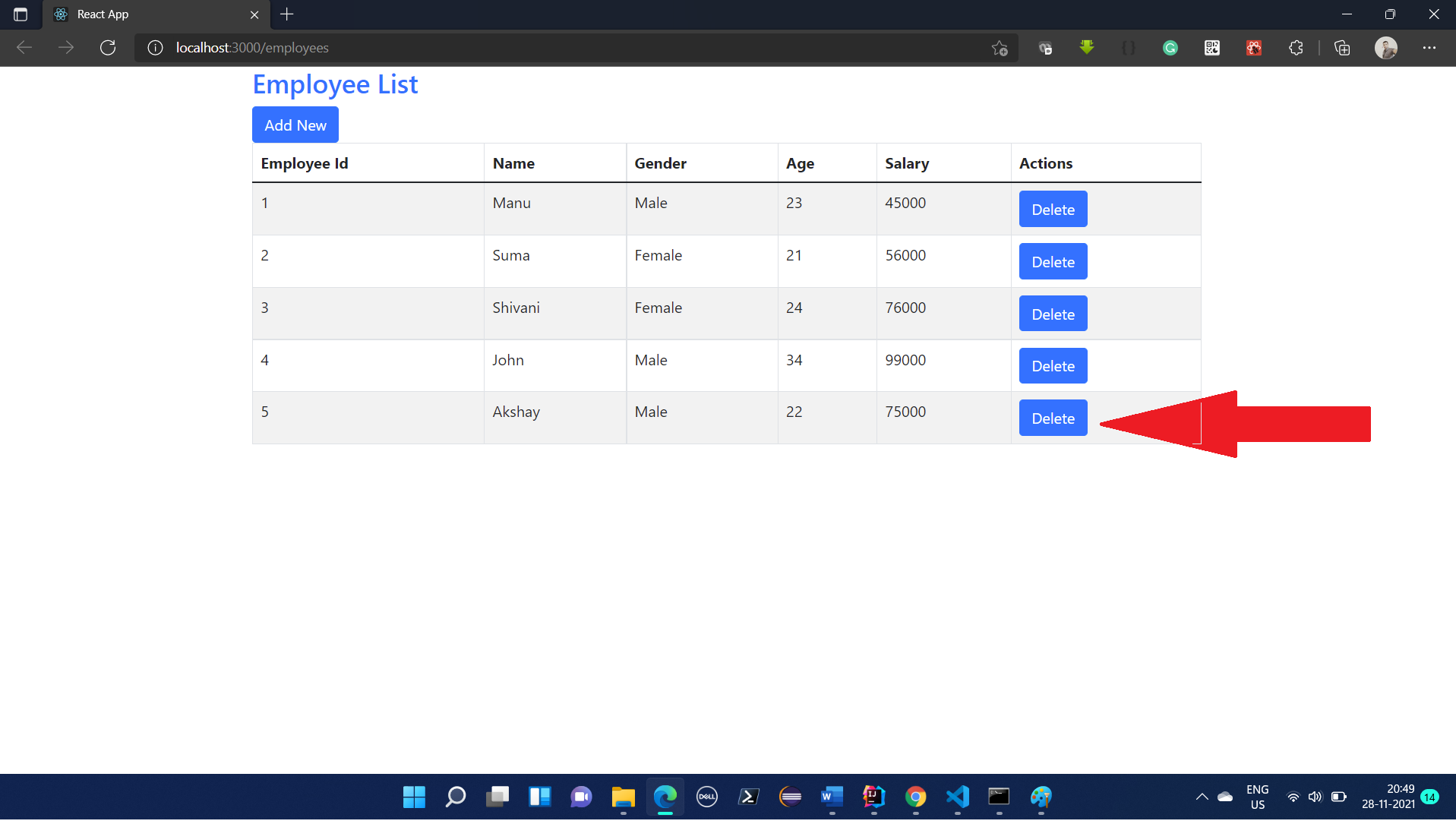


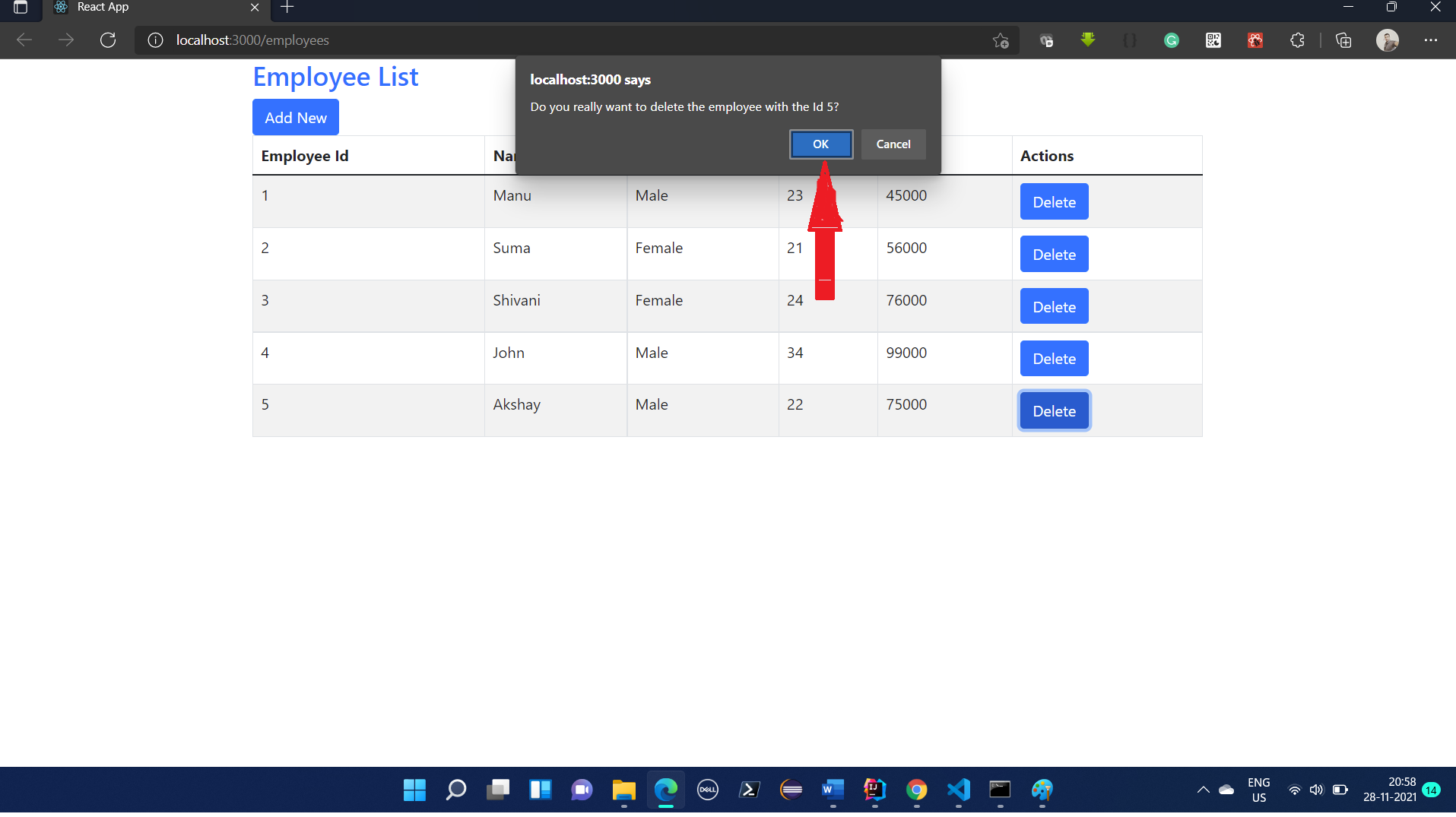
1. On clicking the “Cancel” button, application should navigate back to the main screen.
2. On entering the new Employee details and clicking on the “Add” button , the data should be stored in the database and you should be navigated back to the main screen and should display the newly added employee also.





1. On clicking the “Delete” Button, user should be Prompted with “Delete Employee” confirmation screen as shown below.





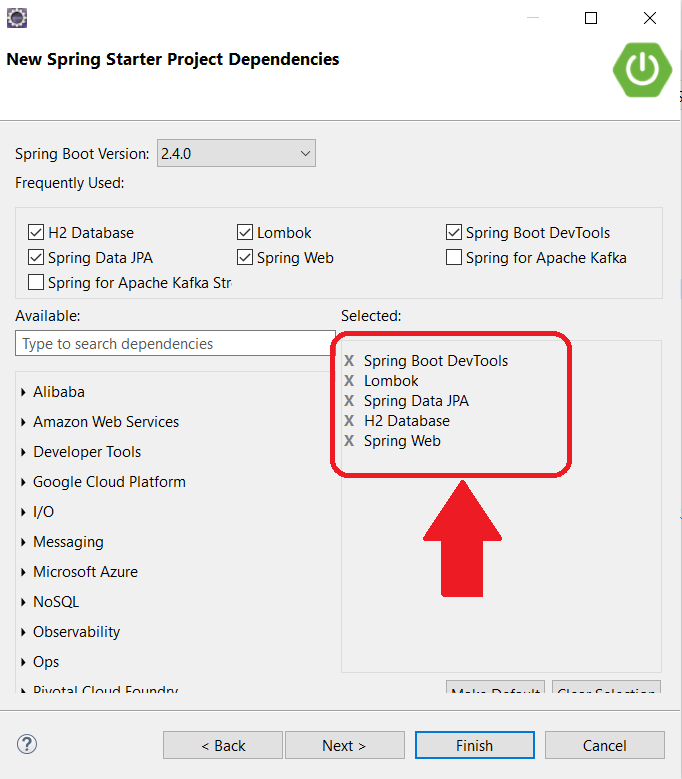
1. On Clicking “Ok” button, the corresponding employee record should be deleted from the database and the same status should be reflected in the employee list page.

Graphical user interface, application

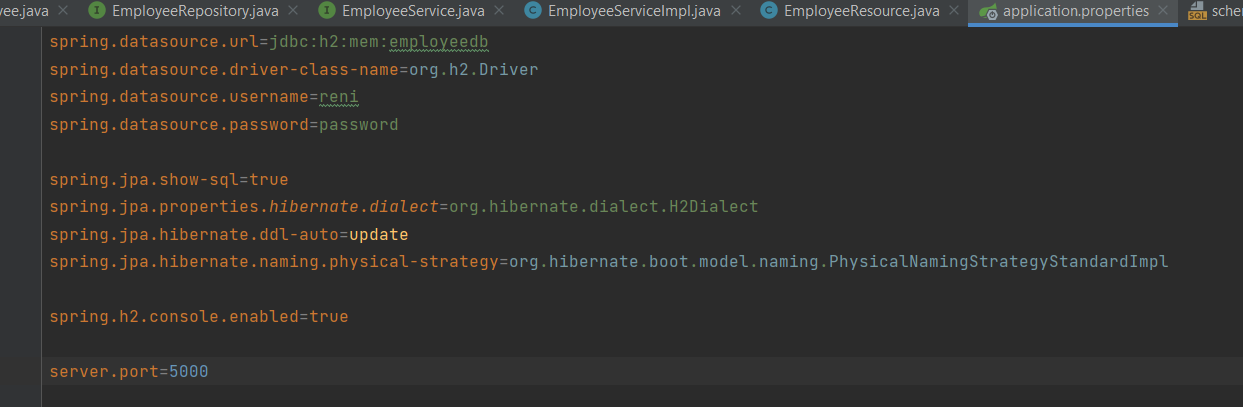
Description automatically generated

**Steps**

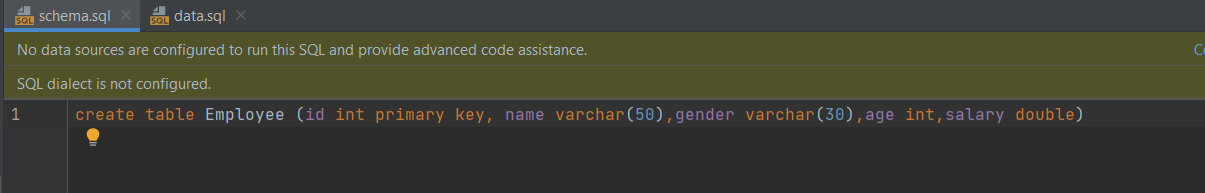
1. Create a Spring boot application with the dependencies shown in the screen below.

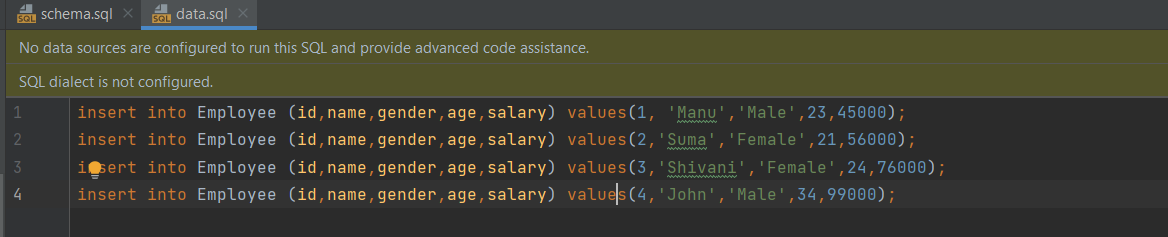


1. Update the “application.properties” with the following configurations.



1. Create “schema.sql” and “data.sql” file in resources folder with the following create and insert statements.

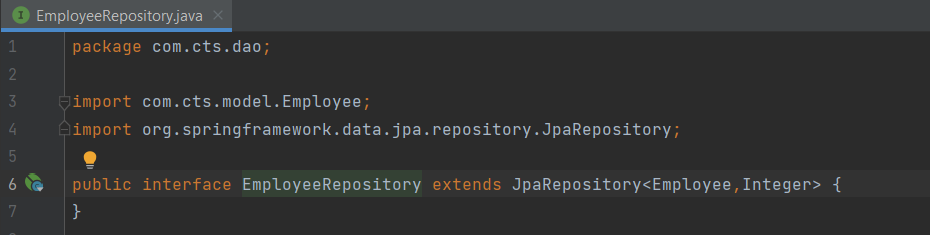




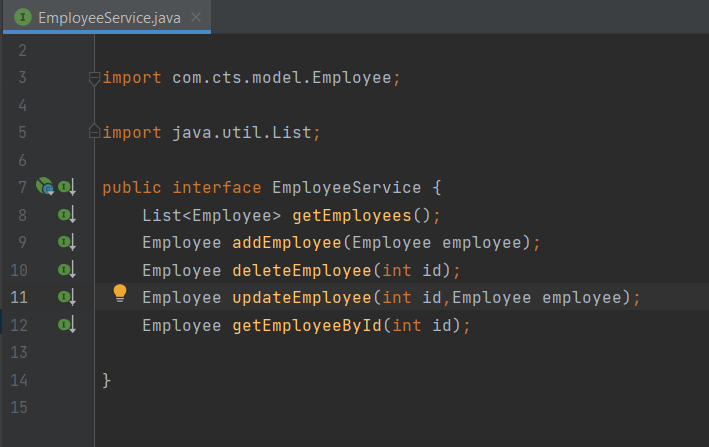
1. Create a model class “Employee” as shown below.



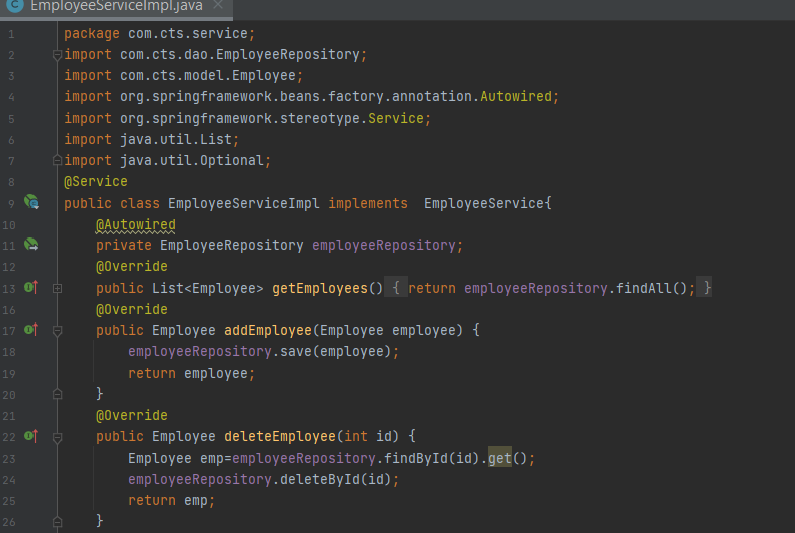
1. Create a repository class like the one shown below.

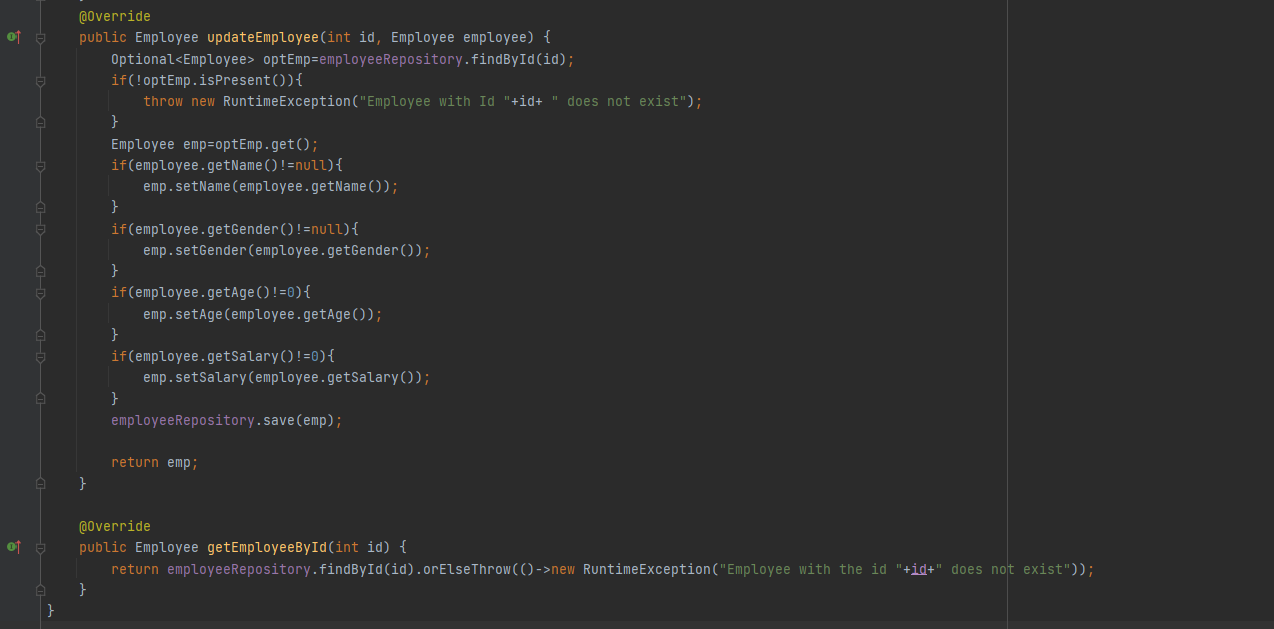


1. Create a service interface as shown below.



1. Create service implementation class as shown below.





1. Create an Employee Resource class as shown below.



1. Compile and Package the application and deploy the application to AWS Elastic Beanstalk.
2. Create an React Application by issuing the following command in the command prompt.

**Create-react-app emp-client**

1. Go to the application folder (emp-app) and issue the below mentioned commands.

**Npm install bootstrap react-bootstrap –save**

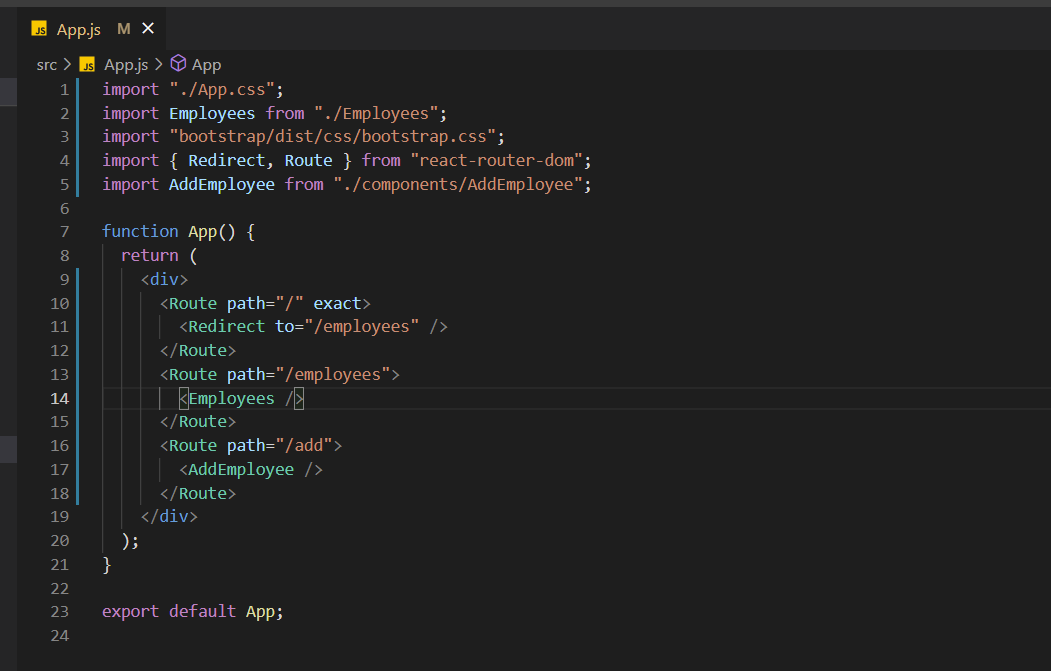
**Npm install axios –save**

**Npm install react-router-dom --save**

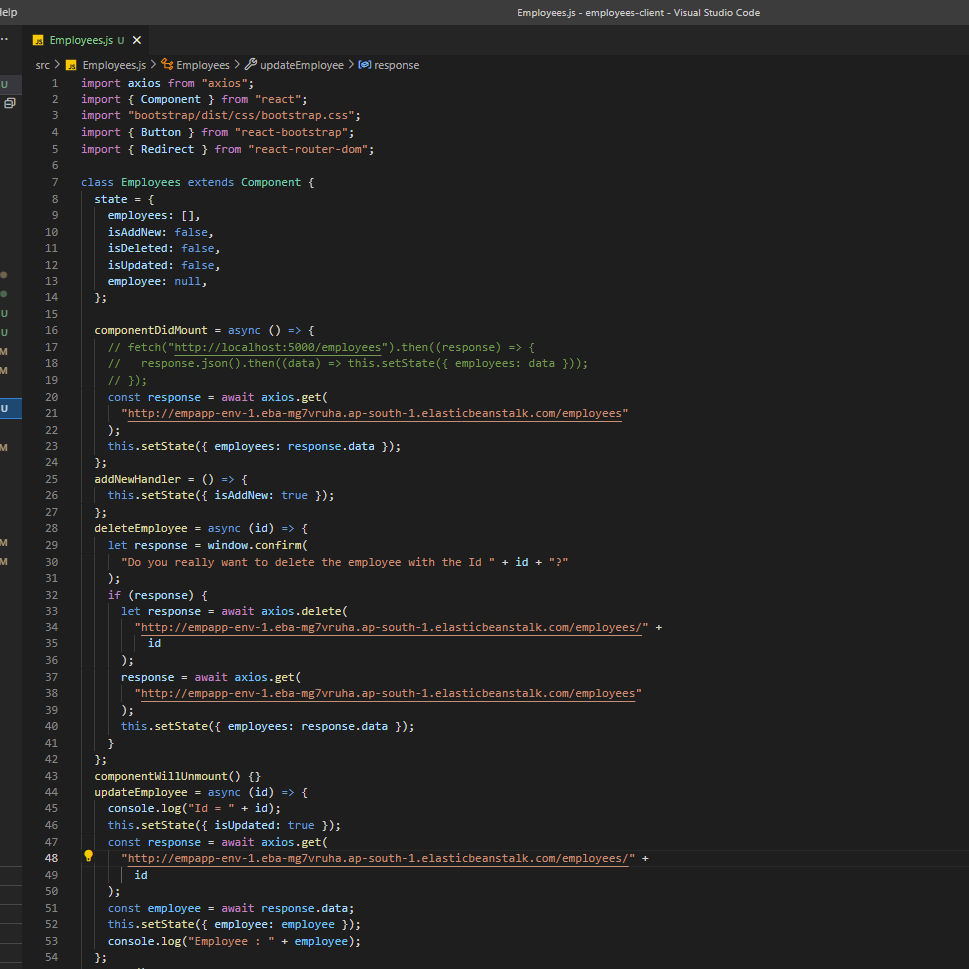
1. Modify index.js as shown below.

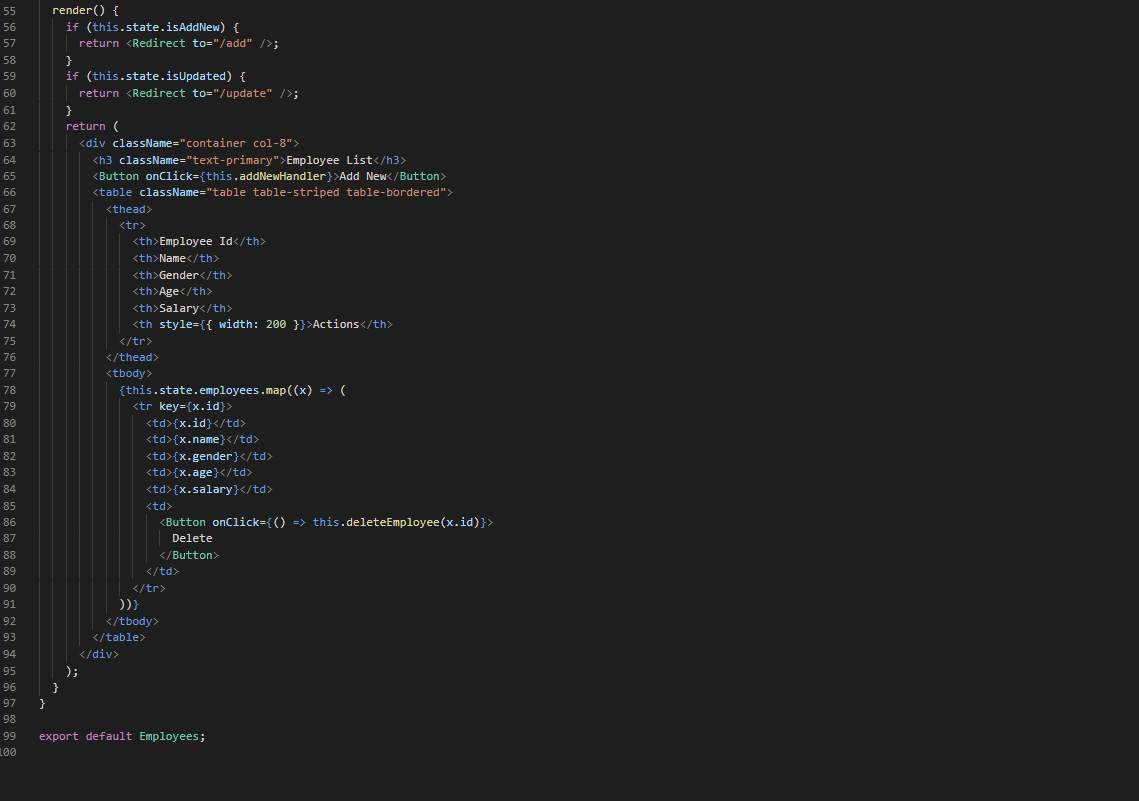


1. Modify App.js as shown below.

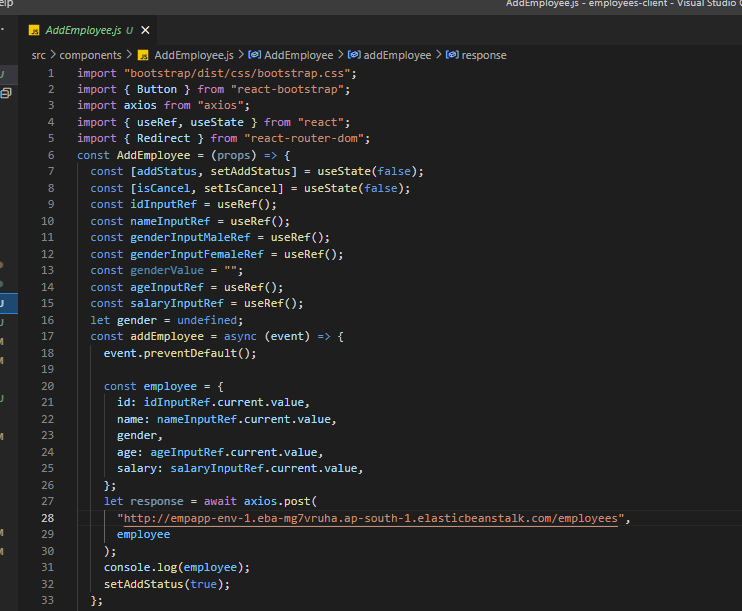


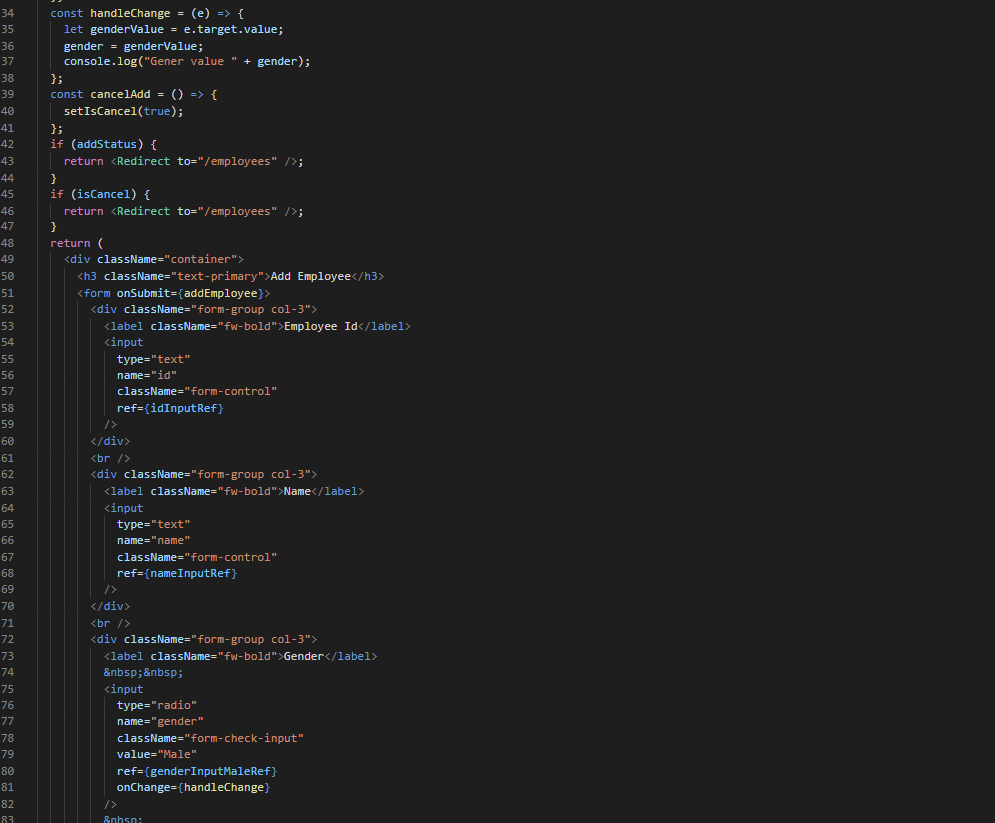
1. Create a component “Employee” in Employee.js and include the following code.

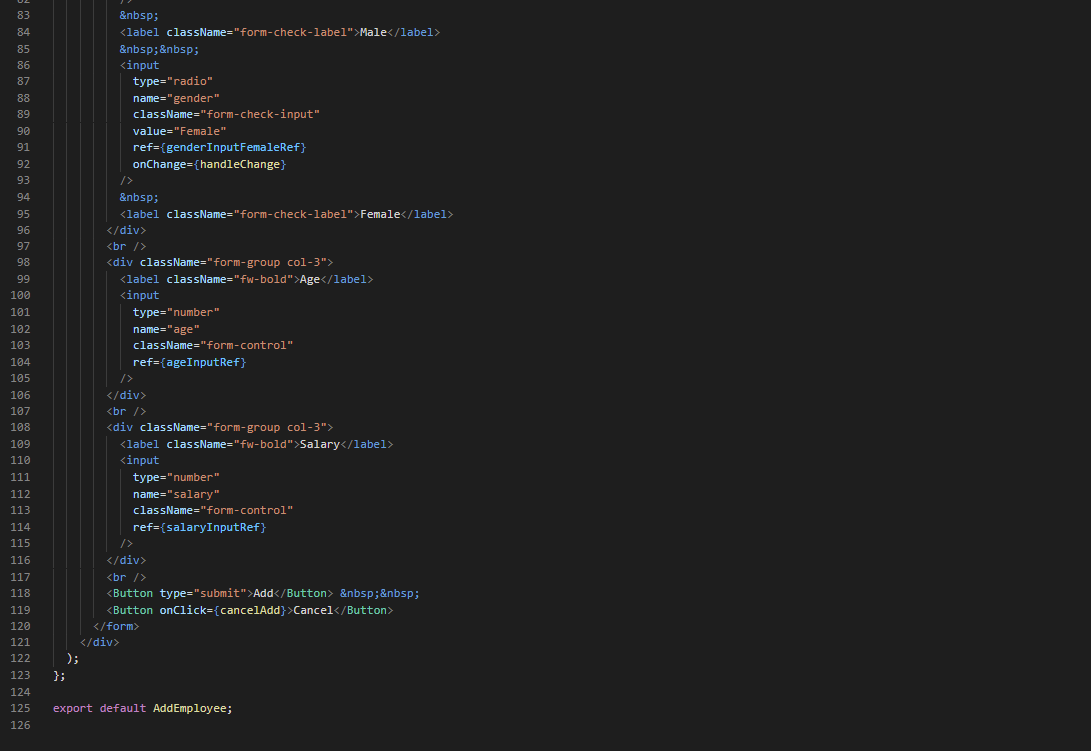




1. Create an AddEmployee component in AddEmployee.js and include the following code.







1. Run the react application by typing the following command

Npm start

1. Test the application for all the functionalities.
2. Make sure that the AWS Elastic Beanstalk is released once you are done with the application.