Class Activity 10

1. In the src/Features folder, update the StudentSlice.js with the following requirements.
2. Create a slice with name students and define the following as initial values in redux store.
   1. An empty array to store courses objects.
   2. status with "idle" as initial value.
   3. studentId, semester, academicYear and msg as null.
3. Create a reducer function named "addCourses" which is used to add the selected courses along with student id, semester and academic year to state.
4. In the src/components folder, the CourseRegistration component enables users to enter the name, semester details and to choose the courses for the semester. When user clicks on ADD button, dispatch an action to add all the information entered in the textboxes and chosen courses in the state.
5. In the src/components folder, the RegistrationSummary component is to display the registration summary as shown below.
6. Declare a selector variable to get the courses registration details from redux state.
7. Display the registration summary as shown in the screen. Make sure that same course should not be added more than one time in the selected course list.
8. In the same component RegistrationSummary component, when the user clicks on REGISTER button, dispatch an action to insert registration data to Redux thunk for sending data to server to insert MongoDB collection.
9. In the StudentSlice, define the registerCourses thunk to send the data received through the action from RegistrationSummary component to insert into the CourseRegInfos collection. Receive the server response and return as a message and do the following in the extra reducer.
   1. Add the pending, fulfilled and rejected extra reducers.
   2. When the action is fulfilled, change the status to success and update the msg value as server response.
   3. When the action is rejected, change the status to failed and update the msg value as server response.
10. In serverfolder, update the index.jsfile to create a REST API method to receive the course registration data from the client and insert it into the CourseRegInfos collection.

**Expected Output:**

