

studentCollegeFest

Above project uses Spring MVC along with Hibernate ORM to perform basic CRUD operations on College Fest Registration Portal.

The model *Student* has 3 visible as well as 1 hidden attribute.

- They are ->
1. *Id* (hidden)
 2. *Name*
 3. *Department*
 4. *Country*

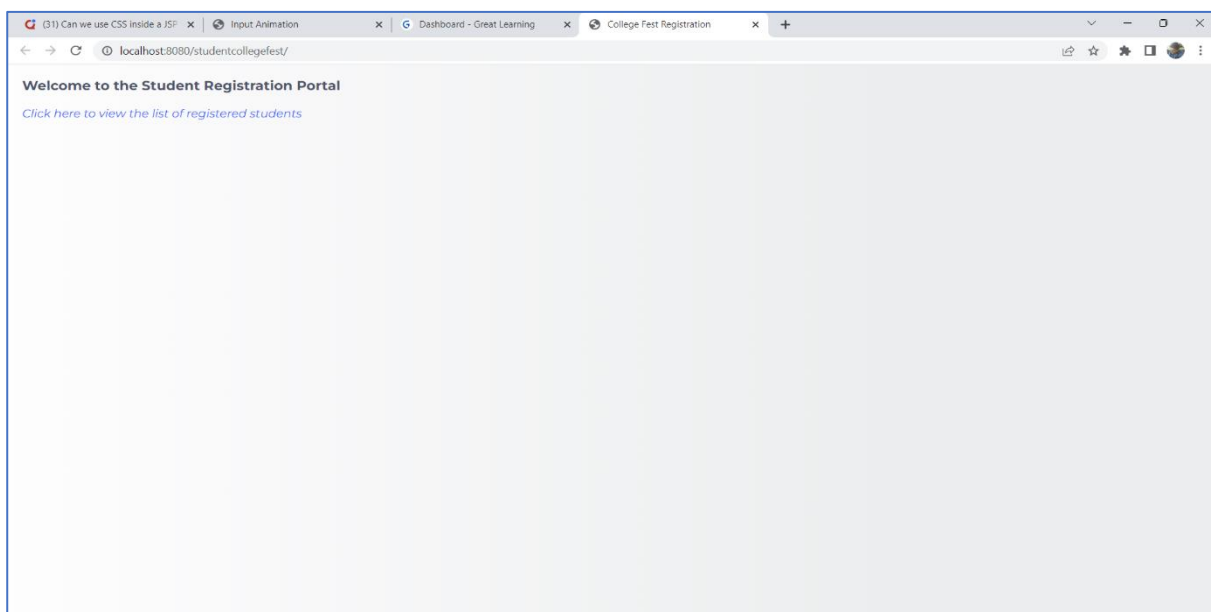
The application performs following functions ->

1. **Create** a new Student record
2. **Read** and display existing records
3. **Update** existing records and save any changes
4. **Delete** from existing records

We shall further see each of these functionalities in greater detail

The Main page

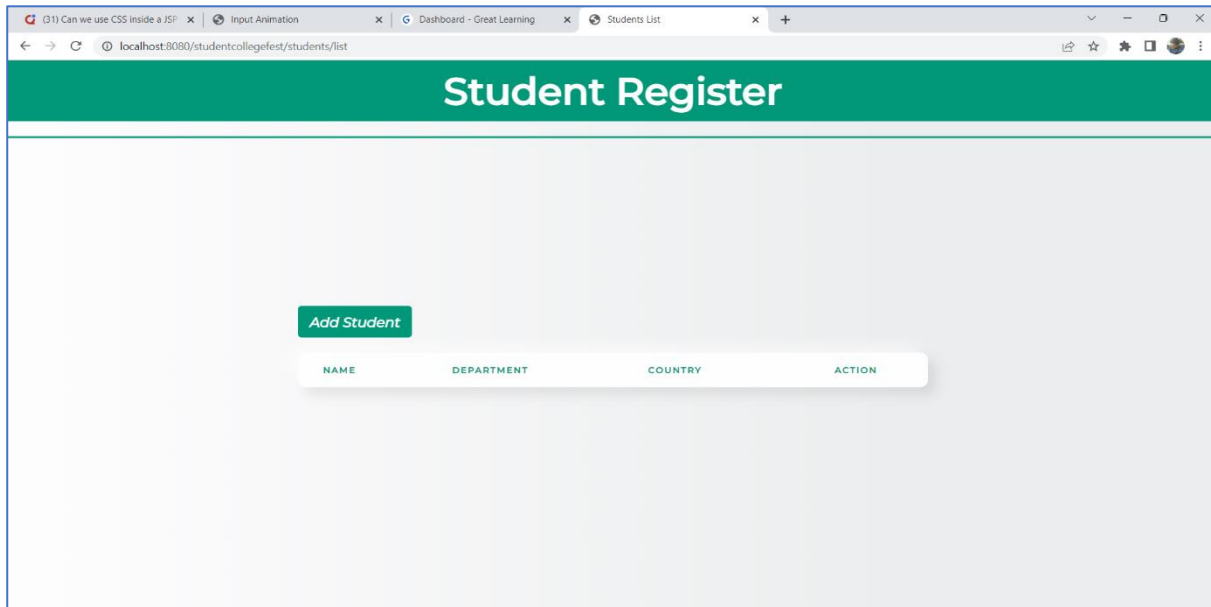
This is the default page that gets loaded when the application is booted on Tomcat server. It has a welcome message and link to **Student** list page.



Student List page

Upon clicking on before mentioned link, we arrive at Student list page. This page contains a header, a table and an **Add Student** button.

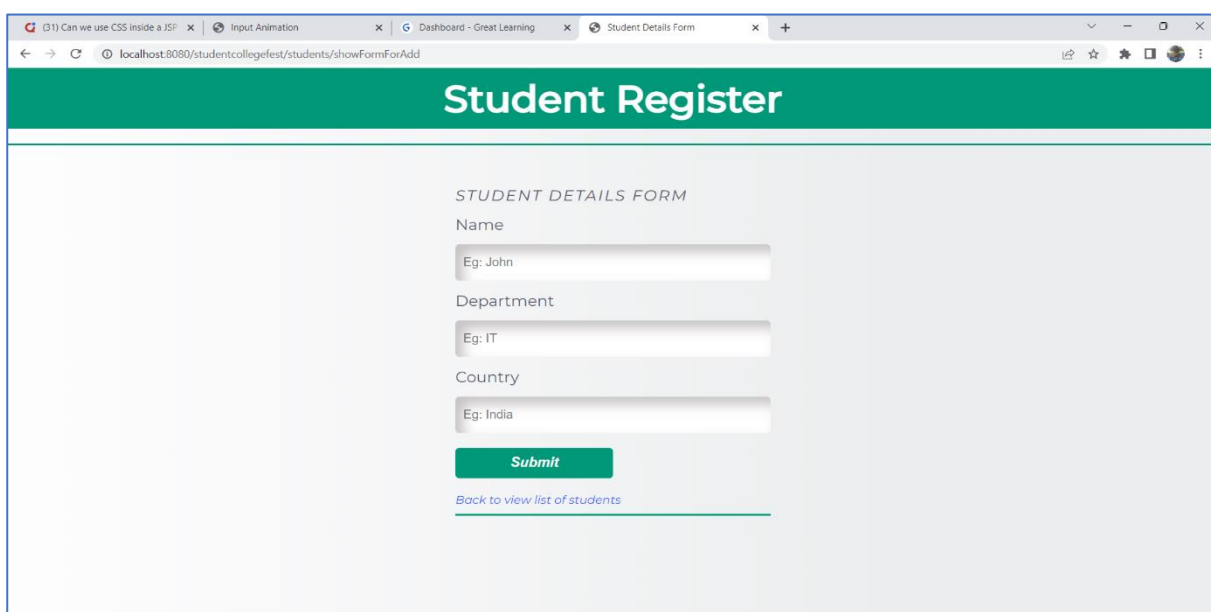
There are no records to be shown yet.



The table has 4 columns. First three of which are customer attributes whereas the fourth column is an action column. This column will display the action buttons viz. **Update** and **Delete**.

Clicking on **Add Student** button takes us to student form page.

Student Form page



This page has a header row displaying text “Student Details Form”. Below it are the 3 fields along with their labels followed by the **Submit** button

Student Register

STUDENT DETAILS FORM

Name
Sohan

Department
Mechanical

Country
India

Submit

[Back to view list of students](#)

The submit button and text fields have Neumorphic design. The **Submit** button changes its appearance when cursor hovers over it.

Student Register

STUDENT DETAILS FORM

Name
Sohan

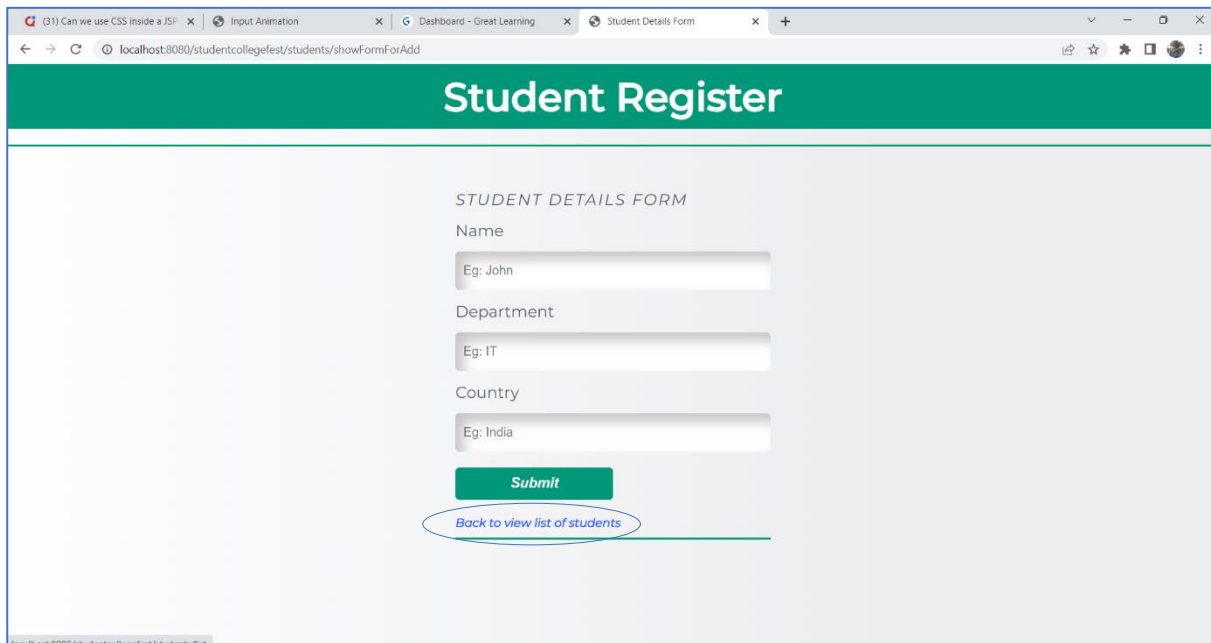
Department
Mechanical

Country
India

Submit

[Back to view list of students](#)

At the very end is provided a link to go back to student list page.



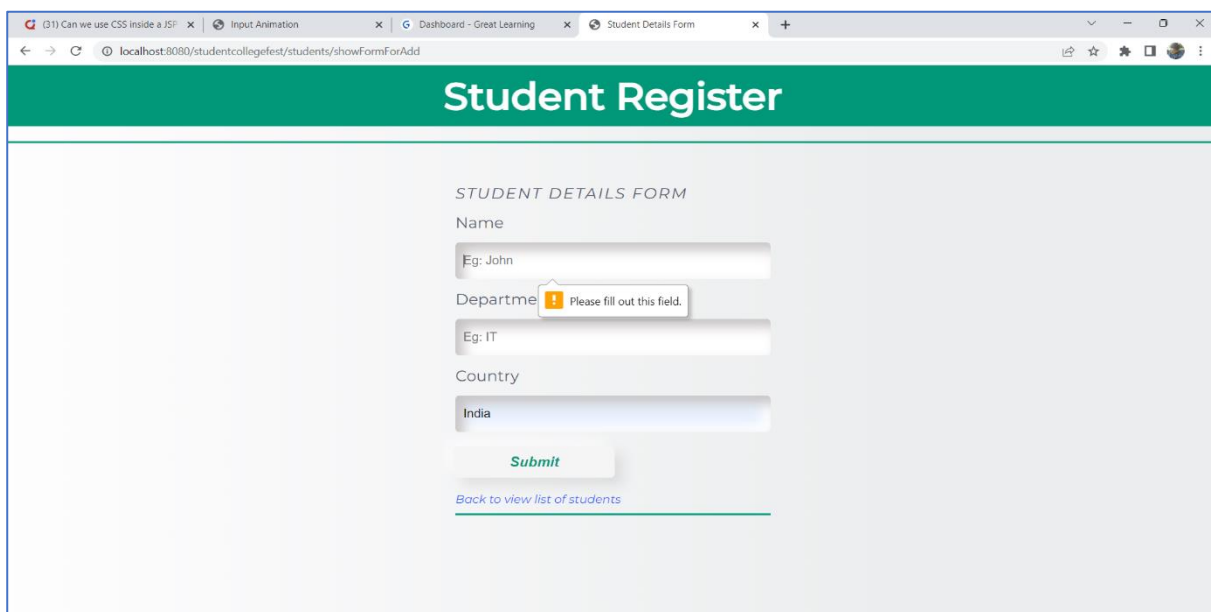
The screenshot shows a web browser window with the URL `localhost:8080/studentcollegefest/students/showFormForAdd`. The page has a green header with the text "Student Register". Below the header is a form titled "STUDENT DETAILS FORM". The form contains four input fields: "Name" (with placeholder text "Eg: John"), "Department" (with placeholder text "Eg: IT"), "Country" (with placeholder text "Eg: India"), and a "Submit" button. Below the "Submit" button is a link that says "Back to view list of students".

On clicking the **Submit** button, the newly entered records are displayed on student list page.

#Note->

The Name and Department fields are required fields which have to be populated before submitting the details.

*A warning message displays when we click on **Submit** button without entering data as per convention in those fields.*



The screenshot shows the same web browser window as before, but now the "Department" field has a validation error. A yellow warning message box with an exclamation mark icon is displayed next to the "Department" input field, containing the text "Please fill out this field." The "Name" field is also populated with "John". The "Country" field is populated with "India". The "Submit" button is still visible, and the "Back to view list of students" link is at the bottom.

Name is a required field

The screenshot shows a web browser window with the URL `localhost:8080/studentcollegefest/students/showFormForAdd`. The page has a green header with the text "Student Register". Below the header, the form is titled "STUDENT DETAILS FORM". It contains three input fields: "Name" with the value "Ravi", "Department" with the value "Eg: IT", and "Country" with the value "India". A red error message "Please fill out this field." is displayed next to the Department field. Below the fields is a green "Submit" button and a link "Back to view list of students".

STUDENT DETAILS FORM

Name

Ravi

Department

Eg: IT

Country

Please fill out this field.

India

Submit

[Back to view list of students](#)

Department is a required field

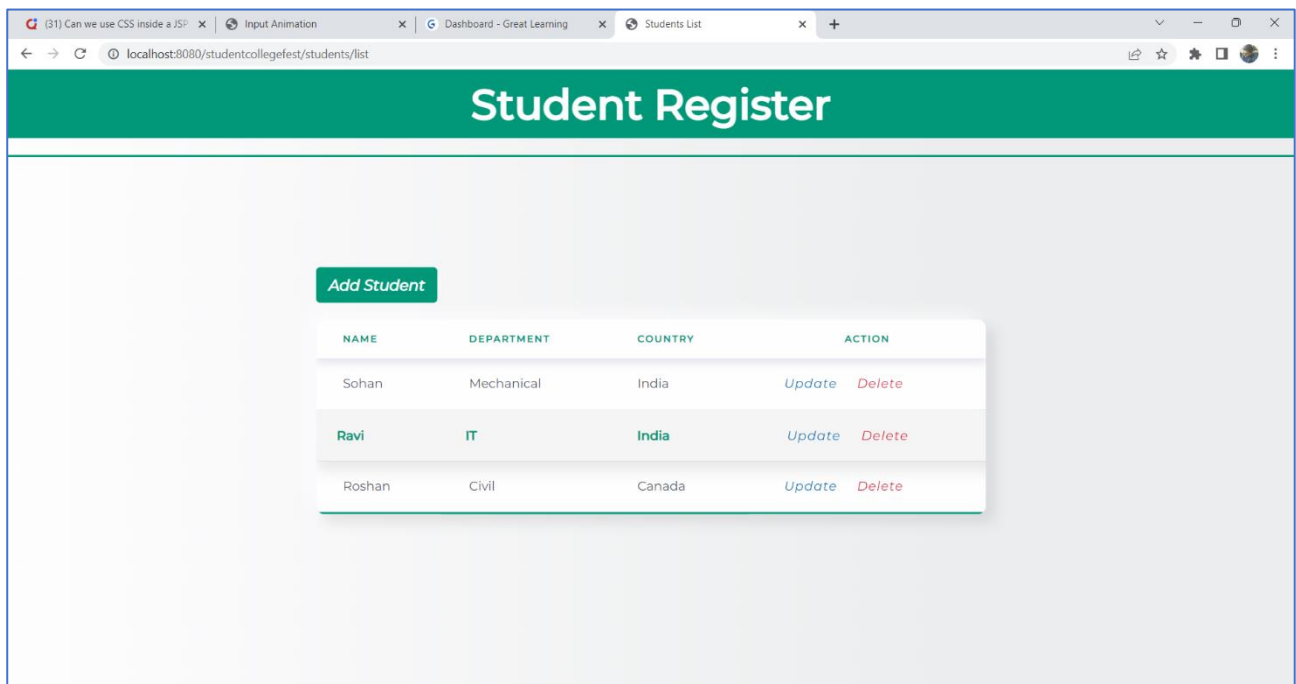
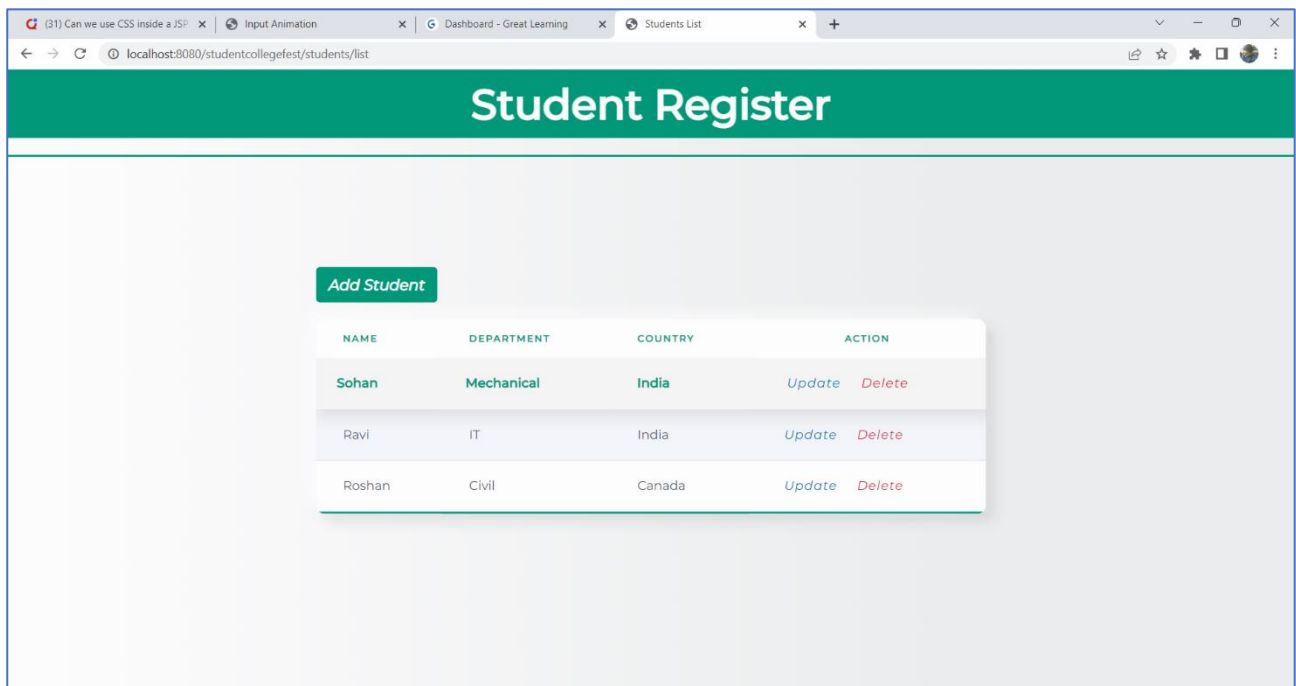
Student table

The entered records get displayed on student list page.

The screenshot shows a web browser window with the URL `localhost:8080/studentcollegefest/students/list`. The page has a green header with the text "Student Register". Below the header, there is a green "Add Student" button and a table displaying student records.

NAME	DEPARTMENT	COUNTRY	ACTION
Sohan	Mechanical	India	Update Delete

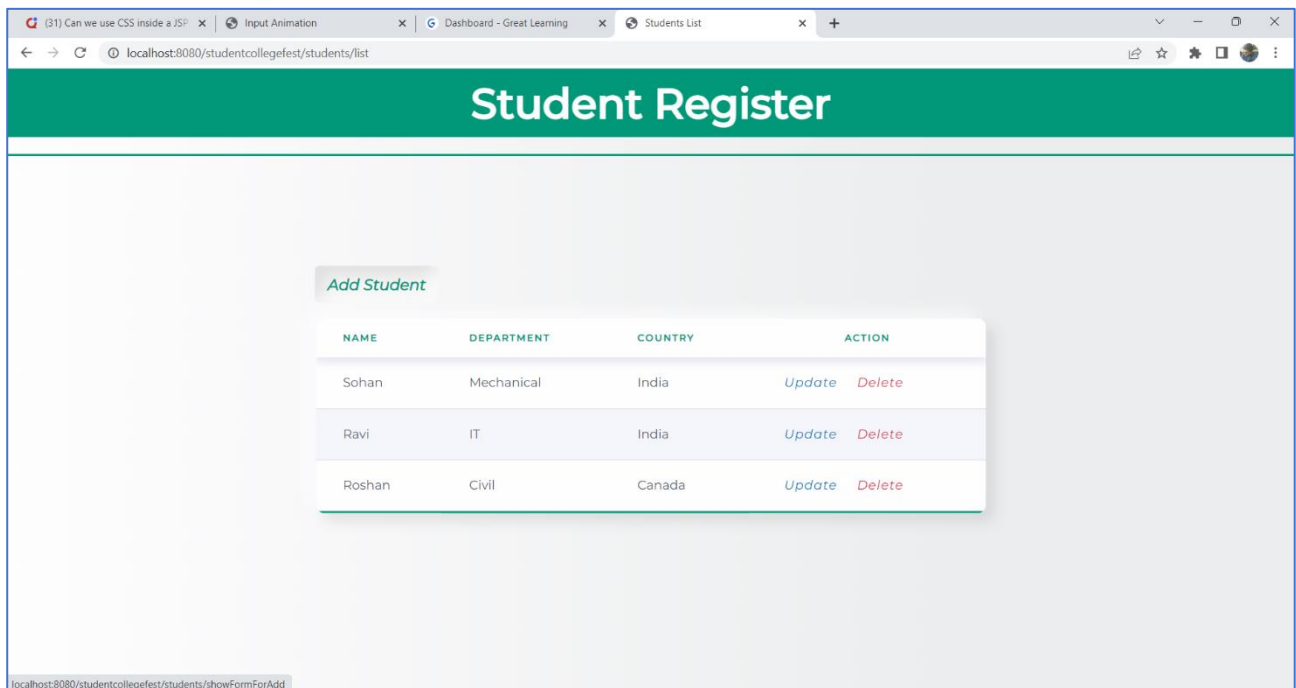
User can further add more records.



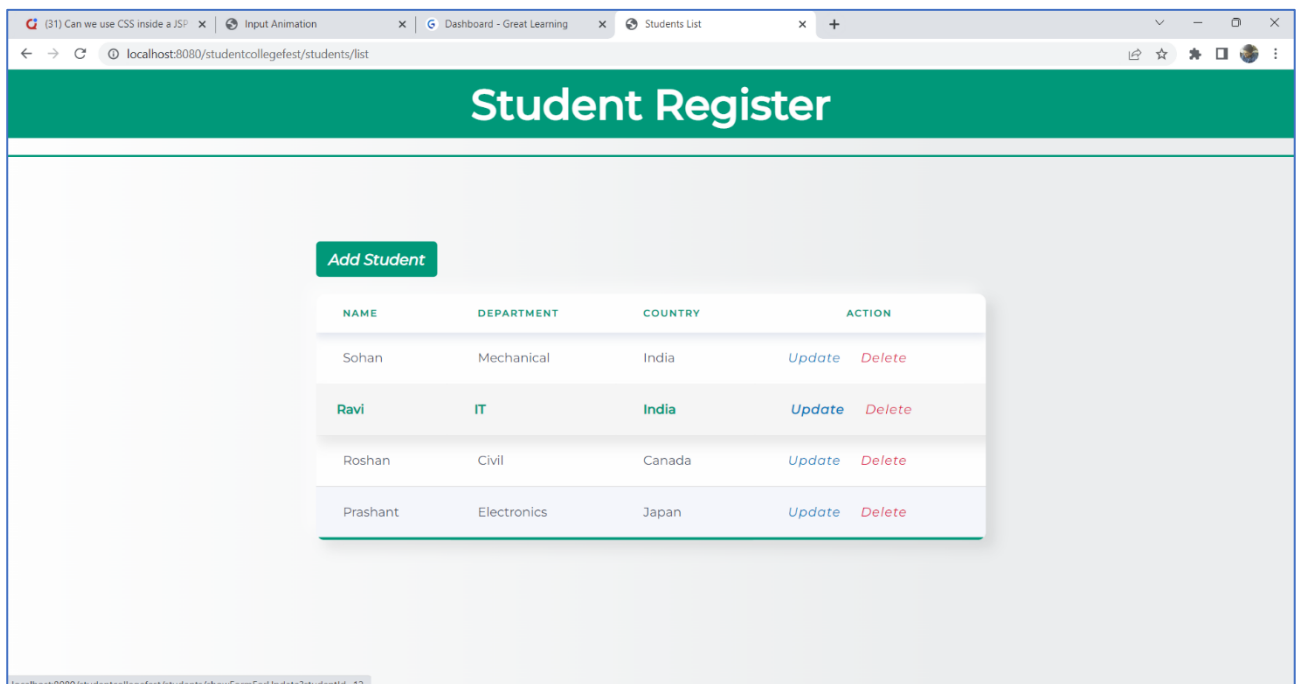
Both the **Add Student** and table body rows change appearance when cursor hovers on them.

While the rows get slightly enlarged (by 2%) and the inline text changes in colour, size and thickness.

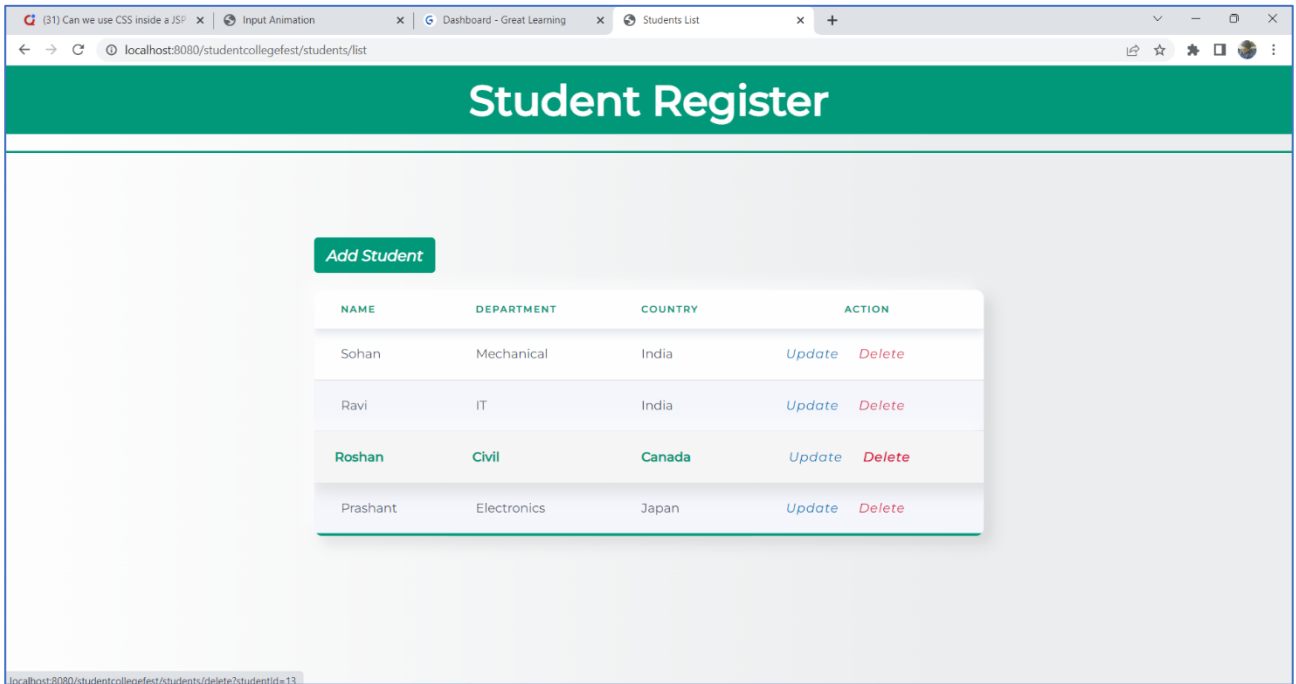
Whereas the **Add Student** button takes appearance of an inset button.



Similarly, the **Update** and **Delete** hyperlinks increase in font-weight when the cursor hovers on them.



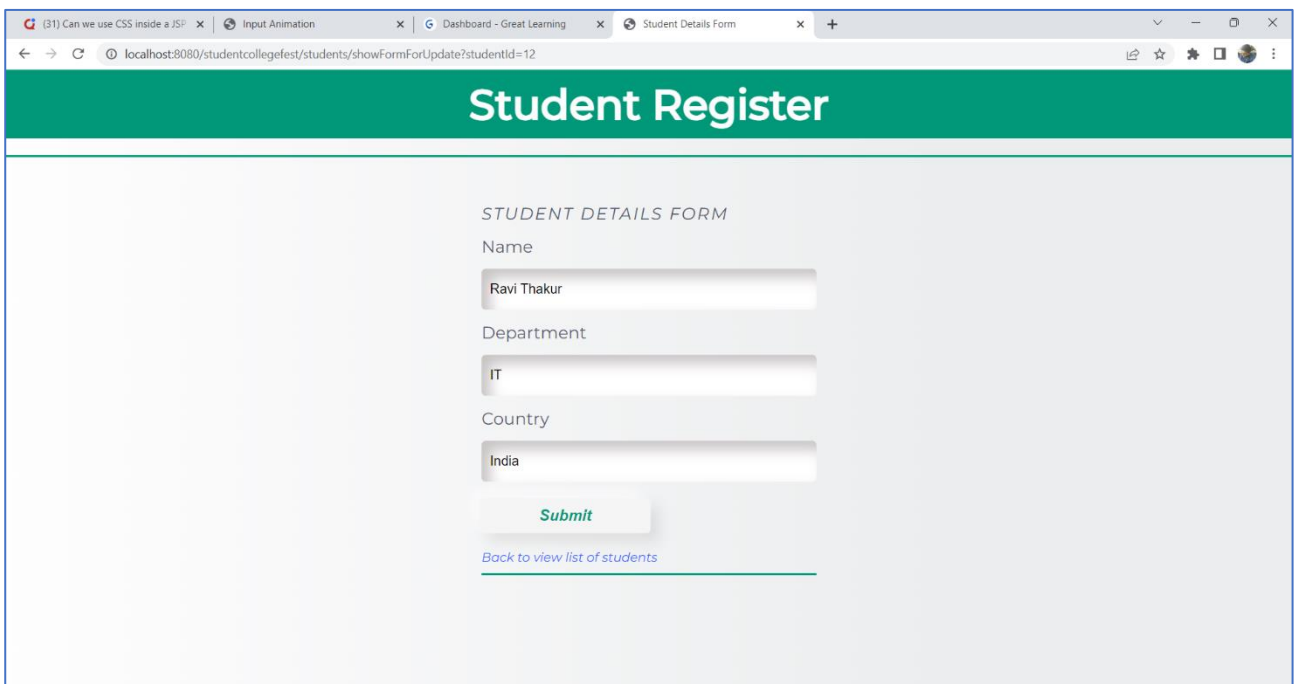
Update link hover



Delete link hover

On clicking the **Update** link, the “Student details” form gets loaded, pre-populated with existing data.

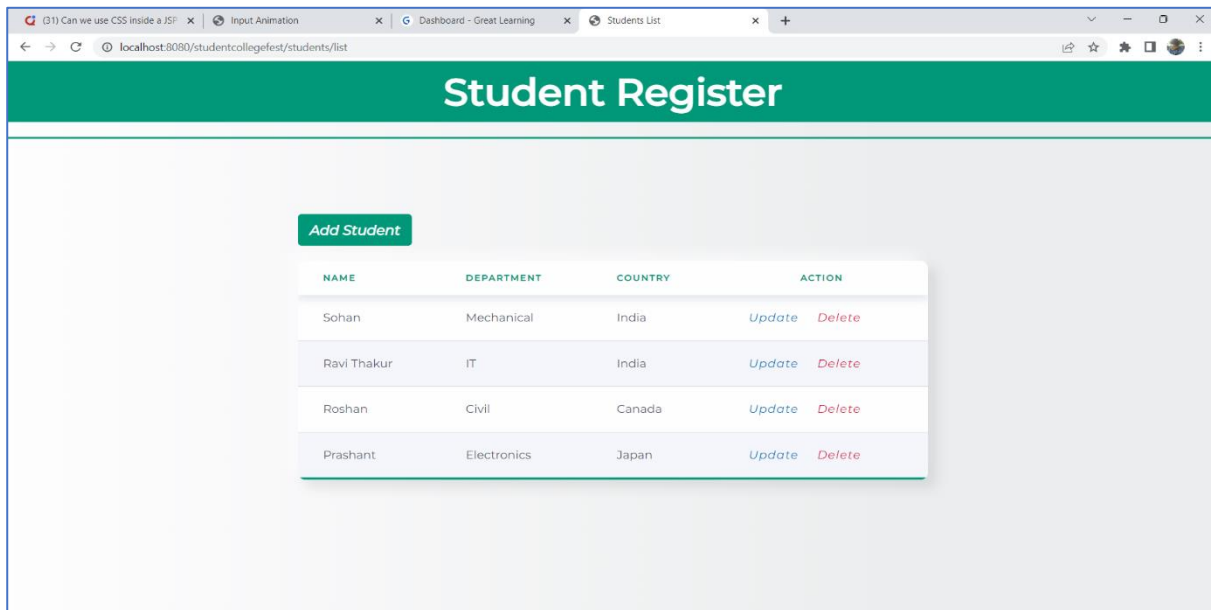
User can make necessary changes and click on **Submit** button.



After clicking on **Submit** button. The updated record gets displayed on student list page.

Eg: Name *Ravi* is updated to *Ravi Thakur*

Here we see the record has been updated!



Finally, on clicking the **Delete** link, a message pops up on the page asking to confirm whether to delete the concerned record from database.

