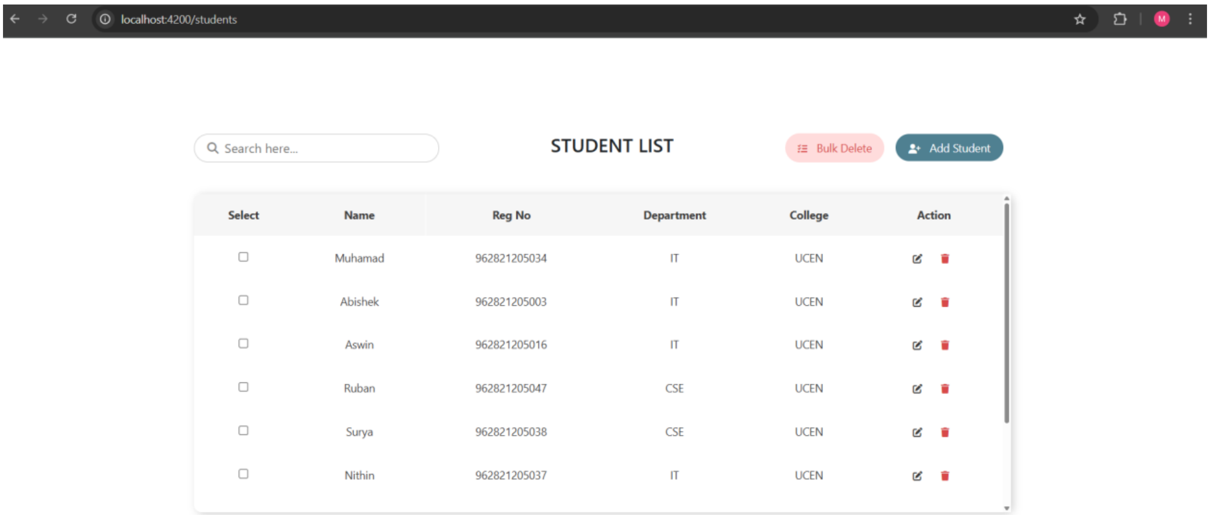
OUTPUT:

ember-vertical-collection

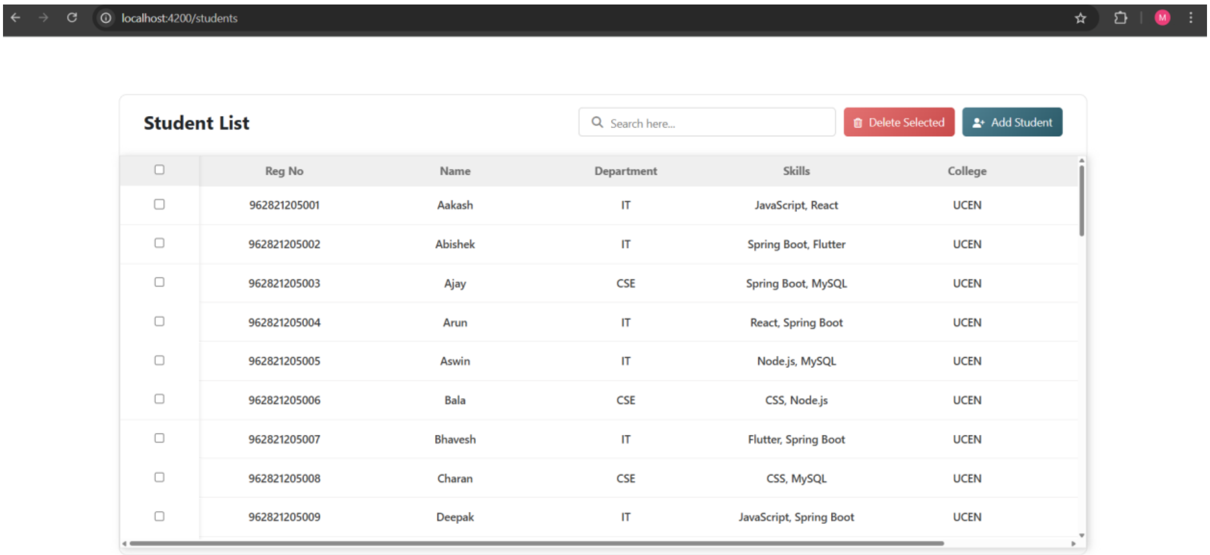
As a table :



STUDENT LIST

Select	Name	Reg No	Department	College	Action
□	Muhamad	962821205034	IT	UCEN	
□	Abishek	962821205003	IT	UCEN	
□	Aswin	962821205016	IT	UCEN	
□	Ruban	962821205047	CSE	UCEN	
□	Surya	962821205038	CSE	UCEN	
□	Nithin	962821205037	IT	UCEN	

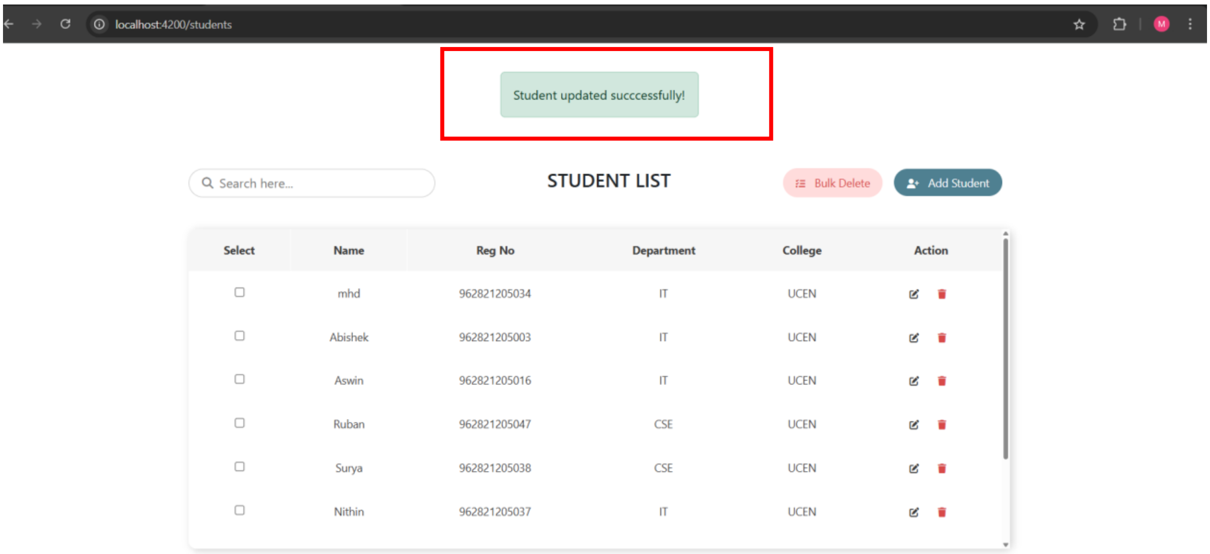
Bidirectional scrollable div and scrollable body:



Student List

□	Reg No	Name	Department	Skills	College
□	962821205001	Aakash	IT	JavaScript, React	UCEN
□	962821205002	Abishek	IT	Spring Boot, Flutter	UCEN
□	962821205003	Ajay	CSE	Spring Boot, MySQL	UCEN
□	962821205004	Arun	IT	React, Spring Boot	UCEN
□	962821205005	Aswin	IT	Node.js, MySQL	UCEN
□	962821205006	Bala	CSE	CSS, Node.js	UCEN
□	962821205007	Bhavesh	IT	Flutter, Spring Boot	UCEN
□	962821205008	Charan	CSE	CSS, MySQL	UCEN
□	962821205009	Deepak	IT	JavaScript, Spring Boot	UCEN

ember-cli-flash



localhost:4200/students

Student updated succcessfully!

Search here...

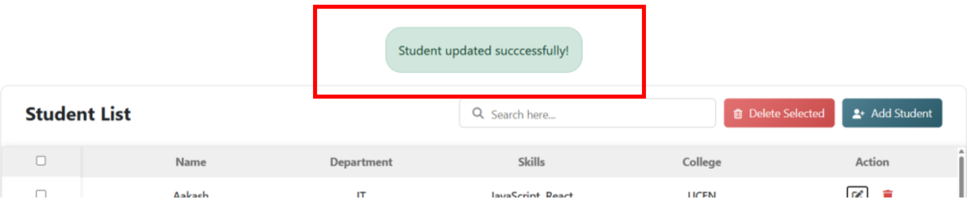
STUDENT LIST

Bulk Delete

Add Student

Select	Name	Reg No	Department	College	Action
□	mhd	962821205034	IT	UCEN	✎ 🗑
□	Abishek	962821205003	IT	UCEN	✎ 🗑
□	Aswin	962821205016	IT	UCEN	✎ 🗑
□	Ruban	962821205047	CSE	UCEN	✎ 🗑
□	Surya	962821205038	CSE	UCEN	✎ 🗑
□	Nithin	962821205037	IT	UCEN	✎ 🗑

The style of the cli-flash is customized



Student updated succcessfully!

Student List

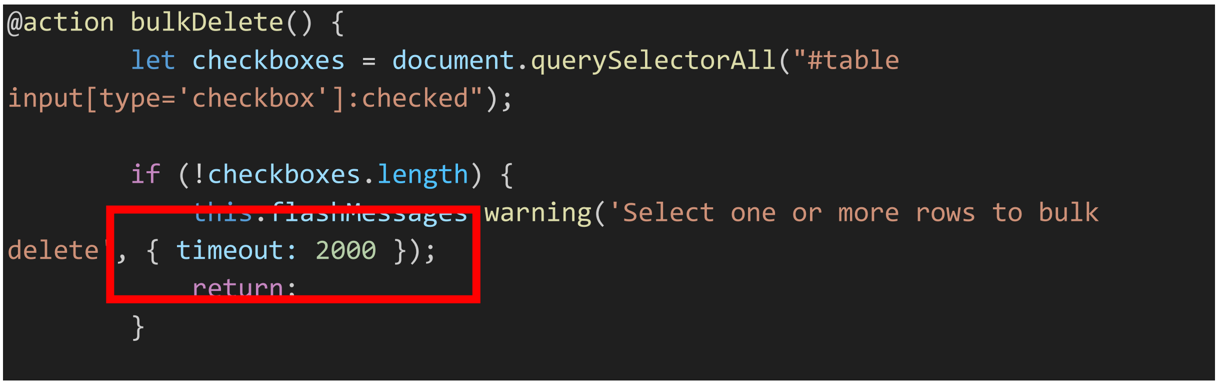
Search here...

Delete Selected

Add Student

□	Name	Department	Skills	College	Action
□	Aakash	IT	JavaScript, React	UCEN	✎ 🗑

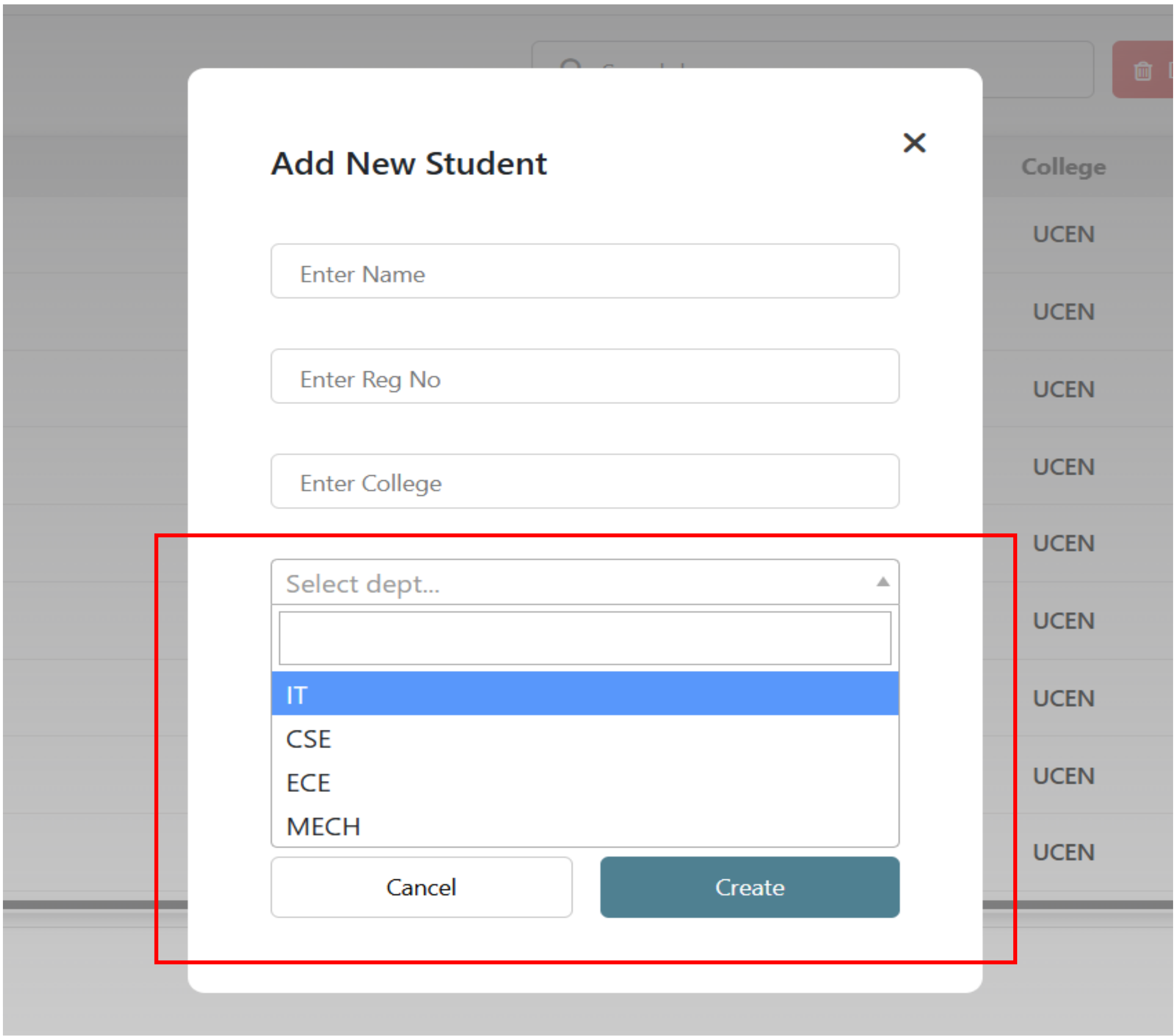
Timeout is set manually to hide the flash message



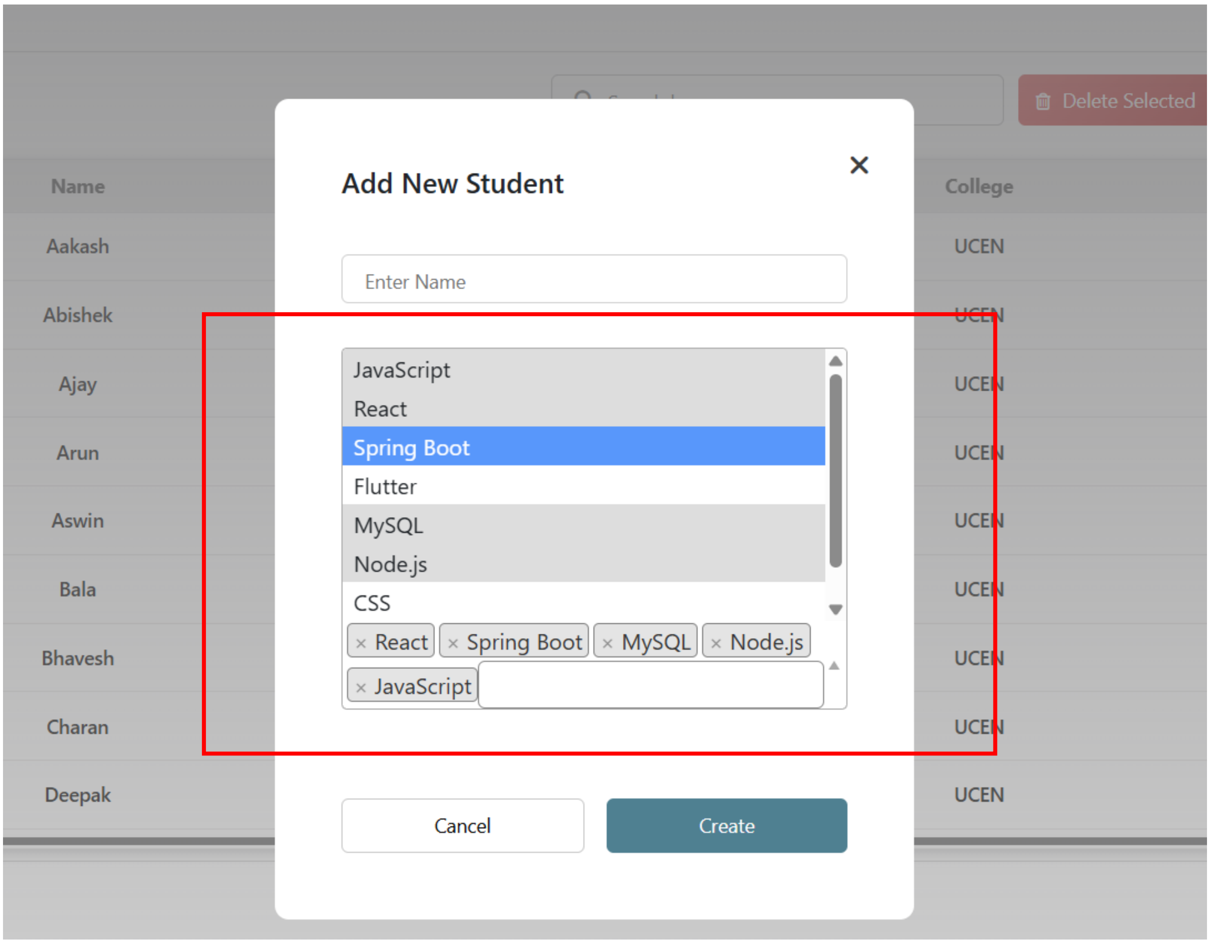
```
@action bulkDelete() {
        let checkboxes = document.querySelectorAll("#table
input[type='checkbox']:checked");

        if (!checkboxes.length) {
            this.flashMessages.warning('Select one or more rows to bulk
delete', { timeout: 2000 });
            return;
        }
```

ember-power-select

1.power select



2.power select multiple



Name	College
Aakash	UCEN
Abishek	UCEN
Ajay	UCEN
Arun	UCEN
Aswin	UCEN
Bala	UCEN
Bhavesh	UCEN
Charan	UCEN
Deepak	UCEN

index.html

```
<!DOCTYPE html>

<html>

 <head>

  <meta charset="utf-8">

  <title>Ember-Student-List</title>

  <meta name="description" content="">

  <meta name="viewport" content="width=device-width, initial-scale=1">

  {{content-for "head"}}

  <link integrity="" rel="stylesheet" href="{{rootURL}}assets/vendor.css">

  <link integrity="" rel="stylesheet" href="{{rootURL}}assets/ember-student-list.css">
```

```
    <link rel="stylesheet" href="https://cdnjs.cloudflare.com/ajax/libs/font-awesome/6.7.2/css/all.min.css" integrity="sha512-Evv84Mr4kqVGRNSgIGL/F/aIDqQb7xQ2vcrdIwxfjThSH8CSR7PBEakCr51Ck+w+/U6swU2Im1vVX0SVk9ABhg==" crossorigin="anonymous" referrerpolicy="no-referrer" />
    <link rel="stylesheet" href="https://cdn.jsdelivr.net/npm/bootstrap@5.3.0/dist/css/bootstrap.min.css">

    {{content-for "head-footer"}}
  </head>
  <body>
    {{content-for "body"}}

    <script src="{{rootURL}}assets/vendor.js"></script>
    <script src="{{rootURL}}assets/ember-student-list.js"></script>

    {{content-for "body-footer"}}
  </body>
</html>
```

templates/application.hbs

```
{{outlet}}
<BasicDropdownWormhole />
```

templates/students.hbs

```
{{#each this.flashMessages.queue as |flash|}}
<FlashMessage @flash={{flash}} @messageStyle='bootstrap' />
{{/each}}
```

```
<div  class="{{if (or (eq this.currentRoute 'students.add') (eq this.currentRoute
'students.edit')) 'full-screen-outlet' ''}}">
  {{liquid-outlet}}
</div>


<div class="student-container">
  <div class="search-dlt-cont">
    <span>Student List</span>
    <div class="search-cd-cont">
      <div class="search-cont">
        <i class="fa-solid fa-magnifying-glass"></i>
        <input type="text" id="search" placeholder="Search here..." {{on "keyup"
this.handleSearch}}>
      </div>
      <button {{on "click" this.bulkDelete}} class="btn-cd dlt" {{on-key 'Delete'
this.bulkDelete}}>
        <i class="fa-regular fa-trash-can"></i> Delete Selected
      </button>
      <LinkTo @route="students.add" class="btn-cd add" {{on-key 'alt+a' this.addStud}}>
        <i class="fa-solid fa-user-plus"></i> Add Student
      </LinkTo>
    </div>
  </div>
  <StudentList @students={{this.students.filteredStudents}} @deleteRow={{this.deleteRow}}
@editRow={{this.editRow}}
    @updateSelection={{this.updateSelection}} @toggleSelectAll={{this.toggleSelectAll}}
    @selectAll={{this.selectAll}} />
</div>
```

templates/students/add.hbs

```
<div id="overlay"></div>
<div {{on-click-outside this.closeForm}}>
   <StudentForm @addRow={{this.addRow}} @closeForm={{this.closeForm}} />
</div>
```

templates/students/edit.hbs

```
<div id="overlay"></div>
<div {{on-click-outside this.closeForm}}>
  <StudentForm @editStudent={{this.model}} @students={{this.students.students}} @saveStudent={{this.saveStudent}}
   @updateStudents={{this.updateStudents}} @closeForm={{this.closeForm}} />
</div>
```

services/students.js

```
import Service from '@ember/service';
import { tracked } from '@glimmer/tracking';

export default class StudentsService extends Service {
   @tracked students = [
      { name: "Aakash", reg: 962821205001, dept: "IT", clg: "UCEN", skills: ["JavaScript", "React"] },
      { name: "Abishek", reg: 962821205002, dept: "IT", clg: "UCEN", skills: ["Spring Boot", "Flutter"] },
      { name: "Ajay", reg: 962821205003, dept: "CSE", clg: "UCEN", skills: ["Spring Boot", "MySQL"] },
      { name: "Arun", reg: 962821205004, dept: "IT", clg: "UCEN", skills: ["React", "Spring Boot"] },
      { name: "Aswin", reg: 962821205005, dept: "IT", clg: "UCEN", skills: ["Node.js", "MySQL"] },
      { name: "Bala", reg: 962821205006, dept: "CSE", clg: "UCEN", skills: ["CSS", "Node.js"] },
      { name: "Bhavesh", reg: 962821205007, dept: "IT", clg: "UCEN", skills: ["Flutter", "Spring Boot"] },
      { name: "Charan", reg: 962821205008, dept: "CSE", clg: "UCEN", skills: ["CSS", "MySQL"] },
      { name: "Deepak", reg: 962821205009, dept: "IT", clg: "UCEN", skills: ["JavaScript", "Spring Boot"] },
      { name: "Dev", reg: 962821205010, dept: "IT", clg: "UCEN", skills: ["React", "HTML"] },
      { name: "Dinesh", reg: 962821205011, dept: "IT", clg: "UCEN", skills: ["JavaScript", "HTML"] },
      { name: "Ethan", reg: 962821205012, dept: "CSE", clg: "UCEN", skills: ["Node.js", "JavaScript"] },
      { name: "Elavarasan", reg: 962821205013, dept: "IT", clg: "UCEN", skills: ["HTML", "React"] },
      { name: "Faiz", reg: 962821205014, dept: "IT", clg: "UCEN", skills: ["Spring Boot", "CSS"] },
      { name: "Gokul", reg: 962821205015, dept: "CSE", clg: "UCEN", skills: ["MySQL", "Flutter"] },
      { name: "Guhan", reg: 962821205016, dept: "CSE", clg: "UCEN", skills: ["Flutter", "React"] },
      { name: "Hari", reg: 962821205017, dept: "CSE", clg: "UCEN", skills: ["CSS", "HTML"] },
      { name: "Harish", reg: 962821205018, dept: "IT", clg: "UCEN", skills: ["Node.js", "CSS"] },
      { name: "Harsha", reg: 962821205019, dept: "IT", clg: "UCEN", skills: ["MySQL", "HTML"] },
      { name: "Imran", reg: 962821205020, dept: "CSE", clg: "UCEN", skills: ["Node.js", "JavaScript"] },
      { name: "Irfan", reg: 962821205021, dept: "CSE", clg: "UCEN", skills: ["Spring Boot", "JavaScript"] },
      { name: "Jagan", reg: 962821205022, dept: "IT", clg: "UCEN", skills: ["React", "Spring Boot"] },
      { name: "Jeeva", reg: 962821205023, dept: "IT", clg: "UCEN", skills: ["HTML", "Node.js"]
  },
      { name: "Karthik", reg: 962821205024, dept: "IT", clg: "UCEN", skills: ["Flutter", "React"]
  },
      { name: "Keerthi", reg: 962821205025, dept: "IT", clg: "UCEN", skills: ["React", "CSS"] },
      { name: "Kiran", reg: 962821205026, dept: "IT", clg: "UCEN", skills: ["CSS", "Flutter"] },
      { name: "Kavin", reg: 962821205027, dept: "CSE", clg: "UCEN", skills: ["Spring Boot",
  "React"] },
      { name: "Lalith", reg: 962821205028, dept: "CSE", clg: "UCEN", skills: ["HTML",
  "MySQL"] },
      { name: "Lokesh", reg: 962821205029, dept: "CSE", clg: "UCEN", skills: ["Flutter",
  "MySQL"] },
      { name: "Manoj", reg: 962821205030, dept: "IT", clg: "UCEN", skills: ["JavaScript",
  "React"] },
      { name: "Muhamad", reg: 962821205031, dept: "IT", clg: "UCEN", skills: ["JavaScript",
  "React"] },
      { name: "Naveen", reg: 962821205032, dept: "CSE", clg: "UCEN", skills: ["Spring Boot",
  "Node.js"] },
      { name: "Nithin", reg: 962821205033, dept: "IT", clg: "UCEN", skills: ["Node.js", "CSS"] },
      { name: "Omprakash", reg: 962821205034, dept: "IT", clg: "UCEN", skills: ["CSS",
  "HTML"] },
      { name: "Pranav", reg: 962821205035, dept: "CSE", clg: "UCEN", skills: ["MySQL",
  "JavaScript"] },
      { name: "Praveen", reg: 962821205036, dept: "CSE", clg: "UCEN", skills: ["MySQL",
  "HTML"] },
      { name: "Raj", reg: 962821205037, dept: "IT", clg: "UCEN", skills: ["React", "Flutter"] },
      { name: "Ragul", reg: 962821205038, dept: "IT", clg: "UCEN", skills: ["Node.js", "Flutter"]
  },
      { name: "Ruban", reg: 962821205039, dept: "CSE", clg: "UCEN", skills: ["HTML", "CSS"] },
      { name: "Sanjay", reg: 962821205040, dept: "IT", clg: "UCEN", skills: ["Node.js", "Spring
  Boot"] },
      { name: "Sathish", reg: 962821205041, dept: "IT", clg: "UCEN", skills: ["JavaScript",
  "Node.js"] },
      { name: "Suriya", reg: 962821205042, dept: "IT", clg: "UCEN", skills: ["HTML",
  "JavaScript"] },
      { name: "Surya", reg: 962821205043, dept: "CSE", clg: "UCEN", skills: ["React", "Flutter"]
  },
      { name: "Tharun", reg: 962821205044, dept: "CSE", clg: "UCEN", skills: ["HTML", "CSS"]
  },
      { name: "Udhay", reg: 962821205045, dept: "IT", clg: "UCEN", skills: ["JavaScript",
  "MySQL"] },
      { name: "Venkatesh", reg: 962821205046, dept: "CSE", clg: "UCEN", skills: ["React",
  "Flutter"] },
      { name: "Vijay", reg: 962821205047, dept: "IT", clg: "UCEN", skills: ["Flutter", "MySQL"]
  },
      { name: "Vimal", reg: 962821205048, dept: "CSE", clg: "UCEN", skills: ["React", "CSS"] },
      { name: "Yogesh", reg: 962821205049, dept: "IT", clg: "UCEN", skills: ["Spring Boot",
  "CSS"] },
      { name: "Zaid", reg: 962821205050, dept: "CSE", clg: "UCEN", skills: ["Node.js", "HTML"]
  },
     ];

    @tracked filteredStudents = this.students;

    addStudent(student) {
      this.students = [...this.students, student];
      this.filteredStudents = this.students;
    }

    searchStudents(query) {
      query = query.toLowerCase();
      this.filteredStudents = this.students.filter(student => {
        const reg = student.reg ? student.reg.toString().toLowerCase() : '';
        return student.name.toLowerCase().includes(query) ||
          reg.includes(query) ||
          student.dept.toLowerCase().includes(query) ||
          student.clg.toLowerCase().includes(query);
      });
    }

  }
```

routes/students.js

```
import Route from '@ember/routing/route';
export default class StudentsRoute extends Route {}
```

routes/students/add.js

```
import Route from '@ember/routing/route';
import { inject as service } from '@ember/service';
export default class StudentsAddRoute extends Route {
  @service students;
  model() {
    return { students:this.students.students, newStudent: { name: '', reg: '', dept: '', clg: '' } };
  }
}
```

routes/students/edit.js

```
import Route from '@ember/routing/route';
import { inject as service } from '@ember/service';
```

```
export default class StudentsEditRoute extends Route {
   @service students;
   model(params) {
      return this.students.students.find(student => student.reg == Number(params.reg));
   }
}
```

controllers/students.js

```
import Controller from '@ember/controller';
import { action } from '@ember/object';
import { tracked } from '@glimmer/tracking';
import { service } from '@ember/service';
import { task, timeout } from 'ember-concurrency';

export default class StudentsController extends Controller {

   @service router;
   @service students;
   @service flashMessages;
   @tracked isFormVisible = false;
   @tracked editStudent = null;
   @tracked selectAll = false;

   get currentRoute() {
      return this.router.currentRouteName;
   }

   @action addStud() {
```

```
  this.router.transitionTo('students.add')
}

@action toggleSelectAll(event) {
  this.selectAll = event.target.checked;

  let checkboxes = document.querySelectorAll("#table input[type='checkbox']");
  checkboxes.forEach(checkbox => {
    checkbox.checked = this.selectAll;
  });
}

@task
*searchTable(event) {
  let input = event.target.value.toLowerCase();
  yield timeout(300);
  this.students.searchStudents(input);
}

@action handleSearch(event) {
  this.searchTable.perform(event);
}

@action bulkDelete() {
  let checkboxes = document.querySelectorAll("#table input[type='checkbox']:checked");

  if (!checkboxes.length) {
      this.flashMessages.warning('Select one or more rows to bulk delete', { timeout: 2000 });
      return;
    }

    let selectedRegs = Array.from(checkboxes).map(checkbox => {
      let row = checkbox.closest("tr");
      return row.cells[1].textContent.trim();
    });

    this.students.filteredStudents = this.students.filteredStudents.filter(
      student => !selectedRegs.includes(String(student.reg)));

    this.selectAll = false;
    checkboxes.forEach(checkbox => checkbox.checked = false);

    this.flashMessages.success('Selected students deleted successfully', { timeout: 2000 });
  }

  @action deleteRow(reg) {
    this.students.filteredStudents = this.students.filteredStudents.filter(student => student.reg !== reg);
  }

  @action editRow(reg) {
    this.editStudent = this.students.filteredStudents.find(student => student.reg === reg);
    this.router.transitionTo('students.edit', reg);
  }
}
```

controllers/students/add.js

```
import Controller from '@ember/controller';

import { service } from '@ember/service';

import { action } from '@ember/object';


export default class StudentsAddController extends Controller {

    @service router;

    @service students;

    @service flashMessages;


    @action addRow(student) {

        this.students.addStudent(student);

        this.router.transitionTo('students');

        this.flashMessages.success("Student added succcessfully!");

    }


    @action closeForm() {

        this.router.transitionTo('students');

    }

}
```

controllers/students/edit.js

```
import Controller from '@ember/controller';

import { service } from '@ember/service';

import { action } from '@ember/object';


export default class StudentsEditController extends Controller {
  @service router;

  @service students;

  @service flashMessages;


  @action updateStudents(updatedStudents) {

    this.students.filteredStudents = updatedStudents;

    this.router.transitionTo('students');

    this.flashMessages.success("Student updated succcessfully!");


  }


  @action closeForm() {

    this.router.transitionTo('students');

  }

}
```

components/student-form.hbs

```
<div id="formCont" {{on-key 'Enter' this.handleEnter}} {{on-key 'Escape' @closeForm}}>

  <h5>{{if this.isEditing "Edit Student" "Add New Student"}}</h5>

  <span id="close-btn" {{on "click" @closeForm}}>

    <i class="fa-solid fa-xmark"></i>

  </span>


  <input type="text" value={{this.name}} {{on "input" this.updateName}}
placeholder="Enter Name" required>

  <input type="number" value={{this.reg}} {{on "input" this.updateReg}} placeholder="Enter
Reg No">
```

```
    <input type="text" value={{this.clg}} {{on "input" this.updateClg}} placeholder="Enter College">

    <PowerSelect @options={{this.departments}} @selected={{this.dept}} @searchEnabled={{true}} @onChange={{this.updateDept}}
    @placeholder="Select dept..." as |department|>{{department}}</PowerSelect>

    <PowerSelectMultiple @options={{this.skills}} @selected={{this.selectedSkills}} @searchEnabled={{true}} @onChange={{this.updateSkills}} @placeholder="Select skills..." as |skill|>{{skill}}</PowerSelectMultiple>

    <div class="decision">
        <button class="btn-cd cancel" type="button" {{on "click" @closeForm}}>Cancel</button>
        <button class="btn-cd uc" type="button" {{on "click" this.saveStudent}}>
            {{if this.isEditing "Update" "Create"}}
        </button>
    </div>
</div>
```

components/student-list.hbs

```
<div class="table-container">
    <table id="table">
        <thead>
            <tr>
                <th><input type="checkbox" {{on "change" @toggleSelectAll}} checked={{@selectAll}} /></th>
                <th>Reg No</th>
```

```
          <th>Name</th>

          <th>Department</th>

          <th>Skills</th>

          <th>College</th>

          <th>Action</th>

        </tr>

      </thead>

      <tbody>

        {{#if (and @students (gt (get @students "length") 0))}}

        <VerticalCollection @estimateHeight={{50}} @items={{@students}} @key="reg"

          @containerSelector=".table-container" as |student|>

          <tr>

            <td><input type="checkbox" /></td>

            <td>{{student.reg}}</td>

            <td>{{student.name}}</td>

            <td>{{student.dept}}</td>

            <td>{{join-array student.skills ", "}}</td>

            <td>{{student.clg}}</td>

            <td>

              <button class="btn-ed edit" type="button" {{on "click" (fn @editRow
student.reg)}}>

                <i class="fa-solid fa-pen-to-square"></i>

              </button>

              <button class="btn-ed delete" type="button" {{on "click" (fn @deleteRow
student.reg)}}>

                <i class="fa-solid fa-trash"></i>

              </button>

            </td>

          </tr>
        </VerticalCollection>
        {{else}}
        <tr>
          <td colspan="7">No students found</td>
        </tr>
        {{/if}}
      </tbody>
    </table>
</div>
```

components/student-form.js

```
import Component from '@glimmer/component';
import { action } from '@ember/object';
import { tracked } from '@glimmer/tracking';
import { service } from '@ember/service';

export default class StudentFormComponent extends Component {
  @service flashMessages;

  @tracked name = this.args.editStudent?.name || '';
  @tracked reg = this.args.editStudent?.reg || '';
  @tracked selectedSkills = this.args.editStudent?.skills || [];
  @tracked dept = this.args.editStudent?.dept || null;
  @tracked clg = this.args.editStudent?.clg || '';
  @tracked isEditing = !!this.args.editStudent;

  skills = ['JavaScript', 'React', 'Spring Boot', 'Flutter', 'MySQL', 'Node.js', 'CSS', 'HTML'];
```

```
@action updateSkills(selected) {
  this.selectedSkills = selected;
  console.log(this.selectedSkills);
}

departments = ['IT', 'CSE', 'ECE', 'MECH'];
@action updateDept(selectedDept) {
  this.dept = selectedDept;
}

@action updateName(event) {
  this.name = event.target.value;
}

@action updateReg(event) {
  this.reg = event.target.value;
}

@action updateClg(event) {
  this.clg = event.target.value;
}

@action saveStudent() {

  if (!this.name || !this.reg || !this.dept || !this.clg || this.selectedSkills.length === 0) {
    this.flashMessages.warning("Fill all the details");
    return;
  }
```

```
if (this.isEditing) {
      let updatedStudents = this.args.students.map(student =>
        student.reg === this.reg
          ? { ...student, name: this.name, dept: this.dept,skills: this.selectedSkills, clg:
this.clg }
          : student
      );

      this.args.updateStudents(updatedStudents);
    } else {
      this.args.addRow({
        name: this.name,
        reg: this.reg,
        dept: this.dept,
        skills: this.selectedSkills,
        clg: this.clg
      });
    }

    this.args.closeForm();
  }

  @action
  handleEnter(event) {
    event.preventDefault();
    if (event.key === "Enter") {
      this.saveStudent();
    }
    }
}
```

styles/app.css

```
/* Ember supports plain CSS out of the box. More info:
https://cli.emberjs.com/release/advanced-use/stylesheets/ */


body {
    display: flex;
    justify-content: center;
    align-items: center;
    flex-direction: column;
    height: 100vh;
    font-family: Arial, Helvetica, sans-serif;
}


#overlay {
    position: fixed;
    top: 0;
    left: 0;
    height: 100%;
    width: 100%;
    background: linear-gradient(to bottom, rgba(107, 107, 108, 0.7), rgba(102, 102, 102, 0.3));
    z-index: 1000;
}


.flash-message {
    text-align: center;
    min-width: 250px !important;
    border-radius: 20px !important;
    padding: 1rem !important;
    z-index: 1050;
}

.student-container {
    display: flex;
    align-items: center;
    flex-direction: column;
    border: 1px solid #dddddd;
    border-radius: 10px;
    height: 80vh;
    width: 80%;
}

.full-screen-outlet {
    position: fixed;
    top: 0;
    left: 0;
    bottom: 0;
    width: 100vw;
    height: 100vh;
    z-index: 1000;
}

#formCont {
    display: flex;
    flex-direction: column;
    gap: 2rem;
    left: 35%;
    background-color: white;
    padding: 3rem;
    border-radius: 10px;
    width: 30%;
    z-index: 2000;
    position: relative;
    margin: 7% 0;
}

input {
    padding: 0.5rem 1rem;
    border: 1px solid #d0d0d0;
    border-radius: 5px;
    background-color: #fff;
    outline: none;
    height: 35px;
}

input:focus {
    border: 1px solid #868686;
}

input[type="checkbox"] {
    cursor: pointer;
    height: auto;
```

```
}

input::placeholder {
    font-size: 14px;
}

.search-dlt-cont span {
    font-size: 24px;
    font-weight: 700;
}

.table-container {
    overflow: auto;
    width: 100%;
    box-shadow: 3px 2px 10px #d3d3d3;
    background: white;
    border: 1px solid #e4e4e4;
    scrollbar-width: thin;
    margin: 0.5rem 0;
}

.table-container tr {
    height: auto;
    min-height: 40px;
}

table {
    background-color: #fff;
    border-radius: 10px;
    border-collapse: collapse;
    color: #3e3e3e;
    width: 100%;
    min-width: 1400px;
    table-layout: fixed;
}

thead {
    position: sticky;
    top: 0;
    background-color: #efefef;
    color: #6e6e6e;
    z-index: 10;
    text-align: center;
    font-size: 14px;
}

tbody {
    overflow-y: auto;
    width: 100%;
    scrollbar-width: thin;
    background-color: #ffffff;
    font-weight: 600;
}

tr th {
    padding: 0.5rem;
```

```
}

table tr {
   border-bottom: 1px solid #eeeded;
}

td {
   padding: 0.8rem;
   font-size: 14px;
   text-align: center;
   word-wrap: break-word;
}

th:first-child {
   background-color: #efefef;
}

td:first-child {
   background-color: #ffffff;
}

th:first-child,
td:first-child {
   position: sticky;
   left: 0;
   top: 0;
   z-index: 5;
   width: 100px;
```

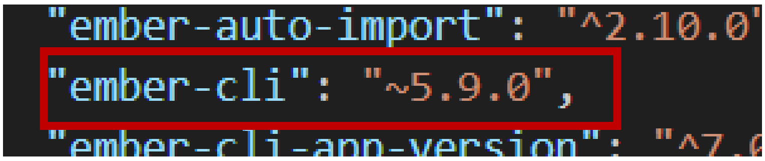
```
}

th:first-child::after,
td:first-child::after {
  content: "";
  position: absolute;
  top: 0;
  right: -5px;
  width: 5px;
  height: 100%;
  background: linear-gradient(to right, rgba(0, 0, 0, 0.05), transparent);
  pointer-events: none;
}

.btn-cd {
  padding: 0.5rem 0.8rem;
  border-radius: 5px;
  border: none;
  background-color: #4f8091;
  color: white;
  cursor: pointer;
  font-size: 14px;
  margin-left: 10px;
  text-decoration: none;
}

.cancel, .uc {
  margin: 0;
```

```
}

.add {
   background: linear-gradient(135deg, #4f8091, #2b5a68);
}

.btn-cd i {
   font-size: 12px;
   margin-right: 5px;
}

.dlt {
   background: linear-gradient(135deg, #e37272, #c94b4b);
   color: #ffffff;
}

.cancel {
   background-color: white;
   color: black;
   border: 1px solid #c8c8c8;
}

.btn-ed {
   background: transparent;
   border: none;
   cursor: pointer;
   margin: 0 0.3rem;
}
```

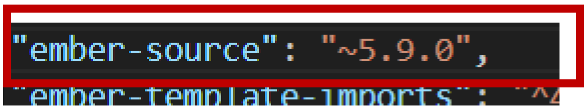
```
.btn-ed i {
    font-size: 14px;
}

.delete {
    color: #d84949;
}

.edit {
    color: #535353;
}

button i {
    font-size: 12px;
}

.decision {
    display: flex;
    width: 100%;
    gap: 1rem;
}

.decision button {
    width: 100%;
}

#close-btn {
    cursor: pointer;
    position: absolute;
    top: 2rem;
    right: 2rem;
    color: #3e3e3e;
    font-size: 20px;
}

.title-cont {
    display: flex;
    justify-content: center;
    align-items: center;
}

.search-dlt-cont {
    width: 95%;
    display: flex;
    justify-content: space-between;
    align-items: center;
    margin: 1rem;
    gap: 1rem;
}

#search {
    padding: 5px 0;
    border: none;
    width: 100%;
}
```

```
.search-cont {
   display: flex;
   align-items: center;
   border: 1px solid #ccc;
   width: 40%;
   color: #878787;
   padding: 0 1rem;
   border-radius: 5px;
}

.search-cont i {
   font-size: 14px;
   margin-right: 10px;
}

.search-cd-cont {
   display: flex;
   justify-content: flex-end;
   width: 70%;
}
```

package.json



```
  "ember-auto-import": "^2.10.0
  "ember-cli": "~5.9.0",
  "ember-cli-app-version": "^7.
```



```
"ember-source": "~5.9.0",
"ember-template-imports": "^
```