

OUTPUT:

Nested tables using yield (clicking a row should expand it to show another set of details in a nested table).

Student List

Customize Columns













Filters

Search here...

× Name

Delete Selected

Add Student

<input type="checkbox"/>	Reg No	Name	Department	Skills	College	Action											
<input type="checkbox"/>	963021203014	Faiz	MECH	Spring Boot, CSS	CIT	 											
<input type="checkbox"/>	963021201037	Raj	IT	React, Flutter	CIT	 											
	<table><tr><th>Course</th><th>Score</th><th>Credits</th><th>Grade</th></tr><tr><td>Web Development</td><td>90</td><td>4</td><td>A</td></tr><tr><td>Mobile App Dev</td><td>88</td><td>3</td><td>B+</td></tr></table>					Course	Score	Credits	Grade	Web Development	90	4	A	Mobile App Dev	88	3	B+
Course	Score	Credits	Grade														
Web Development	90	4	A														
Mobile App Dev	88	3	B+														
<input type="checkbox"/>	963121201011	Dinesh	IT	JavaScript, HTML	MIT	 											
<input type="checkbox"/>	963021201007	Bhaves	IT	Flutter, Spring Boot	CIT	 											
<input type="checkbox"/>	962821204019	Harsha	ECE	MySQL, HTML	UCEN	 											
<input type="checkbox"/>	963021202039	Ruban	CSE	HTML, CSS	CIT	 											

Student List

Customize Columns







Filters

Search here...

× Name

Delete Selected

Add Student

<input type="checkbox"/>	Reg No	Name	Department	Skills	College	Action												
<input type="checkbox"/>	963021203014	Faiz	MECH	Spring Boot, CSS	CIT	 												
<input type="checkbox"/>	<table><tr><th>Course</th><th>Score</th><th>Credits</th><th>Grade</th></tr><tr><td>Thermodynamics</td><td>85</td><td>4</td><td>A-</td></tr><tr><td>Fluid Mechanics</td><td>78</td><td>3</td><td>B</td></tr></table>						Course	Score	Credits	Grade	Thermodynamics	85	4	A-	Fluid Mechanics	78	3	B
	Course	Score	Credits	Grade														
	Thermodynamics	85	4	A-														
	Fluid Mechanics	78	3	B														
	<input type="checkbox"/>	963021201037	Raj	IT	React, Flutter	CIT	 											
<table><tr><th>Course</th><th>Score</th><th>Credits</th><th>Grade</th></tr><tr><td>Web Development</td><td>90</td><td>4</td><td>A</td></tr><tr><td>Mobile App Dev</td><td>88</td><td>3</td><td>B+</td></tr></table>						Course	Score	Credits	Grade	Web Development	90	4	A	Mobile App Dev	88	3	B+	
Course	Score	Credits	Grade															
Web Development	90	4	A															
Mobile App Dev	88	3	B+															
<input type="checkbox"/>	963121201011	Dinesh	IT	JavaScript, HTML	MIT	 												

## 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-  
Evv84Mr4kqVGRNSglGL/F/aIDqQb7xQ2vcrdlwxfjThSH8CSR7PBEakCr51Ck+w+/U6swU2Im1v  
VX0SVk9ABhg==" crossorigin="anonymous" referrerpolicy="no-referrer" />
```

```
    <link rel="stylesheet"  
href="https://cdn.jsdelivr.net/npm/bootstrap@5.3.0/dist/css/bootstrap.min.css">
```

```
    <link rel="stylesheet" href="https://cdn.jsdelivr.net/npm/bootstrap-  
icons@1.11.3/font/bootstrap-icons.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**

```
<nav class="navbar">
```

```
  <div class="navbar-title"><i class="fa-solid fa-graduation-cap"></i>Student Management  
  System</div>
```

```
  <div class="toggle-profile">
```

```
    <button id="theme-toggle-btn" {{on "click" this.toggleTheme}}>
```

```
      {{#if this.isDarkMode}}
```

```
        <i class="bi bi-sun"></i>
```

```
      {{else}}
```

```
        <i class="bi bi-moon-stars"></i>
```

```
      {{/if}}
```

```
    </button>
```

```
    <i class="bi bi-person-circle"></i>
```

```
  </div>
```

```
</nav>
```

```

{{#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">
    <div><span>Student List</span></div>

    <div class="search-cd-cont">
      <button class="btn-filter" {{on "click" this.toggleCustomize}}>
        <i class="bi bi-layout-sidebar-inset"></i> Customize Columns
      </button>
      {{#if this.isCustomizeVisible}}
      <div class="customize-column" {{on-click-outside this.toggleCustomize}}>
        <div class="customize-column-header">
          <div>
            <strong>Columns</strong>
            <p class="customize-desc">Drag <i class="bi bi-list"></i> to reorder
columns</p>
          </div>
          <div class="customize-column-rc">
            <p {{on "click" this.resetColumns}} class="btn-reset">Reset Order</p>
            <p {{on "click" this.toggleCustomize}} {{on-key 'Escape' this.toggleCustomize}}

```

```

        class="close-modal"><i class="fa-solid fa-xmark"></i></p>

</div>

</div>

<ul {{sortable-group onChange=this.reorderItems}}>
    {{#each this.items as |col|}}
        <li {{sortable-item model=col}}>
            <div>
                <label>
                    <input type="checkbox" checked={{this.isColumnVisible col.key}} {{on
"change" (fn
                    this.toggleColumnVisibility col.key)}} />
                    {{capitalize col.label}}
                </label>
                <span class="handle" {{sortable-handle}}><i class="bi bi-list"></i></span>
            </div>
        </li>
    {{/each}}
</ul>
</div>
{{/if}}

<button class="btn-filter" {{on "click" this.toggleFilter}}>
    <i class="bi bi-funnel"></i> Filters
</button>

{{#if this.isFilterVisible}}
    <div class="advanced-filters" {{on-click-outside this.toggleFilter}}>

```

```

<div class="filter-section">

  <div class="filter-section-header">

    <strong>Departments</strong>

    <span {{on "click" this.toggleFilter}} {{on-key 'Escape' this.toggleFilter}}

      class="close-modal"><i class="fa-solid fa-xmark"></i></span>

  </div>

  <div class="filter-dept">

    {{#each this.availableDepartments as |dept|}}

      <label>

        <input type="checkbox" value={{dept}} {{on "change" this.updateDeptFilter}}

          checked={{this.isDeptChecked dept}} />

        {{dept}}

      </label>

    {{/each}}

  </div>

</div>

<div class="filter-section">

  <strong>Skills:</strong>

  <div class="filter-skill">

    {{#each this.availableSkills as |skill|}}

      <label>

        <input type="checkbox" value={{skill}} {{on "change" this.updateSkillsFilter}}

          checked={{includes this.filter.skills skill}} />

        {{skill}}

      </label>

    {{/each}}

  </div>

</div>

```

```

        </div>

    </div>

    <button type="button" {{on "click" this.applyFilter}} class="btn-cd filter">Apply
Filters</button>

    <button type="button" {{on "click" this.clearAllFilters}} class="btn-cd cancel">Clear
All</button>

</div>

{{/if}}

<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 class="power-select-wrapper">

        <PowerSelectMultiple @selected={{this.selectedColumnValues}}
@options={{this.items}}

            @onChange={{this.updateSearchColumns}} @placeholder="Search by"
@searchEnabled={{false}} as

                |option|>

                    {{option.label}}

        </PowerSelectMultiple>

    </div>

</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 @items={{this.items}} @students={{this.students.filteredStudents}}  
@deleteRow={{this.deleteRow}}

@editRow={{this.editRow}} @toggleSelectAll={{this.toggleSelectAll}}  
@selectAll={{this.selectAll}}

@sortByName={{this.sortByName}} @isAscending={{this.isAscending}}  
@visibleColumns={{this.visibleColumns}}

@sortKey={{this.sortKey}} @sortByKey={{this.sortByKey}}  
@makeResizable={{this.makeResizable}}

@generateWidthStyle={{this.generateWidthStyle}} @handleResize={{this.handleResize}}

@expandedRows={{this.expandedRows}} @toggleRow={{this.toggleRow}} as |student|>

<table class="nested-table">

<thead style="z-index: 0;">

<tr>

<th>Course</th>

<th>Score</th>

<th>Credits</th>

<th>Grade</th>

</tr>

</thead>

<tbody>

{{#each student.courses as |course|}}

<tr>

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

<td>{{course.score}}</td>

<td>{{course.credits}}</td>

<td>{{course.grade}}</td>



```
        </tr>
      {{/each}}
    </tbody>
  </table>
</StudentList>
</div>
```

### **templates/students/add.hbs**

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

### **templates/students/edit.hbs**

```
<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: "Faiz",
```

```
reg: 963021203014,
dept: "MECH",
clg: "CIT",
skills: ["Spring Boot", "CSS"],
courses: [
  { name: "Thermodynamics", score: 85, credits: 4, grade: "A-" },
  { name: "Fluid Mechanics", score: 78, credits: 3, grade: "B" }
],
{
  name: "Raj",
  reg: 963021201037,
  dept: "IT",
  clg: "CIT",
  skills: ["React", "Flutter"],
  courses: [
    { name: "Web Development", score: 90, credits: 4, grade: "A" },
    { name: "Mobile App Dev", score: 88, credits: 3, grade: "B+" }
  ],
  {
    name: "Dinesh",
    reg: 963121201011,
    dept: "IT",
    clg: "MIT",
    skills: ["JavaScript", "HTML"],
    courses: [
      { name: "Data Structures", score: 82, credits: 4, grade: "B+" },
```

```
{ name: "Algorithms", score: 79, credits: 3, grade: "B" }  
]  
,  
{  
  name: "Bhavesh",  
  reg: 963021201007,  
  dept: "IT",  
  clg: "CIT",  
  skills: ["Flutter", "Spring Boot"],  
  courses: [  
    { name: "Mobile App Dev", score: 92, credits: 4, grade: "A" },  
    { name: "Web Services", score: 85, credits: 3, grade: "A-" }  
  ]  
},  
{  
  name: "Harsha",  
  reg: 962821204019,  
  dept: "ECE",  
  clg: "UCEN",  
  skills: ["MySQL", "HTML"],  
  courses: [  
    { name: "Digital Electronics", score: 88, credits: 4, grade: "B+" },  
    { name: "Signals & Systems", score: 76, credits: 3, grade: "B-" }  
  ]  
},  
{  
  name: "Ruban",  
  reg: 963021202039,
```

```
dept: "CSE",
clg: "CIT",
skills: ["HTML", "CSS"],
courses: [
  { name: "Computer Networks", score: 84, credits: 4, grade: "B+" },
  { name: "Operating Systems", score: 81, credits: 3, grade: "B" }
],
{
  name: "Sanjay",
  reg: 963121201040,
  dept: "IT",
  clg: "MIT",
  skills: ["Node.js", "Spring Boot"],
  courses: [
    { name: "Web Development", score: 89, credits: 4, grade: "B+" },
    { name: "Cloud Computing", score: 83, credits: 3, grade: "B" }
  ],
{
  name: "Guhan",
  reg: 963121203016,
  dept: "MECH",
  clg: "MIT",
  skills: ["Flutter", "React"],
  courses: [
    { name: "Machine Design", score: 77, credits: 4, grade: "B-" },
    { name: "CAD/CAM", score: 82, credits: 3, grade: "B+" }
```

```
]
},
{
  name: "Jagan",
  reg: 962821201022,
  dept: "IT",
  clg: "UCEN",
  skills: ["React", "Spring Boot"],
  courses: [
    { name: "Database Systems", score: 91, credits: 4, grade: "A" },
    { name: "Web Programming", score: 87, credits: 3, grade: "B+" }
  ]
},
{
  name: "Charan",
  reg: 963121202008,
  dept: "CSE",
  clg: "MIT",
  skills: ["CSS", "MySQL"],
  courses: [
    { name: "Software Engineering", score: 84, credits: 4, grade: "B+" },
    { name: "Compiler Design", score: 79, credits: 3, grade: "B" }
  ]
},
{
  name: "Vijay",
  reg: 962821204047,
  dept: "ECE",
```

```
    clg: "UCEN",
    skills: ["Flutter", "MySQL"],
    courses: [
      { name: "Microprocessors", score: 85, credits: 4, grade: "A-" },
      { name: "VLSI Design", score: 80, credits: 3, grade: "B" }
    ]
  },
  {
    name: "Manoj",
    reg: 962821201030,
    dept: "IT",
    clg: "UCEN",
    skills: ["JavaScript", "React"],
    courses: [
      { name: "AI Fundamentals", score: 83, credits: 4, grade: "B" },
      { name: "Machine Learning", score: 78, credits: 3, grade: "B-" }
    ]
  },
  {
    name: "Kiran",
    reg: 962821203026,
    dept: "MECH",
    clg: "UCEN",
    skills: ["CSS", "Flutter"],
    courses: [
      { name: "Heat Transfer", score: 76, credits: 4, grade: "B-" },
      { name: "Material Science", score: 82, credits: 3, grade: "B+" }
    ]
  }
]
```

```
},
{
  name: "Surya",
  reg: 963121203043,
  dept: "MECH",
  clg: "MIT",
  skills: ["React", "Flutter"],
  courses: [
    { name: "Dynamics", score: 79, credits: 4, grade: "B" },
    { name: "Control Systems", score: 81, credits: 3, grade: "B" }
  ]
},
{
  name: "Dev",
  reg: 963021201010,
  dept: "IT",
  clg: "CIT",
  skills: ["React", "HTML"],
  courses: [
    { name: "Data Mining", score: 87, credits: 4, grade: "B+" },
    { name: "Big Data", score: 84, credits: 3, grade: "B+" }
  ]
},
{
  name: "Aswin",
  reg: 962821203005,
  dept: "MECH",
  clg: "UCEN",
```

```
skills: ["Node.js", "MySQL"],
courses: [
  { name: "Robotics", score: 83, credits: 4, grade: "B" },
  { name: "Mechatronics", score: 79, credits: 3, grade: "B" }
]
},
{
  name: "Yogesh",
  reg: 962821203049,
  dept: "MECH",
  clg: "UCEN",
  skills: ["Spring Boot", "CSS"],
  courses: [
    { name: "Automobile Engg", score: 78, credits: 4, grade: "B-" },
    { name: "IC Engines", score: 82, credits: 3, grade: "B+" }
  ]
},
{
  name: "Lokesh",
  reg: 963021204029,
  dept: "ECE",
  clg: "CIT",
  skills: ["Flutter", "MySQL"],
  courses: [
    { name: "Embedded Systems", score: 86, credits: 4, grade: "B+" },
    { name: "DSP", score: 80, credits: 3, grade: "B" }
  ]
},
```



```
{
  name: "Kavin",
  reg: 963121202027,
  dept: "CSE",
  clg: "MIT",
  skills: ["Spring Boot", "React"],
  courses: [
    { name: "Computer Graphics", score: 89, credits: 4, grade: "B+" },
    { name: "AI Principles", score: 85, credits: 3, grade: "A-" }
  ]
},
{
  name: "Omprakash",
  reg: 962821203034,
  dept: "MECH",
  clg: "UCEN",
  skills: ["CSS", "HTML"],
  courses: [
    { name: "Production Tech", score: 77, credits: 4, grade: "B-" },
    { name: "Operations Research", score: 81, credits: 3, grade: "B" }
  ]
},
{
  name: "Ethan",
  reg: 962821202012,
  dept: "CSE",
  clg: "UCEN",
  skills: ["Node.js", "JavaScript"],
```

```
courses: [  
  { name: "Cryptography", score: 90, credits: 4, grade: "A" },  
  { name: "Network Security", score: 87, credits: 3, grade: "B+" }  
]  
,  
{  
  name: "Ajay",  
  reg: 963021204003,  
  dept: "ECE",  
  clg: "CIT",  
  skills: ["Spring Boot", "MySQL"],  
  courses: [  
    { name: "Communication Systems", score: 84, credits: 4, grade: "B+" },  
    { name: "Antenna Theory", score: 79, credits: 3, grade: "B" }  
  ]  
},  
{  
  name: "Zaid",  
  reg: 963021201050,  
  dept: "IT",  
  clg: "CIT",  
  skills: ["Node.js", "HTML"],  
  courses: [  
    { name: "Cloud Architecture", score: 88, credits: 4, grade: "B+" },  
    { name: "DevOps", score: 83, credits: 3, grade: "B" }  
  ]  
},  
{
```

```
name: "Abishek",
reg: 962821202002,
dept: "CSE",
clg: "UCEN",
skills: ["Spring Boot", "Flutter"],
courses: [
  { name: "Distributed Systems", score: 86, credits: 4, grade: "B+" },
  { name: "Parallel Computing", score: 82, credits: 3, grade: "B+" }
]
},
{
  name: "Irfan",
  reg: 963021202021,
  dept: "CSE",
  clg: "CIT",
  skills: ["Spring Boot", "JavaScript"],
  courses: [
    { name: "Web Technologies", score: 89, credits: 4, grade: "B+" },
    { name: "Mobile Computing", score: 84, credits: 3, grade: "B+" }
  ]
},
{
  name: "Hari",
  reg: 963021202017,
  dept: "CSE",
  clg: "CIT",
  skills: ["CSS", "HTML"],
  courses: [
```

```
{ name: "Human-Computer Interaction", score: 85, credits: 4, grade: "A-" },
{ name: "UI/UX Design", score: 88, credits: 3, grade: "B+" }
]
},
{
  name: "Elavarasan",
  reg: 962821201013,
  dept: "IT",
  clg: "UCEN",
  skills: ["HTML", "React"],
  courses: [
    { name: "Information Security", score: 83, credits: 4, grade: "B" },
    { name: "Ethical Hacking", score: 79, credits: 3, grade: "B" }
  ]
},
{
  name: "Nithin",
  reg: 963021201033,
  dept: "IT",
  clg: "CIT",
  skills: ["Node.js", "CSS"],
  courses: [
    { name: "Blockchain Tech", score: 87, credits: 4, grade: "B+" },
    { name: "IoT Fundamentals", score: 82, credits: 3, grade: "B+" }
  ]
},
{
  name: "Lalith",
```

```
reg: 963021202028,
dept: "CSE",
clg: "CIT",
skills: ["HTML", "MySQL"],
courses: [
  { name: "Database Admin", score: 90, credits: 4, grade: "A" },
  { name: "Data Warehousing", score: 85, credits: 3, grade: "A-" }
]
},
{
  name: "Sathish",
  reg: 962821201041,
  dept: "IT",
  clg: "UCEN",
  skills: ["JavaScript", "Node.js"],
  courses: [
    { name: "Natural Language Processing", score: 84, credits: 4, grade: "B+" },
    { name: "Computer Vision", score: 79, credits: 3, grade: "B" }
  ]
},
{
  name: "Udhay",
  reg: 963121202045,
  dept: "CSE",
  clg: "MIT",
  skills: ["JavaScript", "MySQL"],
  courses: [
    { name: "Software Testing", score: 86, credits: 4, grade: "B+" },
```

```
{ name: "Quality Assurance", score: 82, credits: 3, grade: "B+" }  
]  
,  
{  
  name: "Keerthi",  
  reg: 962821203025,  
  dept: "MECH",  
  clg: "UCEN",  
  skills: ["React", "CSS"],  
  courses: [  
    { name: "Finite Element Analysis", score: 78, credits: 4, grade: "B-" },  
    { name: "Mech Vibrations", score: 83, credits: 3, grade: "B" }  
  ]  
},  
{  
  name: "Deepak",  
  reg: 962821201009,  
  dept: "IT",  
  clg: "UCEN",  
  skills: ["JavaScript", "Spring Boot"],  
  courses: [  
    { name: "Web Services", score: 89, credits: 4, grade: "B+" },  
    { name: "REST APIs", score: 86, credits: 3, grade: "B+" }  
  ]  
},  
{  
  name: "Jeeva",  
  reg: 962821201023,
```

```
    dept: "IT",
    clg: "UCEN",
    skills: ["HTML", "Node.js"],
    courses: [
      { name: "Cloud Security", score: 84, credits: 4, grade: "B+" },
      { name: "Cyber Laws", score: 80, credits: 3, grade: "B" }
    ]
  },
  {
    name: "Aakash",
    reg: 962821201001,
    dept: "IT",
    clg: "UCEN",
    skills: ["JavaScript", "React"],
    courses: [
      { name: "Deep Learning", score: 87, credits: 4, grade: "B+" },
      { name: "Neural Networks", score: 83, credits: 3, grade: "B" }
    ]
  },
  {
    name: "Praveen",
    reg: 963121204036,
    dept: "ECE",
    clg: "MIT",
    skills: ["MySQL", "HTML"],
    courses: [
      { name: "Wireless Comm", score: 85, credits: 4, grade: "A-" },
      { name: "Satellite Comm", score: 79, credits: 3, grade: "B" }
```

```
]
},
{
  name: "Karthik",
  reg: 963121201024,
  dept: "IT",
  clg: "MIT",
  skills: ["Flutter", "React"],
  courses: [
    { name: "Mobile Security", score: 88, credits: 4, grade: "B+" },
    { name: "App Performance", score: 84, credits: 3, grade: "B+" }
  ]
},
{
  name: "Muhamad",
  reg: 962821201031,
  dept: "IT",
  clg: "UCEN",
  skills: ["JavaScript", "React"],
  courses: [
    { name: "Data Analytics", score: 86, credits: 4, grade: "B+" },
    { name: "Business Intelligence", score: 82, credits: 3, grade: "B+" }
  ]
},
{
  name: "Harish",
  reg: 962821201018,
  dept: "IT",
```



```
    clg: "UCEN",
    skills: ["Node.js", "CSS"],
    courses: [
      { name: "Agile Methodologies", score: 85, credits: 4, grade: "A-" },
      { name: "Scrum Master", score: 88, credits: 3, grade: "B+" }
    ]
  },
  {
    name: "Arun",
    reg: 963121201004,
    dept: "IT",
    clg: "MIT",
    skills: ["React", "Spring Boot"],
    courses: [
      { name: "Microservices", score: 89, credits: 4, grade: "B+" },
      { name: "Containerization", score: 86, credits: 3, grade: "B+" }
    ]
  },
  {
    name: "Pranav",
    reg: 962821202035,
    dept: "CSE",
    clg: "UCEN",
    skills: ["MySQL", "JavaScript"],
    courses: [
      { name: "NoSQL Databases", score: 87, credits: 4, grade: "B+" },
      { name: "MongoDB", score: 83, credits: 3, grade: "B" }
    ]
  }
]
```

```
},  
{  
  name: "Venkatesh",  
  reg: 963021203046,  
  dept: "MECH",  
  clg: "CIT",  
  skills: ["React", "Flutter"],  
  courses: [  
    { name: "CAD Modeling", score: 82, credits: 4, grade: "B+" },  
    { name: "3D Printing", score: 85, credits: 3, grade: "A-" }  
  ]  
},  
{  
  name: "Imran",  
  reg: 963121202020,  
  dept: "CSE",  
  clg: "MIT",  
  skills: ["Node.js", "JavaScript"],  
  courses: [  
    { name: "Web Security", score: 88, credits: 4, grade: "B+" },  
    { name: "Penetration Testing", score: 84, credits: 3, grade: "B+" }  
  ]  
},  
{  
  name: "Gokul",  
  reg: 962821204015,  
  dept: "ECE",  
  clg: "UCEN",
```

```
skills: ["MySQL", "Flutter"],
courses: [
  { name: "IoT Systems", score: 85, credits: 4, grade: "A-" },
  { name: "Sensor Networks", score: 80, credits: 3, grade: "B" }
]
},
{
  name: "Naveen",
  reg: 963121202032,
  dept: "CSE",
  clg: "MIT",
  skills: ["Spring Boot", "Node.js"],
  courses: [
    { name: "Enterprise Architecture", score: 87, credits: 4, grade: "B+" },
    { name: "SOA", score: 83, credits: 3, grade: "B" }
  ]
},
{
  name: "Vimal",
  reg: 963121202048,
  dept: "CSE",
  clg: "MIT",
  skills: ["React", "CSS"],
  courses: [
    { name: "Frontend Frameworks", score: 89, credits: 4, grade: "B+" },
    { name: "Responsive Design", score: 86, credits: 3, grade: "B+" }
  ]
},
```

```
{
  name: "Bala",
  reg: 962821202006,
  dept: "CSE",
  clg: "UCEN",
  skills: ["CSS", "Node.js"],
  courses: [
    { name: "Web Performance", score: 85, credits: 4, grade: "A-" },
    { name: "Progressive Web Apps", score: 88, credits: 3, grade: "B+" }
  ]
},
{
  name: "Tharun",
  reg: 962821204044,
  dept: "ECE",
  clg: "UCEN",
  skills: ["HTML", "CSS"],
  courses: [
    { name: "RF Engineering", score: 82, credits: 4, grade: "B+" },
    { name: "Microwave Engg", score: 79, credits: 3, grade: "B" }
  ]
},
{
  name: "Ragul",
  reg: 962821203038,
  dept: "MECH",
  clg: "UCEN",
  skills: ["Node.js", "Flutter"],
```

```
courses: [  
  { name: "Mechatronics", score: 84, credits: 4, grade: "B+" },  
  { name: "Industrial Automation", score: 81, credits: 3, grade: "B" }  
]  
}  
];
```

```
@tracked filteredStudents = this.students;
```

```
searchStudents(query, searchColumns = ['name', 'dept', 'clg']) {  
  query = query.toLowerCase();
```

```
  this.filteredStudents = this.students.filter(student => {  
    return searchColumns.some(column => {  
      switch (column) {  
        case 'reg':  
          return student.reg.toString().toLowerCase().includes(query);  
        case 'skills':  
          return student.skills.join(' ').toLowerCase().includes(query);  
        case 'name':  
        case 'dept':  
        case 'clg':  
          return student[column]?.toLowerCase().includes(query);  
        default:  
          return false;  
        }  
      });  
    });  
  });
```

```
}
```

```
}
```

### **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';  
export default class StudentsAddRoute extends Route { }
```

### **routes/students/edit.js**

```
import Route from '@ember/routing/route';  
import { 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 { htmlSafe } from '@ember/template';

export default class StudentsController extends Controller {

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

  @tracked sortKey = 'name';
  @tracked isAscending = true;

  @tracked recentlyDeletedStudents = [];

  @tracked isFilterVisible = false;
  @tracked filter = {
    depts: [],
    skills: []
  };

  @tracked isCustomizeVisible = false;
  @tracked visibleColumns = ['reg', 'name', 'dept', 'skills', 'clg'];

  @tracked searchColumns = ['name'];
```

```
@tracked lastSearchQuery = '';
```

```
@tracked expandedRows = [];
```

```
DEFAULT_COLUMNS = [  
  { key: 'reg', label: 'Reg No' },  
  { key: 'name', label: 'Name' },  
  { key: 'dept', label: 'Department' },  
  { key: 'skills', label: 'Skills' },  
  { key: 'clg', label: 'College' }  
];
```

```
@tracked items = this.loadSavedColumns();
```

```
loadSavedColumns() {  
  try {  
    const saved = localStorage.getItem('studentColumns');  
    return saved ? JSON.parse(saved) : this.DEFAULT_COLUMNS;  
  } catch (e) {  
    console.error('Failed to load saved columns', e);  
    return this.DEFAULT_COLUMNS;  
  }  
}
```

```
@action reorderItems(itemModels) {  
  this.items = itemModels;  
  this.saveColumns();  
}
```



```
@action resetColumns() {
```

```
  this.items = [...this.DEFAULT_COLUMNS];
```

```
  this.saveColumns();
```

```
}
```

```
saveColumns() {
```

```
  try {
```

```
    localStorage.setItem('studentColumns', JSON.stringify(this.items));
```

```
  } catch (e) {
```

```
    console.error('Failed to save columns', e);
```

```
  }
```

```
}
```

```
@action toggleRow(reg) {
```

```
  if (this.expandedRows.includes(reg)) {
```

```
    this.expandedRows = this.expandedRows.filter(r => r !== reg);
```

```
  } else {
```

```
    this.expandedRows = [...this.expandedRows, reg];
```

```
  }
```

```
}
```

```
get isDeptChecked() {
```

```
  return (dept) => {
```

```
    return this.filter.depts.includes(dept);
```

```
  };
```

```
}
```

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

```
@action makeResizable(th) {  
  let resizer = th.querySelector('.resizer');  
  if (!resizer) return;
```

```
  const key = th.getAttribute('data-key');  
  const minWidth = 100;
```

```
  let savedWidth = localStorage.getItem(`col-width-${key}`);  
  if (savedWidth) {  
    th.style.width = `${savedWidth}px`;  
  }
```

```
  let startX, startWidth;
```

```
  const onMouseDown = (e) => {  
    startX = e.pageX;  
    startWidth = th.offsetWidth;
```

```
    document.addEventListener('mousemove', onMouseMove);  
    document.addEventListener('mouseup', onMouseUp);  
  };
```

```
  const onMouseMove = (e) => {  
    let diff = e.pageX - startX;
```

```
let newWidth = startWidth + diff;
if (newWidth < minWidth) return;

th.style.width = `${newWidth}px`;
};

const onMouseUp = () => {
  const finalWidth = th.offsetWidth;
  localStorage.setItem(`col-width-${key}`, finalWidth);

  document.removeEventListener('mousemove', onMouseMove);
  document.removeEventListener('mouseup', onMouseUp);
};

resizer.addEventListener('mousedown', onMouseDown);
}

@action toggleFilter() {
  this.isFilterVisible = !this.isFilterVisible;
}

@action toggleCustomize() {
  this.isCustomizeVisible = !this.isCustomizeVisible;
}

get availableDepartments() {
  let depts = this.students.students.map(student => student.dept);
  return [...new Set(depts)];
}
```

```
}
```

```
get availableSkills() {  
    let allSkills = this.students.students.flatMap(student => student.skills || []);  
    return [...new Set(allSkills)];  
}
```

```
@action applyFilter() {  
    this.isFilterVisible = !this.isFilterVisible;  
    this.students.filteredStudents = this.students.students.filter(student => {  
        let { depts, skills } = this.filter;  
  
        if (depts.length && !depts.includes(student.dept)) {  
            return false;  
        }  
  
        if (skills.length && !skills.every(skill => student.skills.includes(skill))) {  
            return false;  
        }  
  
        return true;  
    });  
    this.flashMessages.success(htmlSafe('<i class="bi bi-check-circle-fill"></i> Filters applied'));  
}
```

```
@action clearAllFilters() {  
    this.filter = {  
        depts: [],
```

```
    skills: []  
  };  
  this.isFilterVisible = false;  
  this.students.filteredStudents = [...this.students.students];  
  this.flashMessages.success(htmlSafe('<i class="bi bi-check-circle-fill"></i> Filters cleared'));  
}
```

```
@action updateDeptFilter(event) {  
  let value = event.target.value;  
  let isChecked = event.target.checked;  
  
  if (isChecked) {  
    this.filter = {  
      ...this.filter,  
      depts: [...this.filter.depts, value]  
    };  
  } else {  
    this.filter = {  
      ...this.filter,  
      depts: this.filter.depts.filter(dept => dept !== value)  
    };  
  }  
}
```

```
@action updateSkillsFilter(event) {  
  let value = event.target.value;  
  let isChecked = event.target.checked;
```

```
if (isChecked) {  
  this.filter = {  
    ...this.filter,  
    skills: [...this.filter.skills, value]  
  };  
} else {  
  this.filter = {  
    ...this.filter,  
    skills: this.filter.skills.filter(skill => skill !== value)  
  };  
}  
}
```

```
@action toggleColumnVisibility(column) {  
  if (this.visibleColumns.includes(column)) {  
    this.visibleColumns = this.visibleColumns.filter(c => c !== column);  
  } else {  
    this.visibleColumns = [...this.visibleColumns, column];  
  }  
}
```

```
@action isColumnVisible(column) {  
  return this.visibleColumns.includes(column);  
}
```

```
@action sortByKey(key) {  
  this.sortKey = key;  
  console.log(this.sortKey)
```

```
let sorted = [...this.students.filteredStudents].sort((a, b) => {
```

```
  let aVal = a[key];
```

```
  let bVal = b[key];
```

```
  if (Array.isArray(aVal)) aVal = aVal.join(', ');
```

```
  if (Array.isArray(bVal)) bVal = bVal.join(', ');
```

```
  if (typeof aVal === 'string' && typeof bVal === 'string') {
```

```
    return this.isAscending
```

```
      ? aVal.localeCompare(bVal)
```

```
      : bVal.localeCompare(aVal);
```

```
  } else {
```

```
    return this.isAscending ? aVal - bVal : bVal - aVal;
```

```
  }
```

```
});
```

```
this.students.filteredStudents = sorted;
```

```
this.isAscending = !this.isAscending;
```

```
}
```

```
@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;
});
}
```

```
@action handleSearch(event) {
  const query = event.target.value.toLowerCase();
  this.lastSearchQuery = query;
  this.students.searchStudents(query, this.searchColumns);
}
```

```
get selectedColumnValues() {
  return this.searchColumns.map(key => {
    return this.items.find(opt => opt.key === key);
  }).filter(Boolean);
}
```

```
@action
updateSearchColumns(selected) {
  this.searchColumns = selected.map(item => item.key);

  if (this.lastSearchQuery) {
    this.students.searchStudents(this.lastSearchQuery, this.searchColumns);
  }
}
```

```
@action bulkDelete() {
```



```

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

if (!checkboxes.length) {
  this.flashMessages.warning(htmlSafe('<i class="bi bi-exclamation-triangle-fill"></i> Select
one or more rows to delete'));
  return;
}

let selectedRegs = Array.from(checkboxes).map(checkbox => {
  let row = checkbox.closest("tr");
  return row.dataset.reg;
});

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(htmlSafe('<i class="bi bi-check-circle-fill"></i> Selected
students deleted successfully', { timeout: 2000 }));
}

@action deleteRow(reg) {
  const index = this.students.filteredStudents.findIndex(s => s.reg === reg);
  const student = this.students.filteredStudents[index];

  this.students.filteredStudents = this.students.filteredStudents.filter(s => s.reg !== reg);

```

```

const id = Date.now();

this.recentlyDeletedStudents.push({ id, student, index });

this.flashMessages.add({
  type: 'success',
  message: htmlSafe(`<i class="bi bi-check-circle-fill"></i> Student deleted. <button
class="undo-btn" data-id="${id}">Undo</button>`),
  timeout: 5000,
  sticky: true,
});

setTimeout(() => {
  const undoBtn = document.querySelector(`.undo-btn[data-id="${id}"]`);
  if (undoBtn) {
    undoBtn.addEventListener('click', () => this.undoDelete(id));
  }
}, 0);

setTimeout(() => {
  this.recentlyDeletedStudents = this.recentlyDeletedStudents.filter(item => item.id !== id);
}, 5000);
}

@action undoDelete(id) {
  const entry = this.recentlyDeletedStudents.find(item => item.id === id);
  if (entry) {
    const studentsCopy = [...this.students.filteredStudents];
    studentsCopy.splice(entry.index, 0, entry.student);
  }
}

```

```
this.students.filteredStudents = studentsCopy;
```

```
this.recentlyDeletedStudents = this.recentlyDeletedStudents.filter(  
  item => item.id !== id  
);
```

```
this.flashMessages.success(htmlSafe('<i class="bi bi-check-circle-fill"></i> Undo  
successful!'));  
  
}  
}
```

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

```
@action toggleTheme() {  
  this.isDarkMode = !this.isDarkMode;  
  const newTheme = this.isDarkMode ? 'dark' : 'light';  
  document.documentElement.setAttribute('data-theme', newTheme);  
}  
}
```

### **controllers/students/add.js**

```
import Controller from '@ember/controller';  
import { service } from '@ember/service';  
import { action } from '@ember/object';
```

```
import { htmlSafe } from '@ember/template';

export default class StudentsAddController extends Controller {

  @service router;

  @service students;

  @service flashMessages;

  @action addRow(student) {

    this.students.students = [...this.students.students, student];

    this.students.filteredStudents = this.students.students;

    this.router.transitionTo('students');

    this.flashMessages.success(htmlSafe('<i class="bi bi-check-circle-fill"></i> Student
added successfully!'));

  }

  @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';

import { htmlSafe } from '@ember/template';

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(htmlSafe('<i class="bi bi-check-circle-fill"></i> Student
updated successfully!'));
    }

    @action closeForm() {
        this.router.transitionTo('students');
    }
}

```

### components/student-form.hbs

```

<form id="formCont" {{on "keydown" this.handleEnter}} {{on-key 'Escape' @closeForm}}
{{on-click-outside @closeForm}}>

    <h4>{{if this.isEditing "Edit Student" "Add New Student"}}</h4>

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

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

    </span>

    <label for="name">Name</label>

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

```

```
<label for="reg">Registration Number</label>
```

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

```
<label for="clg">College</label>
```

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

```
<label for="dept">Department</label>
```

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

```
<label for="skills">Skills</label>
```

```
<PowerSelectMultiple @options={{this.skills}} @selected={{this.selectedSkills}}  
@searchEnabled={{true}}  
  @onChange={{this.updateSkills}} @placeholder="Select skills..." @triggerId="skills"  
  @required={{true}} 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="submit" {{on "click" this.saveStudent}}>
```

```
    {{if this.isEditing "Update" "Add"}}
```

```
  </button>
```

```
</div>
```

```
</form>
```

## components/student-list.hbs

```
<div class="table-container">
```

```
<table id="table">
```

```
<thead>
```

```
<tr class="main-row">
```

```
<th><input type="checkbox" {{on "change" @toggleSelectAll}} checked={{@selectAll}}  
/></th>
```

```
{{#each @items as |item|}}
```

```
{{#if (includes @visibleColumns item.key)}}
```

```
<th class="resizable" data-key={{item.key}} {{did-insert @makeResizable}}>
```

```
{{item.label}}
```

```
{{#if (or (eq item.key "name") (eq item.key "reg") (eq item.key "dept") (eq item.key  
"clg") (eq item.key  
"skills"))}}
```

```
<i class="bi bi-arrow-down-up" {{on "click" (fn @sortByKey item.key)}}></i>
```

```
{{/if}}
```

```
<span class="resizer"></span>
```

```
<i class="bi bi-three-dots-vertical" style="float: right; opacity:0.7; cursor:default;"></i>
```

```
</th>
```

```
{{/if}}
```

```
{{/each}}
```

```

        <th>Action</th>

    </tr>

</thead>

<tbody>

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

        <VerticalCollection @estimateHeight={{60}} @items={{@students}} @key="reg"
@containerSelector=".table-container"

            as |student|>

                <tr data-reg={{student.reg}} {{on "click" (fn @toggleRow student.reg)}} class="main-
row">

                    <td>

                        <input type="checkbox" />

                        <i class="bi bi-chevron-right chevron-icon {{if (includes @expandedRows student.reg)
" rotate"}}"/>

                    </td>

                    {{#each @items as |item|}}

                        {{#if (includes @visibleColumns item.key)}}

                            <td>

                                {{#if (eq item.key "skills")}}

                                    {{join-array student.skills " "}}

                                {{else}}

                                    {{get student item.key}}

                                {{/if}}

                            </td>

                        {{/if}}

                    {{/each}}

```



```

<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>

{{#if (includes @expandedRows student.reg)}}
<tr class="nested-row">

  <td class="nested-row-first-child"></td>

  <td colspan="7">

    <div class="nested-content">

      {{yield student}}

    </div>

  </td>

</tr>
{{/if}}

</VerticalCollection>

{{else}}

<tr>

  <td colspan="6" style="text-align: center;">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';
import { htmlSafe } from '@ember/template';

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(htmlSafe('<i class="bi bi-exclamation-triangle-fill"></i> 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

```
:root {  
  --bg-color: #ffffff;  
  --text-color: #353535;  
  --card-bg: #ffffff;  
  --input-bg: #ffffff;  
  --table-bg: #ffffff;  
  --table-header-bg: #f7f7f7;  
  --table-header-hover: #ebebeb;  
  --table-text: #3e3e3e;  
  --border-color: #dddddd;  
  --shadow-color: rgba(0, 0, 0, 0.1);  
  --navbar-bg: #ffffff;  
  --hover-bg: #f4f4f4;  
  --search-border: #ccc;  
}
```

```
[data-theme='dark'] {  
  --bg-color: #212121;  
  --text-color: #f1f1f1;  
  --card-bg: #2b2b2b;  
  --input-bg: #333333;  
  --table-bg: #272727;  
  --table-header-bg: #2b2b2b;  
  --table-header-hover: #3d3d3d;  
  --table-text: #e0e0e0;  
  --border-color: #3b3b3b;  
  --shadow-color: rgba(0, 0, 0, 0.6);  
}
```

```
--navbar-bg: #2e2e2e;

--hover-bg: #262626;

--search-border: #555555;
}

body {
  background-color: var(--bg-color) !important;
  color: var(--text-color);
  transition: all 0.3s ease;
  display: flex;
  align-items: center;
  flex-direction: column;
  font-family: Arial, Helvetica, sans-serif;
  height: 100vh;
}

.navbar {
  margin-bottom: 2rem;
  width: 100%;
  height: 4rem;
  background-color: var(--navbar-bg);
  box-shadow: 0 2px 8px var(--shadow-color);
  font-weight: bold;
  font-size: 20px;
  display: flex;
  justify-content: space-between;
  color: var(--text-color);
}
```

```
.navbar-title {  
  display: flex;  
  align-items: center;  
  gap: 5px;  
  padding: 0 2rem;  
  color: var(--text-color);  
  font-size: 18px;  
  font-weight: 600;  
}
```

```
#theme-toggle-btn {  
  background: transparent;  
  border: none;  
  font-size: 14px;  
  font-weight: 600;  
  display: flex;  
  justify-content: center;  
  align-items: center;  
}
```

```
#theme-toggle-btn i {  
  font-size: 20px;  
}
```

```
.toggle-profile {  
  display: flex;  
  gap: 1.5rem;
```

```
padding: 0 2rem;  
}
```

```
.toggle-profile i {  
    font-size: 24px;  
    cursor: pointer;  
    color: var(--table-text);  
  
}
```

```
.flash-message {  
    text-align: center;  
    min-width: 250px !important;  
    border-radius: 10px !important;  
    padding: 0.5rem !important;  
    z-index: 1050;  
    position: fixed !important;  
    top: 1rem;  
    right: 2rem;  
    opacity: 1;  
    animation: bounceIn 0.6s ease-out, fadeOut 0.5s ease-in 5s forwards;  
}
```

```
@keyframes bounceIn {  
    0% {  
        transform: scale(0.5);  
        opacity: 0;  
    }  
}
```



```
60% {  
    transform: scale(1.1);  
    opacity: 1;  
}
```

```
100% {  
    transform: scale(1);  
    opacity: 1;  
}  
}
```

```
@keyframes fadeOut {  
    to {  
        opacity: 0;  
        transform: translateY(-20px);  
    }  
}
```

```
.undo-btn {  
    background: transparent;  
    border: none;  
    border-bottom: 1px solid #000;  
    color: rgb(86, 86, 86);  
}
```

```
.student-container {  
    display: flex;
```

```
    align-items: center;
    flex-direction: column;
    border: 1px solid var(--border-color);
    border-radius: 10px;
    height: 80vh;
    width: 80%;
    background-color: var(--card-bg);
}
```

```
.full-screen-outlet {
    position: fixed;
    top: 0;
    left: 0;
    height: 100%;
    width: 100%;
    background: linear-gradient(to bottom, rgba(50, 50, 50, 0.7), rgba(45, 45, 45, 0.6));
    z-index: 1000;
}
```

```
[data-theme='dark'] .full-screen-outlet {
    background: linear-gradient(to bottom, rgba(67, 67, 67, 0.6), rgba(88, 88, 88, 0.6));
}
```

```
#formCont {
    display: flex;
    flex-direction: column;
    gap: 0.5rem;
    left: 35%;
```

```
background-color: var(--card-bg);
color: var(--text-color);
padding: 2.5rem;
border-radius: 20px;
width: 30%;
z-index: 2000;
position: relative;
margin: 4% 0;
}
```

```
label {
  font-weight: 600;
  color: var(--text-color);
  margin: 0.3rem 0;
  font-size: 14px;
}
```

```
input {
  padding: 0.5rem 1rem;
  border: 1px solid var(--border-color);
  background-color: var(--card-bg);
  color: var(--text-color);
  outline: none;
  height: 35px;
}
```

```
input:focus {
  border: 1px solid #a6a6a6;
```

```
}
```

```
input::placeholder {  
  color: var(--text-color);  
  opacity: 0.6;  
}
```

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

```
.ember-power-select-dropdown {  
  background-color: var(--card-bg) !important;  
  color: var(--text-color) !important;  
  border: 1px solid var(--border-color) !important;  
  box-shadow: 0 4px 8px var(--shadow-color) !important;  
}
```

```
.ember-power-select-option {  
  color: var(--text-color) !important;  
}
```

```
.ember-power-select-option[aria-selected='true'] {  
  background-color: var(--search-border) !important;  
  color: var(--text-color) !important;  
  opacity: 0.9 !important;  
}
```

```
.ember-power-select-option[aria-current='true'] {  
  background-color: var(--hover-bg) !important;  
  color: var(--text-color) !important;  
}
```

```
.ember-power-select-option:hover {  
  background-color: #4f8091 !important;  
  color: white !important;  
}
```

```
form .ember-power-select-trigger {  
  background-color: var(--card-bg) !important;  
  color: var(--text-color) !important;  
  border: 1px solid var(--border-color) !important;  
  border-radius: 5px;  
  scrollbar-width: none;  
}
```

```
.ember-power-select-placeholder {  
  color: var(--text-color) !important;  
  opacity: 0.6;  
}
```

```
.ember-power-select-multiple-option {  
  background-color: var(--hover-bg) !important;  
  color: var(--text-color) !important;  
  border: 1px solid var(--border-color) !important;
```

```
margin: 0 4px 4px 0;
padding: 2px 6px;
border-radius: 4px;
}
```

```
.ember-power-select-clear-btn {
  color: var(--text-color) !important;
}
```

```
.ember-basic-dropdown-content {
  background-color: var(--card-bg) !important;
  color: var(--text-color) !important;
  border: 1px solid var(--border-color) !important;
}
```

```
.table-container {
  overflow: auto;
  scroll-behavior: smooth;
  scrollbar-color: var(--border-color) transparent;
  width: 100%;
  box-shadow: 3px 2px 10px var(--shadow-color);
  background: var(--card-bg);
  border: 1px solid var(--border-color);
  scrollbar-width: thin;
}
```

```
table {
```

```
background-color: var(--table-bg);  
border-radius: 10px;  
border-collapse: collapse;  
color: var(--table-text);  
width: max-content;  
min-width: 1200px;  
table-layout: auto;  
}
```

```
.table-container thead {  
    position: sticky;  
    top: 0;  
    background-color: var(--table-header-bg);  
    color: var(--text-color);  
    text-align: center;  
    font-size: 14px;  
    z-index: 200;  
}
```

```
tbody {  
    overflow-y: auto;  
    width: 100%;  
    scrollbar-width: thin;  
    background-color: var(--table-bg);  
    color: var(--table-text);  
}
```

```
tr th {
```

```
padding: 0.6rem;  
margin: 0;  
position: relative;  
}
```

```
th:not(:first-child) {  
    min-width: 200px;  
}
```

```
th:hover {  
    background-color: var(--table-header-hover);  
    box-shadow: 2px 4px 8px rgba(0, 0, 0, 0.8);  
}
```

```
tr th i {  
    cursor: pointer;  
    margin-left: 0.5rem;  
    font-size: 12px;  
}
```

```
th.resizable {  
    position: relative;  
    user-select: none;  
}
```

```
th.resizable .resizer {  
    position: absolute;  
    right: 0;
```



```
    top: 0;
    width: 25px;
    color: black;
    height: 100%;
    cursor: ew-resize;
    z-index: 1;
}
```

```
.table-container tr td {
    border-bottom: 1px solid var(--border-color);
    padding: 0.8rem;
    font-size: 14px;
    text-align: center;
    z-index: 20;
    cursor: pointer;
}
```

```
table tr td:not(:first-child) {
    white-space: nowrap;
    overflow: hidden;
    text-overflow: ellipsis;
    max-width: 200px;
}
```

```
.main-row th:first-child,
.main-row td:first-child {
    background-color: var(--table-header-bg);
}
```

```
position: sticky;
left: 0;
top: 0;
z-index: 50;
width: 60px;
}
```

```
.main-row th:first-child input {
margin-right: 0.7rem
}
```

```
.main-row th:first-child::after,
.main-row td:first-child::after,
.nested-row-first-child::after
{
content: "";
position: absolute;
top: 0;
right: -5px;
width: 5px;
height: 100%;
background: linear-gradient(to right, rgba(0, 0, 0, 0.1), transparent);
pointer-events: none;
}
```

```
.nested-table thead th:hover {
background-color: #ccc;
}
```

```
.nested-table thead tr {  
    background-color: var(--search-border);  
}
```

```
.nested-table tr td {  
    background-color: var(--table-bg);  
}
```

```
.nested-row-first-child {  
    background-color: var(--table-header-bg);  
    position: sticky;  
    top: 0;  
    left: 0;  
    z-index: 10;  
}
```

```
.nested-content {  
    animation: fadeInDown 0.3s ease;  
}
```

```
@keyframes fadeInDown {  
    from {  
        opacity: 0;  
        transform: translateY(-20px);  
    }  
  
    to {
```

```
        opacity: 1;
        transform: translateY(0);
    }
}
```

```
.chevron-icon {
    transition: transform 0.3s ease;
    float: right;
    cursor: pointer;
    font-size: 12px;
}
```

```
.chevron-icon.rotate {
    transform: rotate(90deg);
}
```

```
.btn-cd {
    padding: 0.4rem 0.8rem;
    border-radius: 5px;
    border: none;
    background-color: #f8091;
    color: white;
    cursor: pointer;
    font-size: 12px;
    font-weight: 600;
    text-decoration: none;
}
```

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

```
.cancel,  
.uc {  
    margin: 1rem 0;  
}
```

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

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

```
.cancel {  
    background-color: var(--card-bg);  
    color: var(--text-color);  
    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: var(--text-color);
}
```

```
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: var(--text-color);  
    font-size: 20px;  
}
```

```
.btn-filter {  
    border: 1px solid var(--search-border);  
    background: transparent;  
    color: var(--text-color);  
    opacity: 0.8;  
    border-radius: 5px;  
    padding: 0.3rem 1rem;  
    font-size: 14px;  
    font-weight: 600;  
}
```

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

```
.advanced-filters button {  
    float: right;  
}
```

```
.filter {  
    margin: 1rem;  
}
```

```
.btn-cd:hover,  
.btn-filter:hover {  
    scale: 1.03;  
}
```

```
.advanced-filters,  
.customize-column {  
    position: absolute;  
    top: 100%;  
    z-index: 1000;  
    padding: 1rem;  
    background: var(--bg-color);  
    color: var(--text-color);  
    border: 1px solid #ccc;  
    border-radius: 8px;  
    box-shadow: 0 4px 8px rgba(0, 0, 0, 0.4);  
    margin-top: 0.4rem;  
}
```

```
.filter-section-header {  
    display: flex;  
    justify-content: space-between;  
    align-items: center;
```



```
}
```

```
ul {  
  list-style: none;  
  padding: 0;  
  margin: 0;  
}
```

```
ul li {  
  margin: 0;  
  padding-left: 0;  
}
```

```
ul li div {  
  display: flex;  
  justify-content: space-between;  
  padding: 0.3rem 1rem;  
}
```

```
.advanced-filters {  
  left: 12rem;  
}
```

```
.customize-column {  
  left: 0.5rem;  
  width: 400px;  
}
```

```
.customize-column-header {  
  display: flex;  
  align-items: center;  
  justify-content: flex-end;  
  gap: 4rem;  
  padding-right: 1rem;  
  width: 100%;  
}
```

```
.customize-column-rc {  
  display: flex;  
  justify-content: flex-end;  
  align-items: center;  
  gap: 1rem;  
}
```

```
.customize-desc {  
  color: var(--text-color);  
  opacity: 0.8;  
  font-size: 14px;  
  font-weight: 600;  
}
```

```
.close-modal i {  
  cursor: pointer;  
  font-size: 20px;  
}
```

```
.btn-reset {  
  border: none;  
  border-radius: 5px;  
  background: transparent;  
  border: 1px solid #d84949;  
  color: #d84949;  
  opacity: 0.9;  
  padding: 0.2rem 0.5rem;  
  cursor: pointer;  
  font-size: 12px;  
}
```

```
.filter-section {  
  display: flex;  
  flex-direction: column;  
  gap: 0.5rem;  
}
```

```
.filter-dept,  
.filter-skill,  
.custom-column {  
  display: grid;  
  grid-template-columns: 1fr 1fr 1fr;  
  gap: 1rem;  
  margin: 0.5rem;  
}
```

```
.title-cont {
```

```
display: flex;
justify-content: center;
align-items: center;
}
```

```
.search-dlt-cont {
width: 98%;
display: flex;
justify-content: space-between;
align-items: center;
margin: 1rem;
}
```

```
.search-dlt-cont span {
font-size: 20px;
font-weight: 700;
color: var(--text-color);
}
```

```
.search-cont {
display: flex;
align-items: center;
border: 1px solid var(--search-border);
border-radius: 5px;
width: 40%;
background-color: var(--input-bg);
padding: 0 1rem;
height: 33px;
```

```
}
```

```
.search-cont i {  
  font-size: 14px;  
  margin-right: 8px;  
  color: var(--text-color);  
  opacity: 0.7;  
}
```

```
#search {  
  flex: 1;  
  border: none;  
  background: transparent;  
  color: var(--text-color);  
  padding: 0.3rem;  
  min-width: 120px;  
}
```

```
.search-cont .ember-power-select-trigger {  
  background: transparent !important;  
  border: none !important;  
  height: 30px !important;  
  overflow-y: scroll !important;  
  scrollbar-width: none !important;  
  max-width: 170px !important;  
  min-width: 100px !important;  
}
```

```
.search-cont .ember-power-select-multiple-option {  
  background-color: var(--table-bg) !important;  
  color: var(--text-color) !important;  
  border: none !important;  
  margin: 0.2rem !important;  
  padding: 0 8px !important;  
  font-size: 10px !important;  
  border-radius: 20px !important;  
  font-weight: 500 !important;  
  display: inline-flex;  
  align-items: center;  
  gap: 4px;  
  box-shadow: 0 1px 3px rgba(0, 0, 0, 0.2);  
}
```

```
.search-cont .ember-power-select-placeholder {  
  font-size: 12px !important;  
  font-weight: 400 !important;  
  opacity: 0.7 !important;  
  color: var(--text-color) !important;  
  padding: 0.3rem 0.7rem;  
}
```

```
.ember-power-select-multiple-remove-btn {  
  font-size: 16px !important;  
}
```

```
.search-cd-cont {
```

```
display: flex;
justify-content: flex-end;
align-items: center;
width: 80%;
gap: 0.5rem;
position: relative;
}
```