

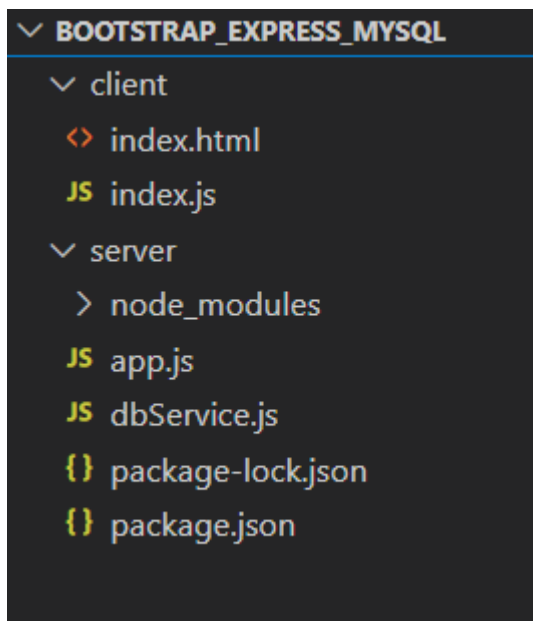
Assignment: CRUD Bootstrap + Express + MySQL

Aim: Create html form using following information

1. Course-id, Course Name [text box]
2. Department Name [Radio Button]
3. Institute Name [Check Box]
4. University Name [Drop Down]

Display all Record into dynamic table with edit and delete operation.

File Structure:



app.js

```
const express = require('express');
const app = express();
const cors = require('cors'); //Cross-Origin Resource Sharing vs Same-Origin Policy (SOP).
const dotenv = require('dotenv');
dotenv.config();

const DBService = require('./dbService');

app.use(cors());
app.use(express.json());
app.use(express.urlencoded({ extended : false }));

// create
app.post('/insert', (request, response) => {
  const { courseId, courseName, departmentName, instituteName, universityName } = request.body;
```

```
const db = DBService.getDbServiceInstance();

const result = db.insertNewCourse(courseId, courseName, departmentName, instituteName, universityName);
result
  .then(data => response.json({ data: data}))
  .catch(err => console.log(err));
});

// read
app.get('/getAll', (request, response) => {
  const db = DBService.getDbServiceInstance();

  const result = db.getAllData();

  result
    .then(data => response.json({data : data}))
    .catch(err => console.log(err));
})

// delete
app.delete('/delete/:id', (request, response) => {
  const { id } = request.params;
  const db = DBService.getDbServiceInstance();

  const result = db.deleteRowById(id);

  result
    .then(data => response.json({success : data}))
    .catch(err => console.log(err));
});

// update
app.patch('/update', (request, response) => {
  const { id, courseId, courseName, departmentName, instituteName, universityName } = request.body;
  const db = DBService.getDbServiceInstance();

  const result = db.updateCourse(id, courseId, courseName, departmentName, instituteName, universityName);

  result
    .then(data => response.json({success : data}))
    .catch(err => console.log(err));
});

app.listen(5000,()=>{
  console.log("Server is running on port number 5000")
})
```

```
}}
```

dbService.js

```
const mysql = require('mysql');
const dotenv = require('dotenv');
let instance = null;

const connection = mysql.createConnection({
  host: 'localhost',
  user: 'root',
  password: '',
  database: 'course'
});

connection.connect((err) => {
  if (err) {
    console.log(err.message);
  }
});

class DBService{
  static getDbServiceInstance() {
    return instance ? instance : new DBService();
  }

  async getAllData() {
    try {
      const response = await new Promise((resolve, reject) => {
        const query = "SELECT * FROM courseinfo;";

        connection.query(query, (err, results) => {
          if (err) reject(new Error(err.message));
          resolve(results);
        })
      });
      return response;
    } catch (error) {
      console.log(error);
    }
  }

  async insertNewCourse(courseId, courseName, departmentName, instituteName,
    universityName) {
    try {
      const dateAdded = new Date();
      const insertId = await new Promise((resolve, reject) => {
```

```
        const query = "INSERT INTO courseinfo (courseId, courseName, departmentName, instituteName, universityName, dateAdded) VALUES (?, ?, ?, ?, ?, ?);";
    };

    connection.query(query, [courseId, courseName, departmentName, instituteName, universityName, dateAdded], (err, result) => {
        if (err) reject(new Error(err.message));
        resolve(result.insertId);
    })
    });
    return {
        id : insertId, courseId
        : courseId, courseName:
        courseName,
        departmentName: departmentName,
        instituteName: instituteName,
        universityName: universityName,
        dateAdded : dateAdded
    };
} catch (error) {
    console.log(error);
}
}

async deleteRowById(id) {
    try {
        id = parseInt(id, 10); //base decimal
        const response = await new Promise((resolve, reject) => {
            const query = "DELETE FROM courseinfo WHERE id = ?";

            connection.query(query, [id], (err, result) => {
                if (err) reject(new Error(err.message));
                resolve(result.affectedRows);
            })
        });

        return response === 1 ? true : false;
    } catch (error) {
        console.log(error);
        return false;
    }
}

async updateCourse(id, courseId, courseName, departmentName, instituteName, universityName) {
    try {
        id = parseInt(id, 10);
        const response = await new Promise((resolve, reject) => {
```

```

        const query = "UPDATE courseinfo SET courseId = ?, courseName
= ?, departmentName = ?, instituteName = ?, universityName = ? WHERE id = ?";
        connection.query(query, [courseId, courseName, departmentName,
instituteName, universityName, id] , (err, result) => {
            if (err) reject(new Error(err.message));
            resolve(result.affectedRows);
        })
    });

    return response === 1 ? true : false;
} catch (error) {
    console.log(error);
    return false;
}
}
}

module.exports = DBService;

```

index.html

```

<!DOCTYPE html>
<html lang="en">

<head>
    <meta charset="UTF-8">
    <meta http-equiv="X-UA-Compatible" content="IE=edge">
    <meta name="viewport" content="width=device-width, initial-scale=1.0">
    <title>CRUD Operation Using Express JS and MySQL</title>
    <link href="https://cdn.jsdelivr.net/npm/bootstrap@5.1.1/dist/css/bootstrap.min.css" rel="stylesheet"
        integrity="sha384-
F3w7mX95PdgyTmZZMECAngseQB83DfGTowi0iMjiWaeVhAn4FJkqJByhZMI3AhiU" crossorigin=
"anonymous">
    <script src="https://cdn.jsdelivr.net/npm/bootstrap@5.1.1/dist/js/bootstrap.bundle.min.js"
        integrity="sha384-
/bQdsTh/da6pkI1MST/rWKFJjaCP5gBSY4sEBT38Q/9RBh9AH40zEOg7Hlq2THRZ"
        crossorigin="anonymous"></script>
</head>

<body>
    <div class="row">
        <div class="col-md-12 mt-3 d-flex justify-content-center">
            <div class="card card-primary">
                <div class="card-header">
                    <h3 class="card-title">
                        Add Course Details

```

```

        </h3>
    </div>

    <div class="card-body">
        <div class="form-group row">
            <div class="col-md-2">
                <label for="courseId" class="text-md col-form-
label">Course Id</label>
            </div>

            <div class="col-md-3">
                <input type="text" class="form-
control" id="courseId" maxlength="10"
                placeholder="Enter Course Id...">
            </div>

            <div class="col-md-3">
                <label for="courseName" class="text-md col-form-
label">Course Name</label>
            </div>

            <div class="col-md-4">
                <input type="text" class="form-
control" id="courseName" maxlength="50"
                placeholder="Enter Course Name...">
            </div>
        </div>

        <div class="form-group row mt-2">
            <div class="col-md-3">
                <label for="departmentName" class="text-md col-
form-label">Department Name</label>
            </div>

            <div class="form-check col-md-2 mt-2">
                <input class="form-check-
input" type="radio" id="CSE" name="departmentName" value="CSE">
                <label class="form-check-label" for="CSE">
                    CSE
                </label>
            </div>

            <div class="form-check col-md-2 mt-2">
                <input class="form-check-
input" type="radio" id="CE" name="departmentName" value="CE">
                <label class="form-check-label" for="CE">
                    CE
                </label>
            </div>
        </div>
    </div>

```

```

        </div>

        <div class="form-check col-md-2 mt-2">
            <input class="form-check-
input" type="radio" id="IT" name="departmentName" value="IT">
            <label class="form-check-label" for="IT">
                IT
            </label>
        </div>
    </div>

    <div class="form-group row mt-2">
        <div class="col-md-3">
            <label for="instituteName" class="text-md col-
form-label">Institute Name</label>
        </div>

        <div class="form-check col-md-2 mt-2">
            <input class="form-check-
input" type="checkbox" id="CSPIT" value="CSPIT">
            <label class="form-check-label" for="CSPIT">
                CSPIT
            </label>
        </div>

        <div class="form-check col-md-2 mt-2">
            <input class="form-check-
input" type="checkbox" id="DEPSTAR" value="DEPSTAR">
            <label class="form-check-label" for="DEPSTAR">
                DEPSTAR
            </label>
        </div>
    </div>

    <div class="form-group row mt-2">
        <div class="col-md-3">
            <label for="univeristyName" class="text-md col-
form-label">University Name</label>
        </div>

        <div class="col-md-4">
            <select name="universityName" id="universityName"
class="form-select">
                <option value="-1">Select University</option>
                <option value="1">CHARUSAT</option>
                <option value="2">Nirma</option>
                <option value="3">DAIICT</option>
                <option value="4">PDPU</option>
            </select>
        </div>
    </div>

```

```

        </select>
      </div>
    </div>
  </div>

  <div class="card-footer">
    <button type="submit" value="submit" id="addCourse" class=
"btn btn-primary">Add</button>
  </div>
</div>
</div>
</div>

<div class="row">
  <div class="col-md-12 mt-3 d-flex justify-content-center">
    <div class="card card-primary">
      <div class="card-header">
        <h3 class="card-title">
          Manage Course Details
        </h3>
      </div>

      <div class="card-body table-responsive">
        <table class="table table-bordered" id="courseDetails">
          <thead>
            <tr>
              <th>ID</th>
              <th>Course Id</th>
              <th>Course Name</th>
              <th>Department Name</th>
              <th>Institute Name</th>
              <th>University Name</th>
              <th>Date Added</th>
              <th>Edit</th>
              <th>Delete</th>
            </tr>
          </thead>
          <tbody></tbody>
        </table>
      </div>
    </div>
  </div>
</div>

<section hidden id="updateRow">
  <div class="row">
    <div class="col-md-12 mt-3 d-flex justify-content-center">
      <div class="card card-primary">

```



```

        <div class="card-header">
            <h3 class="card-title">
                Edit Course Details
            </h3>
        </div>

        <div class="card-body">
            <div class="form-group row">
                <div class="col-md-2">
                    <label for="editCourseId" class="text-md col-
form-label">Course Id</label>
                </div>

                <div class="col-md-3">
                    <input type="text" class="form-
control" id="editCourseId" maxLength="10"
                        placeholder="Enter Course Id...">
                </div>

                <div class="col-md-3">
                    <label for="editCourseName" class="text-
md col-form-label">Course Name</label>
                </div>

                <div class="col-md-4">
                    <input type="text" class="form-
control" id="editCourseName" maxLength="50"
                        placeholder="Enter Course Name...">
                </div>
            </div>

            <div class="form-group row mt-2">
                <div class="col-md-3">
                    <label for="editDepartmentName" class="text-
md col-form-label">Department Name</label>
                </div>

                <div class="form-check col-md-2 mt-2">
                    <input class="form-check-
input" type="radio" id="editCSE" name="editDepartmentName" value="CSE">
                    <label class="form-check-label" for="CSE">
                        CSE
                    </label>
                </div>

                <div class="form-check col-md-2 mt-2">
                    <input class="form-check-
input" type="radio" id="editCE" name="editDepartmentName" value="CE">

```

```

        <label class="form-check-label" for="CE">
            CE
        </label>
    </div>

    <div class="form-check col-md-2 mt-2">
        <input class="form-check-
input" type="radio" id="editIT" name="editDepartmentName" value="IT">
        <label class="form-check-label" for="IT">
            IT
        </label>
    </div>
</div>

<div class="form-group row mt-2">
    <div class="col-md-3">
        <label for="editInstituteName" class="text-
md col-form-label">Institute Name</label>
    </div>

    <div class="form-check col-md-2 mt-2">
        <input class="form-check-
input" type="checkbox" id="editCSPIT" value="CSPIT">
        <label class="form-check-label" for="CSPIT">
            CSPIT
        </label>
    </div>

    <div class="form-check col-md-2 mt-2">
        <input class="form-check-
input" type="checkbox" id="editDEPSTAR" value="DEPSTAR">
        <label class="form-check-label" for="DEPSTAR">
            DEPSTAR
        </label>
    </div>
</div>

<div class="form-group row mt-2">
    <div class="col-md-3">
        <label for="editUniveristyName" class="text-
md col-form-label">University Name</label>
    </div>

    <div class="col-md-4">
        <select name="editUniversityName" id="editUniv
ersityName" class="form-select">
            <option value="-
1">Select University</option>

```

```

        <option value="1">CHARUSAT</option>
        <option value="2">Nirma</option>
        <option value="3">DAIICT</option>
        <option value="4">PDPU</option>
    </select>
</div>
</div>
</div>

    <div class="card-footer">
        <button type="submit" value="submit" id="updateCourse"
class="btn btn-primary">Update</button>
    </div>
</div>
</div>
</div>
</section>
<script src="index.js"></script>
</body>

</html>

```

index.js

```

document.addEventListener('DOMContentLoaded', function () {
    fetch('http://localhost:5000/getAll')
        .then(response => response.json())
        .then(data => loadHTMLTable(data['data']));
});

document.querySelector('#courseDetails tbody').addEventListener('click', function (event) {
    if (event.target.className === "deleteCourse btn btn-danger") {
        deleteRowById(event.target.dataset.id);
    }
    if (event.target.className === "editCourse btn btn-primary") {
        handleEditRow(event.target.dataset.id);
    }
});

const updateBtn = document.querySelector('#updateCourse');

const addBtn = document.querySelector('#addCourse');

addBtn.onclick = function () {
    courseIdInput = document.querySelector('#courseId').value;
    console.log(courseIdInput);
};

```

```
courseNameInput = document.querySelector('#courseName').value;
console.log(courseNameInput);
departmentNameInput = "";
if (document.getElementById('CSE').checked == true) {
    departmentNameInput = "CSE";
} else if (document.getElementById('CE').checked == true) {
    departmentNameInput = "CE";
} else if (document.getElementById('IT').checked == true) {
    departmentNameInput = "IT";
}
console.log(departmentNameInput);

instituteNameInput = "";
if (document.getElementById('CSPIT').checked == true && document.getElementById('DEPSTAR').checked == true) {
    instituteNameInput = "CSPIT, DEPSTAR";
} else if (document.getElementById('DEPSTAR').checked == true) {
    instituteNameInput = "DEPSTAR";
} else if (document.getElementById('CSPIT').checked == true) {
    instituteNameInput = "CSPIT";
}
console.log(instituteNameInput);
universityNameInput = document.getElementById('universityName');
universityNameInput = universityNameInput.options[universityNameInput.selectedIndex].text;

console.log(universityNameInput);

fetch('http://localhost:5000/insert', {
    headers: {
        'Content-type': 'application/json'
    },
    method: 'POST',
    body: JSON.stringify({
        courseId: courseIdInput,
        courseName: courseNameInput,
        departmentName: departmentNameInput,
        instituteName: instituteNameInput,
        universityName: universityNameInput
    })
})
.then(response => response.json())
.then(data => insertRowIntoTable(data['data']))
.then(() => {
    document.getElementById('courseId').value = "";
    document.getElementById('courseName').value = "";
    var ele = document.getElementsByName("departmentName");
    for (var i = 0; i < ele.length; i++)
```

```

        ele[i].checked = false;

        document.getElementById('CSPIT').checked = false;
        document.getElementById('DEPSTAR').checked = false;

        document.getElementById('universityName').value = '-1';
    });
}

function deleteRowById(id) {
    fetch('http://localhost:5000/delete/' + id, {
        method: 'DELETE'
    })
    .then(response => response.json())
    .then(data => {
        if (data.success) {
            location.reload();
        }
    });
}

function handleEditRow(id) {
    const updateSection = document.querySelector('#updateRow');
    updateSection.hidden = false;
    document.querySelector('#editCourseId').dataset.id = id;
}

updateBtn.onclick = function() {
    updateCourseId = document.querySelector('#editCourseId');
    updateCourseName = document.querySelector('#editCourseName').value;

    updateDepartmentName = "";
    if (document.getElementById('editCSE').checked == true) {
        updateDepartmentName = "CSE";
    } else if (document.getElementById('editCE').checked == true) {
        updateDepartmentName = "CE";
    } else if (document.getElementById('editIT').checked == true) {
        updateDepartmentName = "IT";
    }

    updateInstituteName = "";
    if (document.getElementById('editCSPIT').checked == true && document.getElementById('editDEPSTAR').checked == true) {
        updateInstituteName = "CSPIT, DEPSTAR";
    } else if (document.getElementById('editDEPSTAR').checked == true) {
        updateInstituteName = "DEPSTAR";
    } else if (document.getElementById('editCSPIT').checked == true) {
        updateInstituteName = "CSPIT";
    }
}

```

```

    }

    updateUniversityName = document.getElementById('editUniversityName');
    updateUniversityName = updateUniversityName.options[updateUniversityName.selectedIndex].text;

    fetch('http://localhost:5000/update', {
      method: 'PATCH',
      headers: {
        'Content-type' : 'application/json'
      },
      body: JSON.stringify({
        id: updateCourseId.dataset.id,
        courseId: updateCourseId.value,
        courseName: updateCourseName,
        departmentName: updateDepartmentName,
        instituteName: updateInstituteName,
        universityName: updateUniversityName
      })
    })
    .then(response => response.json())
    .then(data => {
      if (data.success) {
        location.reload();
      }
    })
  })
}

function insertRowIntoTable(data) {
  console.log(data);
  const table = document.querySelector('table tbody');
  const isTableData = table.querySelector('.no-data');

  let tableHtml = "<tr>";

  for (var key in data) {
    if (data.hasOwnProperty(key)) {
      if (key === 'dateAdded') {
        data[key] = new Date(data[key]).toLocaleString('en-US', { timeZone: 'UTC' }); // ko-KR toLocaleString('en-GB', { timeZone: 'UTC' })
      }
      tableHtml += `<td>${data[key]}</td>`;
    }
  }

  tableHtml += `<td><button class="editCourse btn btn-primary" data-id=${data.id}>Edit</td>`;

```

```

        tableHtml += `<td><button class="deleteCourse btn btn-danger" data-
id=${data.id}>Delete</td>`;

        tableHtml += "</tr>";

        if (isTableData) {
            table.innerHTML = tableHtml;
        } else {
            const newRow = table.insertRow();
            newRow.innerHTML = tableHtml;
        }
    }

function loadHTMLTable(data) {
    const table = document.querySelector('table tbody');

    if (data.length === 0) {
        table.innerHTML = "<tr><td class='no-
data' colspan='5'>No Data</td></tr>";
        return;
    }

    let tableHtml = "";

    data.forEach(function ({ id, courseId, courseName, departmentName, institu
teName, universityName, dateAdded }) {
        tableHtml += "<tr>";
        tableHtml += `<td>${id}</td>`;
        tableHtml += `<td>${courseId}</td>`;
        tableHtml += `<td>${courseName}</td>`;
        tableHtml += `<td>${departmentName}</td>`;
        tableHtml += `<td>${instituteName}</td>`;
        tableHtml += `<td>${universityName}</td>`;
        tableHtml += `<td>${new Date(dateAdded).toLocaleString()}</td>`;
        tableHtml += `<td><button class="editCourse btn btn-primary" data-
id=${id}>Edit</td>`;
        tableHtml += `<td><button class="deleteCourse btn btn-danger" data-
id=${id}>Delete</td>`;
        tableHtml += "</tr>";
    });

    table.innerHTML = tableHtml;
}

```

Output:

```

Server is running on port number 5000

```

Before Deleting Info:

Add Course Details

Course Id Course Name

Department Name ☐ CSE ☐ CE ☐ IT

Institute Name ☐ CSPIT ☐ DEPSTAR

University Name

Manage Course Details

ID	Course Id	Course Name	Department Name	Institute Name	University Name	Date Added	Edit	Delete
1	CE377	AWT	CE	CSPIT, DEPSTAR	CHARUSAT	9/25/2021, 12:00:00 AM	<input type="button" value="Edit"/>	<input type="button" value="Delete"/>
2	CE355	DAA	CSE	CSPIT	Nirma	9/25/2021, 12:00:00 AM	<input type="button" value="Edit"/>	<input type="button" value="Delete"/>
3	CE354	OS	IT	DEPSTAR	DAIICT	9/25/2021, 12:00:00 AM	<input type="button" value="Edit"/>	<input type="button" value="Delete"/>
4	CE343	SE	CSE	CSPIT	PDPU	9/25/2021, 12:00:00 AM	<input type="button" value="Edit"/>	<input type="button" value="Delete"/>
5	CE356	SGP	CE	DEPSTAR	CHARUSAT	9/25/2021, 12:00:00 AM	<input type="button" value="Edit"/>	<input type="button" value="Delete"/>

After Deleting Info:

Add Course Details

Course Id Course Name

Department Name ☐ CSE ☐ CE ☐ IT

Institute Name ☐ CSPIT ☐ DEPSTAR

University Name

Manage Course Details

ID	Course Id	Course Name	Department Name	Institute Name	University Name	Date Added	Edit	Delete
1	CE377	AWT	CE	CSPIT, DEPSTAR	CHARUSAT	9/25/2021, 12:00:00 AM	<input type="button" value="Edit"/>	<input type="button" value="Delete"/>
2	CE355	DAA	CSE	CSPIT	Nirma	9/25/2021, 12:00:00 AM	<input type="button" value="Edit"/>	<input type="button" value="Delete"/>
3	CE354	OS	IT	DEPSTAR	DAIICT	9/25/2021, 12:00:00 AM	<input type="button" value="Edit"/>	<input type="button" value="Delete"/>
5	CE356	SGP	CE	DEPSTAR	CHARUSAT	9/25/2021, 12:00:00 AM	<input type="button" value="Edit"/>	<input type="button" value="Delete"/>

After Updating:

Add Course Details

Course Id	<input type="text" value="Enter Course Id..."/>	Course Name	<input type="text" value="Enter Course Name..."/>
Department Name	<input type="radio"/> CSE <input type="radio"/> CE <input type="radio"/> IT		
Institute Name	<input type="checkbox"/> CSPIT <input type="checkbox"/> DEPSTAR		
University Name	<input type="text" value="Select University"/> ▼		
<input type="button" value="Add"/>			

Manage Course Details

ID	Course Id	Course Name	Department Name	Institute Name	University Name	Date Added	Edit	Delete
1	CE377	AWT	CE	CSPIT, DEPSTAR	CHARUSAT	9/25/2021, 12:00:00 AM	<input type="button" value="Edit"/>	<input type="button" value="Delete"/>
2	CE355	DAA	CSE	CSPIT	Nirma	9/25/2021, 12:00:00 AM	<input type="button" value="Edit"/>	<input type="button" value="Delete"/>
3	CE354	OS	IT	DEPSTAR	DAIICT	9/25/2021, 12:00:00 AM	<input type="button" value="Edit"/>	<input type="button" value="Delete"/>
5	CE356	SGP	CE	CSPIT, DEPSTAR	CHARUSAT	9/25/2021, 12:00:00 AM	<input type="button" value="Edit"/>	<input type="button" value="Delete"/>