**Assignment 3**

1. When Page loads Display your **Full Name** and **NUID**
2. Table should not be expanded when page loads
3. Submit button to be disabled and grayed out and it should not be clickable
4. Add New Student Button should add new student with dummy values and dummy values should be dynamically added e.g for row 4 the values will be Student 4, Teacher 4 etc. and for row 5 the values will be Student 5, Teacher 5 etc
5. For newly added rows make sure existing CSS are applied too
6. After successful add of the record display the pop-up message with newly added student name in the popup message e.g. if you have added row 4 for student 4 then display “**Student 4 Record added successfully**” and display error message in case add of record addition failed.

A picture containing text, screenshot, font, number

Description automatically generated

1. On **Selecting** any of the **checkboxes**,  
   i] Background color to be changed for the particular row to “Yellow” and   
   ii] Submit button to turn orange  
   iii]A delete button should get added **dynamically** in **DELETE column** and should delete the entire row when clicked. Also, pop-up message should be displayed by having student name in the popup message e.g. if you have deleted row 4 for student 4 then display “**Student 4 Record deleted successfully**”

iv) An Edit button should get added **dynamically** in **EDIT COLUMN** and should only display pop up box with message e.g. if you are editing student 2 details then edit pop up should display as, **title** of Edit pop up should be “**Edit details of Student 2**” and it should display all the fields of student 2 row in the pop up details as text with “**Update**” and “**Cancel**” buttons. Data will not be editable as it is text format but when “**Update**” button is clicked then display “**Student 2 data updated successfully**” and when “**Cancel**” button is clicked pop up will be closed

A screenshot of a computer

Description automatically generated with low confidence

1. On **Deselecting** the checkbox will
2. make the row background color white again.
3. Also, if none of the rows are selected, the Submit button should be Grayed out again(disabled). (i.e., if 1 or more of the checkboxes are selected, the Submit button will be enabled and background color will be yellow. If none are selected, it will be disabled and background set to Gray)
4. DELETE button in those de-selected rows will be **hidden**
5. EDIT button in those de-selected rows will be **hidden**

A picture containing text, screenshot, font, number

Description automatically generated

1. Clicking on any of the green arrows will expand the particular row. Clicking it again will collapse it (i.e., toggle view on click)

A picture containing text, screenshot, number, font

Description automatically generated

1. Assignment must be submitted on your **PRIVATE** **GitHub account (use Northeastern email account)** and **zip** file must be submitted on **CANVAS** and github URL of assignment in the canvas remarks

let sum = 0;

const table = document.getElementById("myTable");

const checkboxR = table.getElementsByTagName("input");

configure();

checked();

let boolean = true;

let check = true;

function checked() {

sum = 0;

for (let i = 0; i < checkboxR.length; i++) {

const row = checkboxR[i].parentNode.parentNode;

checkboxR[i].addEventListener("click", () => {

if (!checkboxR[i].checked) {

sum--;

row.style.backgroundColor = "white";

row.lastElementChild.innerHTML = "";

row.lastElementChild.previousElementSibling.innerHTML = "";

hideColumns(row);

} else {

sum++;

row.style.backgroundColor = "yellow";

row.lastElementChild.innerHTML = "<td><button onClick='popUpDetails(this)'>Edit</button></td>";

row.lastElementChild.previousElementSibling.innerHTML = "<td><button onClick='deleteRow(this)'>Delete</button></td>";

showColumns(row);

}

updateButtonState();

});

}

}

function deleteRow(row) {

const studentName = row.parentNode.parentNode.querySelector("td:nth-child(2)").textContent;

const i = row.parentNode.parentNode.rowIndex;

document.getElementById("myTable").deleteRow(i);

alert(studentName + " data deleted successfully");

sum--;

configure();

checked();

}

function insert() {

const row = table.insertRow(table.rows.length);

const rowsum = table.rows.length;

const columnCheckbox = row.insertCell(0);

const columnStudent = row.insertCell(1);

const columnAdvisor = row.insertCell(2);

const ColumnA = row.insertCell(3);

const columnS = row.insertCell(4);

const columnpe = row.insertCell(5);

const columnCost = row.insertCell(6);

const columnper = row.insertCell(7);

const buttondel = row.insertCell(8);

const btne = row.insertCell(9);

columnCheckbox.innerHTML = `<td><input type="checkbox" /><br /><br /><img onClick="display(this)" src="down.png" width="25px" /></td>`;

columnStudent.innerHTML = `Student ${Math.ceil(rowsum / 2)}`;

columnAdvisor.innerHTML = `Teacher ${Math.ceil(rowsum / 2)}`;

ColumnA.innerHTML = "Approved";

columnS.innerHTML = "Fall";

columnpe.innerHTML = " TA ";

columnCost.innerHTML = Math.ceil(Math.random() \* 10000);

columnper.innerHTML = "100%";

try {

setTimeout(() => { alert(`Successfully Added Student${Math.ceil(rowsum / 2)}`) }, 100);

} catch (error) {

alert("Adding Student${Math.ceil(rowsum / 2)} failed!");

}

checked();

displayAdd();

configure();

}

window.addEventListener("click", () => {

updateButtonState();

});

function display(r) {

const row = r.parentNode.parentNode;

const d = row.nextSibling;

if (!check) {

d.style.display = "none";

check = true;

} else {

d.style.display = "table-row";

check = false;

}

}

function popUpDetails(row) {

const studentName = row.parentNode.parentNode.querySelector("td:nth-child(2)").textContent;

const editPopupTitle = "Edit details of " + studentName;

const editPopupContent =

"Student Name: " + studentName + "\n" +

"Advisor: " + row.parentNode.parentNode.querySelector("td:nth-child(3)").textContent + "\n" +

"Award Status: " + row.parentNode.parentNode.querySelector("td:nth-child(4)").textContent + "\n" +

"Semester: " + row.parentNode.parentNode.querySelector("td:nth-child(5)").textContent + "\n" +

"Type: " + row.parentNode.parentNode.querySelector("td:nth-child(6)").textContent + "\n" +

"Budget: " + row.parentNode.parentNode.querySelector("td:nth-child(7)").textContent + "\n" +

"Percentage: " + row.parentNode.parentNode.querySelector("td:nth-child(8)").textContent;

const confirmUpdate = confirm(editPopupTitle + "\n\n" + editPopupContent + "\n\nPress OK to update or Cancel to close.");

if (confirmUpdate) {

alert(studentName + " updated successfully");

checked();

}

}

function displayAdd() {

const row = table.insertRow(table.rows.length);

row.classList.add("dropDownTextArea");

row.innerHTML =

'<td colspan="10"> \

Advisor:<br /><br /> \

Award Details<br /> \

Summer 1-2014(TA)<br /> \

Budget Number: <br /> \

Tuition Number: <br /> \

Comments:<br /><br /><br /> \

Award Status:<br /><br /><br /> \

</td>';

configure();

checked();

}

function configure() {

const nameElement = document.getElementById("name");

const nuidElement = document.getElementById("nuid");

nameElement.textContent = "Full Name: Saheeshna Kamuni";

nuidElement.textContent = "NUID: 002640603";

}

function hideColumns(row) {

document.querySelectorAll("tr")[0].querySelectorAll("th")[8].classList.add("hideColumn");

document.querySelectorAll("tr")[0].querySelectorAll("th")[9].classList.add("hideColumn");

row.querySelectorAll("td")[8].classList.add("hideColumn");

row.querySelectorAll("td")[9].classList.add("hideColumn");

}

function showColumns(row) {

document.querySelectorAll("tr")[0].querySelectorAll("th")[8].classList.remove("hideColumn");

document.querySelectorAll("tr")[0].querySelectorAll("th")[9].classList.remove("hideColumn");

row.querySelectorAll("td")[8].classList.remove("hideColumn");

row.querySelectorAll("td")[9].classList.remove("hideColumn");

}

function updateButtonState() {

let btnSelect = document.getElementById("button");

if (sum <= 0) {

btnSelect.style.backgroundColor = "gray";

btnSelect.style.cursor = "initial";

btnSelect.style.border = "5px solid gray";

btnSelect.disabled = true;

} else {

btnSelect.style.backgroundColor = "orange";

btnSelect.style.cursor = "pointer";

btnSelect.style.border = "5px solid orange";

btnSelect.disabled = false;

}

}

let sum = 0;

const table = document.getElementById("myTable");

const checkboxR = table.getElementsByTagName("input");

configure();

checked();

let boolean = true;

let check = true;

function checked() {

  sum = 0;

  for (let i = 0; i < checkboxR.length; i++) {

    const row = checkboxR[i].parentNode.parentNode;

    checkboxR[i].addEventListener("click", () => {

      if (!checkboxR[i].checked) {

        sum--;

        row.style.backgroundColor = "white";

        row.lastElementChild.innerHTML = "";

        row.lastElementChild.previousElementSibling.innerHTML = "";

        document.querySelectorAll("tr")[0].querySelectorAll("th")[8].classList.add("hideColumn");

        document.querySelectorAll("tr")[0].querySelectorAll("th")[9].classList.add("hideColumn");

        row.querySelectorAll("td")[8].classList.add("hideColumn");

        row.querySelectorAll("td")[9].classList.add("hideColumn");

      } else {

        sum++;

        row.style.backgroundColor = "yellow";

        row.lastElementChild.innerHTML = "<td><button onClick='popUpDetails(this)'>Edit</button></td>";

        row.lastElementChild.previousElementSibling.innerHTML = "<td><button onClick='deleteRow(this)'>Delete</button></td>";

        document.querySelectorAll("tr")[0].querySelectorAll("th")[8].classList.remove("hideColumn");

        document.querySelectorAll("tr")[0].querySelectorAll("th")[9].classList.remove("hideColumn");

        row.querySelectorAll("td")[8].classList.remove("hideColumn");

        row.querySelectorAll("td")[9].classList.remove("hideColumn");

      }

      updateButtonState();

    })

  }

}

function deleteRow(row) {

  const studentName = row.parentNode.parentNode.querySelector("td:nth-child(2)").textContent;

  const i = row.parentNode.parentNode.rowIndex;

  document.getElementById("myTable").deleteRow(i);

  alert(studentName + " data deleted successfully");

  document.querySelectorAll("tr")[0].querySelectorAll("th")[8].classList.add("hideColumn");

  document.querySelectorAll("tr")[0].querySelectorAll("th")[9].classList.add("hideColumn");

  console.log("delete row" + sum)

  console.log("clength:" + checkboxR.length);

  sum--;

  configure();

  checked();

}

function insert() {

  const row = table.insertRow(table.rows.length);

  const rowsum = table.rows.length;

  const columnCheckbox = row.insertCell(0);

  const columnStudent = row.insertCell(1);

  const columnAdvisor = row.insertCell(2);

  const ColumnA = row.insertCell(3);

  const columnS = row.insertCell(4);

  const columnpe = row.insertCell(5);

  const columnCost = row.insertCell(6);

  const columnper = row.insertCell(7);

  const buttondel = row.insertCell(8);

  const btne = row.insertCell(9);

  columnCheckbox.innerHTML = `<td><input type="checkbox" /><br /><br /><img onClick="display(this)" src="down.png" width="25px" /></td>`;

  columnStudent.innerHTML = `Student ${Math.ceil(rowsum / 2)}`;

  columnAdvisor.innerHTML = `Teacher ${Math.ceil(rowsum / 2)}`;

  ColumnA.innerHTML = "Approved";

  columnS.innerHTML = "Fall";

  columnpe.innerHTML = " TA ";

  columnCost.innerHTML = Math.ceil(Math.random() \* 10000);

  columnper.innerHTML = "100%";

  try {

    setTimeout(() => { alert(`Successfully Added Student${Math.ceil(rowsum / 2)}`) }, 100);

  } catch (error) {

    alert("Adding Student${Math.ceil(rowsum / 2)} failed!");

  }

  checked();

  displayAdd();

  configure();

}

window.addEventListener("click", () => {

  let btnSelect = document.getElementById("button");

  if (sum <= 0) {

    btnSelect.style.backgroundColor = "gray";

    btnSelect.style.cursor = "initial";

    btnSelect.style.border = "5px solid gray";

    btnSelect.style.border = "none";

    btnSelect.disabled = true;

  } else {

    btnSelect.style.backgroundColor = "orange";

    btnSelect.style.cursor = "pointer";

    btnSelect.style.border = "5px solid orange";

    btnSelect.disabled = false;

  }

}

)

function display(r) {

  const row = r.parentNode.parentNode;

  const d = row.nextSibling;

  if (!check) {

    d.style.display = "none";

    check = true;

  } else {

    d.style.display = "table-row";

    check = false;

  }

}

function popUpDetails(row) {

  const studentName = row.parentNode.parentNode.querySelector("td:nth-child(2)").textContent;

  const editPopupTitle = "Edit details of " + studentName;

  const editPopupContent =

    "Student Name: " + studentName + "\n" +

    "Advisor: " + row.parentNode.parentNode.querySelector("td:nth-child(3)").textContent + "\n" +

    "Award Status: " + row.parentNode.parentNode.querySelector("td:nth-child(4)").textContent + "\n" +

    "Semester: " + row.parentNode.parentNode.querySelector("td:nth-child(5)").textContent + "\n" +

    "Type: " + row.parentNode.parentNode.querySelector("td:nth-child(6)").textContent + "\n" +

    "Budget: " + row.parentNode.parentNode.querySelector("td:nth-child(7)").textContent + "\n" +

    "Percentage: " + row.parentNode.parentNode.querySelector("td:nth-child(8)").textContent;

  const confirmUpdate = confirm(editPopupTitle + "\n\n" + editPopupContent + "\n\nPress OK to update or Cancel to close.");

  if (confirmUpdate) {

    alert(studentName + " updated successfully");

    checked();

  }

}

function displayAdd() {

  const row = table.insertRow(table.rows.length);

  row.classList.add("dropDownTextArea")

  row.innerHTML =

    '<td colspan="10"> \

      Advisor:<br /><br /> \

      Award Details<br /> \

      Summer 1-2014(TA)<br /> \

      Budget Number: <br /> \

      Tuition Number: <br /> \

      Comments:<br /><br /><br /> \

      Award Status:<br /><br /><br /> \

    </td>';

  configure();

  checked();

}

function configure() {

  const nameElement = document.getElementById("name");

  const nuidElement = document.getElementById("nuid");

  nameElement.textContent = "Full Name: Saheeshna Kamuni";

  nuidElement.textContent = "NUID: 002640603";

  let btnSelect = document.getElementById("button");

  let allChecked = true;

  for (var i = 0; i < checkboxR.length; i++) {

    var row = checkboxR[i].parentNode.parentNode;

    if (!checkboxR[i].checked) {

      allChecked = false;

    }

    if (checkboxR[i].checked) {

      row.querySelectorAll("td")[8].classList.remove("hideColumn");

      row.querySelectorAll("td")[9].classList.remove("hideColumn");

    } else {

      document.querySelectorAll("tr")[0].querySelectorAll("th")[8].classList.add("hideColumn");

      document.querySelectorAll("tr")[0].querySelectorAll("th")[9].classList.add("hideColumn");

      row.querySelectorAll("td")[8].classList.add("hideColumn");

      row.querySelectorAll("td")[9].classList.add("hideColumn");

    }

  }

  if (allChecked) {

    btnSelect.style.backgroundColor = "orange";

    btnSelect.style.cursor = "pointer";

    btnSelect.style.border = "5px solid orange";

    btnSelect.disabled = false;

  } else {

    btnSelect.style.backgroundColor = "gray";

    btnSelect.style.cursor = "initial";

    btnSelect.style.border = "5px solid gray";

    btnSelect.disabled = true;

  }

}

function displayDetails(r) {

  const row = r.parentNode.parentNode;

  const d = row.nextSibling.nextSibling;

  if (boolean) {

    d.style.display = "table-row";

    boolean = false;

  } else {

    d.style.display = "none";

    boolean = true;

  }

}

function updateButtonState() {

  let btnSelect = document.getElementById("button");

  if (sum <= 0) {

    btnSelect.style.backgroundColor = "gray";

    btnSelect.style.cursor = "initial";

    btnSelect.style.border = "5px solid gray";

    btnSelect.style.border = "none";

    btnSelect.disabled = true;

  } else {

    btnSelect.style.backgroundColor = "orange";

    btnSelect.style.cursor = "pointer";

    btnSelect.style.border = "5px solid orange";

    btnSelect.disabled = false;

  }

}