<body>

<h1> </h1>

<div class="student-info" id="student-info">

<div class="category-header">

<h3>INFORMATICIEN-NE</h3>

<button id="toggle-student-info">button>

</div>

<div class="form-container">

**<div class="form-group">**

**<label for="project-name"></label>**

**<input type="text" id="project-name" >**

**</div>**

<div class="form-group full-width">

<label for="remarks"></label>

<textarea id="remarks" class="remark-area" textarea>

</div>

</div>

</div>

<!-- Onglets pour changer de type d'évaluation -->

<div class="tabs">

<div id="tabAuto" class="tab active"></div>

<div id="tabFormative" class="tab inactive" onclick="changeTab('formative')"> div>

<div id="tabsommative" class="tab inactive" onclick="changeTab('sommative')">Évaluation sommative</div>

</div>

<!-- Sections des catégories -->

<div id="categories"><div class="category-container"><div class="category-header">

<h2>PROFESSIONNELLES</h2>

<button onclick="toggleCategoryVisibility('PROFESSIONNELLES')">

Cacher ▲

</button></div><div class="criteria-grid" id="PROFESSIONNELLES"><div class="criterion-card"><div class="criterion-name">Rythme de travail</div>

<div class="slider-container">

<label class="labelEval">Auto:</label>

<input type="range" min="0" max="3" class="slider" id="auto-1" data-id="1" data-level="auto" value="0" oninput="updateSliderValue(this)">

<label class="labelEval">Formative:</label>

<input type="range" min="0" max="3" class="slider" id="formative-1" data-id="1" data-level="formative" value="0" oninput="updateSliderValue(this)" disabled="">

<label class="labelEval">Sommative:</label>

<input type="range" min="0" max="3" class="slider" id="sommative-1" data-id="1" data-level="sommative" value="0" oninput="updateSliderValue(this)" disabled="">

</div>

<div class="appreciation-labels">

<span id="NA-auto-1">NA</span>

<span id="PA-auto-1">PA</span>

<span id="A-auto-1">A</span>

<span id="LA-auto-1">LA</span>

</div>

<div id="result-1" class="result-box bg-gray">NA</div>

<div class="exclude-checkbox">

<label>

<input type="checkbox" id="exclude-1" onclick="toggleExclusion(1)">

Exclure de l'évaluation

</label>

</div></div><div class="criterion-card"><div class="criterion-name">Qualité du travail</div>

<div class="slider-container">

<label class="labelEval">Auto:</label>

<input type="range" min="0" max="3" class="slider" id="auto-2" data-id="2" data-level="auto" value="0" oninput="updateSliderValue(this)">

<label class="labelEval">Formative:</label>

<input type="range" min="0" max="3" class="slider" id="formative-2" data-id="2" data-level="formative" value="0" oninput="updateSliderValue(this)" disabled="">

<label class="labelEval">Sommative:</label>

<input type="range" min="0" max="3" class="slider" id="sommative-2" data-id="2" data-level="sommative" value="0" oninput="updateSliderValue(this)" disabled="">

</div>

<div class="appreciation-labels">

<span id="NA-auto-2">NA</span>

<span id="PA-auto-2">PA</span>

<span id="A-auto-2">A</span>

<span id="LA-auto-2">LA</span>

</div>

<div id="result-2" class="result-box bg-gray">NA</div>

<div class="exclude-checkbox">

<label>

<input type="checkbox" id="exclude-2" onclick="toggleExclusion(2)">

Exclure de l'évaluation

</label>

</div></div><div class="criterion-card"><div class="criterion-name">Niveau de maîtrise technique</div>

<div class="slider-container">

<label class="labelEval">Auto:</label>

<input type="range" min="0" max="3" class="slider" id="auto-3" data-id="3" data-level="auto" value="0" oninput="updateSliderValue(this)">

<label class="labelEval">Formative:</label>

<input type="range" min="0" max="3" class="slider" id="formative-3" data-id="3" data-level="formative" value="0" oninput="updateSliderValue(this)" disabled="">

<label class="labelEval">Sommative:</label>

<input type="range" min="0" max="3" class="slider" id="sommative-3" data-id="3" data-level="sommative" value="0" oninput="updateSliderValue(this)" disabled="">

</div>

<div class="appreciation-labels">

<span id="NA-auto-3">NA</span>

<span id="PA-auto-3">PA</span>

<span id="A-auto-3">A</span>

<span id="LA-auto-3">LA</span>

</div>

<div id="result-3" class="result-box bg-gray">NA</div>

<div class="exclude-checkbox">

<label>

<input type="checkbox" id="exclude-3" onclick="toggleExclusion(3)">

Exclure de l'évaluation

</label>

</div></div><div class="criterion-card"><div class="criterion-name">Autonomie</div>

<div class="slider-container">

<label class="labelEval">Auto:</label>

<input type="range" min="0" max="3" class="slider" id="auto-4" data-id="4" data-level="auto" value="0" oninput="updateSliderValue(this)">

<label class="labelEval">Formative:</label>

<input type="range" min="0" max="3" class="slider" id="formative-4" data-id="4" data-level="formative" value="0" oninput="updateSliderValue(this)" disabled="">

<label class="labelEval">Sommative:</label>

<input type="range" min="0" max="3" class="slider" id="sommative-4" data-id="4" data-level="sommative" value="0" oninput="updateSliderValue(this)" disabled="">

</div>

<div class="appreciation-labels">

<span id="NA-auto-4">NA</span>

<span id="PA-auto-4">PA</span>

<span id="A-auto-4">A</span>

<span id="LA-auto-4">LA</span>

</div>

<div id="result-4" class="result-box bg-gray">NA</div>

<div class="exclude-checkbox">

<label>

<input type="checkbox" id="exclude-4" onclick="toggleExclusion(4)">

Exclure de l'évaluation

</label>

</div></div></div></div><div class="category-container"><div class="category-header">

<h2>METHODOLOGIQUES</h2>

<button onclick="toggleCategoryVisibility('METHODOLOGIQUES')">

Cacher ▲

</button></div><div class="criteria-grid" id="METHODOLOGIQUES"><div class="criterion-card"><div class="criterion-name">Processus de travail</div>

<div class="slider-container">

<label class="labelEval">Auto:</label>

<input type="range" min="0" max="3" class="slider" id="auto-5" data-id="5" data-level="auto" value="0" oninput="updateSliderValue(this)">

<label class="labelEval">Formative:</label>

<input type="range" min="0" max="3" class="slider" id="formative-5" data-id="5" data-level="formative" value="0" oninput="updateSliderValue(this)" disabled="">

<label class="labelEval">Sommative:</label>

<input type="range" min="0" max="3" class="slider" id="sommative-5" data-id="5" data-level="sommative" value="0" oninput="updateSliderValue(this)" disabled="">

</div>

<div class="appreciation-labels">

<span id="NA-auto-5">NA</span>

<span id="PA-auto-5">PA</span>

<span id="A-auto-5">A</span>

<span id="LA-auto-5">LA</span>

</div>

<div id="result-5" class="result-box bg-gray">NA</div>

<div class="exclude-checkbox">

<label>

<input type="checkbox" id="exclude-5" onclick="toggleExclusion(5)">

Exclure de l'évaluation

</label>

</div></div><div class="criterion-card"><div class="criterion-name">Expression orale et écrite</div>

<div class="slider-container">

<label class="labelEval">Auto:</label>

<input type="range" min="0" max="3" class="slider" id="auto-6" data-id="6" data-level="auto" value="0" oninput="updateSliderValue(this)">

<label class="labelEval">Formative:</label>

<input type="range" min="0" max="3" class="slider" id="formative-6" data-id="6" data-level="formative" value="0" oninput="updateSliderValue(this)" disabled="">

<label class="labelEval">Sommative:</label>

<input type="range" min="0" max="3" class="slider" id="sommative-6" data-id="6" data-level="sommative" value="0" oninput="updateSliderValue(this)" disabled="">

</div>

<div class="appreciation-labels">

<span id="NA-auto-6">NA</span>

<span id="PA-auto-6">PA</span>

<span id="A-auto-6">A</span>

<span id="LA-auto-6">LA</span>

</div>

<div id="result-6" class="result-box bg-gray">NA</div>

<div class="exclude-checkbox">

<label>

<input type="checkbox" id="exclude-6" onclick="toggleExclusion(6)">

Exclure de l'évaluation

</label>

</div></div><div class="criterion-card"><div class="criterion-name">Durabilité et innovation</div>

<div class="slider-container">

<label class="labelEval">Auto:</label>

<input type="range" min="0" max="3" class="slider" id="auto-7" data-id="7" data-level="auto" value="0" oninput="updateSliderValue(this)">

<label class="labelEval">Formative:</label>

<input type="range" min="0" max="3" class="slider" id="formative-7" data-id="7" data-level="formative" value="0" oninput="updateSliderValue(this)" disabled="">

<label class="labelEval">Sommative:</label>

<input type="range" min="0" max="3" class="slider" id="sommative-7" data-id="7" data-level="sommative" value="0" oninput="updateSliderValue(this)" disabled="">

</div>

<div class="appreciation-labels">

<span id="NA-auto-7">NA</span>

<span id="PA-auto-7">PA</span>

<span id="A-auto-7">A</span>

<span id="LA-auto-7">LA</span>

</div>

<div id="result-7" class="result-box bg-gray">NA</div>

<div class="exclude-checkbox">

<label>

<input type="checkbox" id="exclude-7" onclick="toggleExclusion(7)">

Exclure de l'évaluation

</label>

</div></div></div></div><div class="category-container"><div class="category-header">

<h2>SOCIALES</h2>

<button onclick="toggleCategoryVisibility('SOCIALES')">

Cacher ▲

</button></div><div class="criteria-grid" id="SOCIALES"><div class="criterion-card"><div class="criterion-name">Aptitude au travail en équipe</div>

<div class="slider-container">

<label class="labelEval">Auto:</label>

<input type="range" min="0" max="3" class="slider" id="auto-8" data-id="8" data-level="auto" value="0" oninput="updateSliderValue(this)">

<label class="labelEval">Formative:</label>

<input type="range" min="0" max="3" class="slider" id="formative-8" data-id="8" data-level="formative" value="0" oninput="updateSliderValue(this)" disabled="">

<label class="labelEval">Sommative:</label>

<input type="range" min="0" max="3" class="slider" id="sommative-8" data-id="8" data-level="sommative" value="0" oninput="updateSliderValue(this)" disabled="">

</div>

<div class="appreciation-labels">

<span id="NA-auto-8">NA</span>

<span id="PA-auto-8">PA</span>

<span id="A-auto-8">A</span>

<span id="LA-auto-8">LA</span>

</div>

<div id="result-8" class="result-box bg-gray">NA</div>

<div class="exclude-checkbox">

<label>

<input type="checkbox" id="exclude-8" onclick="toggleExclusion(8)">

Exclure de l'évaluation

</label>

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

<!-- Conteneur pour afficher les résultats des trois onglets -->

<div id="result-container">

<div class="result-box" id="auto-result-box">

<h3>Self Eval</h3>

<p id="auto-average">NA</p>

<span class="info-evaluation">

L’auto-évaluation permet à l’élève de réfléchir sur ses propres performances et progrès.

Elle encourage la prise de conscience et l’autonomie dans le processus d’apprentissage.

</span>

</div>

<div class="result-box" id="formative-result-box">

<h3>Formative</h3>

<p id="formative-average">NA</p>

<span class="info-evaluation">

L’évaluation formative est un outil continu qui vise à guider l’apprentissage des élèves.

Elle permet d’identifier les forces et les domaines à améliorer pour progresser.

</span>

</div>

<div class="result-box key" id="sommative-result-box">

<h3>Sommative</h3>

<p id="sommative-average">NA</p>

<span class="info-evaluation">

L’évaluation sommative se déroule à la fin d’une période d’apprentissage (comme un semestre ou une année

scolaire).

Elle vise à évaluer les acquis des élèves par rapport à un référentiel ou à des normes.

</span>

</div>

</div>

<!-- JAVASCRIPT -->

<script>

let isStudentInfoVisible = true;

const appreciationLabels = ["NA", "PA", "A", "LA"];

const visibleCategories = {

PROFESSIONNELLES: true,

METHODOLOGIQUES: true,

SOCIALES: true

};

const criteria = [

{ id: 1, name: 'Rythme de travail', category: 'PROFESSIONNELLES' },

{ id: 2, name: 'Qualité du travail', category: 'PROFESSIONNELLES' },

{ id: 3, name: 'Niveau de maîtrise technique', category: 'PROFESSIONNELLES' },

{ id: 4, name: 'Autonomie', category: 'PROFESSIONNELLES' },

{ id: 5, name: 'Processus de travail', category: 'METHODOLOGIQUES' },

{ id: 6, name: 'Expression orale et écrite', category: 'METHODOLOGIQUES' },

{ id: 7, name: 'Durabilité et innovation', category: 'METHODOLOGIQUES' },

{ id: 8, name: 'Aptitude au travail en équipe', category: 'SOCIALES' }

];

let sliderValues = {

auto: {},

formative: {},

sommative: {}

};

let currentTab = 'auto'; // Par défaut, l'onglet actif est l'auto-évaluation

// Restauration des valeurs des sliders au rendu des catégories

function renderCategories() {

const categoriesDiv = document.getElementById('categories');

categoriesDiv.innerHTML = ''; // Réinitialiser l'affichage

const groupedCriteria = criteria.reduce((acc, crit) => {

const category = crit.category.trim().toUpperCase();

acc[category] = acc[category] || [];

acc[category].push(crit);

return acc;

}, {});

Object.keys(groupedCriteria).forEach(category => {

const container = document.createElement('div');

container.className = 'category-container';

const header = document.createElement('div');

header.className = 'category-header';

header.innerHTML = `

<h2>${category}</h2>

<button onclick="toggleCategoryVisibility('${category}')">

${visibleCategories[category] ? 'Cacher ▲' : 'Afficher ▼'}

</button>`;

container.appendChild(header);

const criteriaDiv = document.createElement('div');

criteriaDiv.className = 'criteria-grid';

criteriaDiv.id = category;

groupedCriteria[category].forEach(crit => {

const card = document.createElement('div');

card.className = 'criterion-card';

card.innerHTML = `<div class="criterion-name">${crit.name}</div>

<div class="slider-container">

${['auto', 'formative', 'sommative'].map(level => `

<label class='labelEval'>${level.charAt(0).toUpperCase() + level.slice(1)}:</label>

<input type="range" min="0" max="3" class="slider" id="${level}-${crit.id}"

data-id="${crit.id}" data-level="${level}"

value="${sliderValues[level]?.[crit.id] || 0}"

oninput="updateSliderValue(this)"

${currentTab !== level ? 'disabled' : ''}>

`).join('')}

</div>

<div class="appreciation-labels">

${appreciationLabels.map((label, i) => `

<span id="${label}-${currentTab}-${crit.id}">${label}</span>

`).join('')}

</div>

<div id="result-${crit.id}" class="result-box bg-gray">

${appreciationLabels[sliderValues[currentTab]?.[crit.id] || 0]}

</div>

<div class="exclude-checkbox">

<label>

<input type="checkbox" id="exclude-${crit.id}" onclick="toggleExclusion(${crit.id})"

${excludedCriteria[crit.id] ? 'checked' : ''}>

Exclure de l'évaluation

</label>

</div>`;

criteriaDiv.appendChild(card);

});

container.appendChild(criteriaDiv);

categoriesDiv.appendChild(container);

});

// Initialiser les valeurs des sliders après avoir rendu les catégories

initializeSliderValues();

}

function toggleStudentInfo() {

const studentInfo = document.getElementById('student-info');

const toggleButton = document.getElementById('toggle-student-info');

const secondDiv = studentInfo.querySelector('.form-container');

isStudentInfoVisible = !isStudentInfoVisible;

if (isStudentInfoVisible) {

toggleButton.innerText = 'Cacher ▲';

secondDiv.style.display = '';

studentInfo.style.maxHeight = '';

} else {

toggleButton.innerText = 'Afficher ▼';

secondDiv.style.display = 'none';

studentInfo.style.maxHeight = '40px';

}

}

function toggleCategoryVisibility(category) {

const categoryElement = document.getElementById(category);

if (!categoryElement) {

console.error(`L'élément avec l'ID "${category}" n'a pas été trouvé.`);

return;

}

const container = categoryElement.parentElement;

// Basculer la visibilité

visibleCategories[category] = !visibleCategories[category];

if (visibleCategories[category]) {

container.style.maxHeight = '1000px';

categoryElement.style.visibility = 'visible';

} else {

container.style.maxHeight = '15px';

categoryElement.style.visibility = 'hidden';

}

}

// Recalcul automatique lors de l'interaction avec un slider ou une exclusion

function updateSliderValue(slider) {

const id = slider.dataset.id;

const level = slider.dataset.level;

const value = slider.value;

sliderValues[level][id] = value; // Mémoriser la valeur

document.getElementById(`result-${id}`).textContent = appreciationLabels[value]; // Mettre à jour l'appréciation

calculateFinalResults(); // Recalculer les résultats

}

function getDateDefault() {

// Obtenir la date actuelle

const today = new Date();

// Formater la date au format YYYY-MM-DD

const yyyy = today.getFullYear();

let mm = today.getMonth() + 1; // Les mois commencent à 0

let dd = today.getDate();

// Ajouter un zéro devant les mois ou jours inférieurs à 10

if (mm < 10) mm = '0' + mm;

if (dd < 10) dd = '0' + dd;

// Construire la date au format nécessaire (sans espaces)

const todayFormatted = `${yyyy}-${mm}-${dd}`;

// Affecter la date au champ input

document.getElementById('evaluation-date').value = todayFormatted;

}

function changeTab(tab) {

currentTab = tab;

// Retirer la classe 'active' de tous les onglets

document.getElementById('tabAuto').classList.remove('active');

document.getElementById('tabFormative').classList.remove('active');

document.getElementById('tabsommative').classList.remove('active');

document.getElementById('tabAuto').classList.remove('inactive');

document.getElementById('tabFormative').classList.remove('inactive');

document.getElementById('tabsommative').classList.remove('inactive');

// Ajouter la classe 'active' à l'onglet sélectionné

if (tab === 'auto') {

document.getElementById('tabAuto').classList.add('active');

} else if (tab === 'formative') {

document.getElementById('tabFormative').classList.add('active');

} else {

document.getElementById('tabsommative').classList.add('active');

}

// Mettre à jour les catégories en fonction de l'onglet sélectionné

renderCategories();

}

let excludedCriteria = {}; // Un objet pour suivre les critères exclus

// Gérer les exclusions

function toggleExclusion(criterionId) {

excludedCriteria[criterionId] = document.getElementById(`exclude-${criterionId}`).checked;

calculateFinalResults(); // Recalculer après modification

}

function calculateFinalResults() {

let totalScores = { auto: 0, formative: 0, sommative: 0 };

let count = 8;

let naCount = { auto: 0, formative: 0, sommative: 0 };

let paCount = { auto: 0, formative: 0, sommative: 0 };

let aCount = { auto: 0, formative: 0, sommative: 0 };

let laCount = { auto: 0, formative: 0, sommative: 0 };

criteria.forEach(crit => {

if (excludedCriteria[crit.id]) {

count--;

return; // Ne pas inclure si exclu

}

['auto', 'formative', 'sommative'].forEach(level => {

const value = sliderValues[level][crit.id];

if (value !== undefined) {

totalScores[level] += parseInt(value);

if (value < 1) naCount[level]++;

else if (value < 2) paCount[level]++;

else if (value < 3) aCount[level]++;

else laCount[level]++;

}

});

});

console.log(count, naCount['auto'], paCount['auto'], aCount['auto'], laCount['auto']);

['auto', 'formative', 'sommative'].forEach(level => {

// Ne fonction pas

/\*

Si au moins un critère est évalué NA -> NA

Si au moins un critère est évalué PA et aucun NA -> PA

Si plus de 50% des critères sont évalués A et aucun critère n’est évalué NA ou PA -> A

Si au moins 50% des critères sont évalués LA et aucun critère n’est évalué NA ou PA -> LA

\*/

let result;

if (naCount[level] > 0) { result = "NA"; console.log('NA'); }

else if (laCount[level] >= Math.floor(count[level] / 2)) { result = "LA"; console.log('LA'); }

else if (aCount[level] > Math.floor(count[level] / 2)) { result = "A"; console.log('A'); }

else if (paCount[level] > 0) { result = "PA"; console.log('PA'); }

else { result = "PA"; console.log('Default - PA'); }

document.getElementById(`${level}-average`).textContent = result;

});

}

function initializeSliderValues() {

// Initialiser les valeurs des sliders pour chaque critère et chaque onglet

criteria.forEach(crit => {

['auto', 'formative', 'sommative'].forEach(level => {

if (!sliderValues[level][crit.id]) {

sliderValues[level][crit.id] = 0; // Valeur par défaut

}

// Mettre à jour l'interface utilisateur

const slider = document.querySelector(`input[data-id= "${crit.id}"][data-level="${level}"]`);

if (slider) {

slider.value = sliderValues[level][crit.id];

const resultDiv = document.getElementById(`result-${crit.id}`);

resultDiv.innerText = appreciationLabels[slider.value];

}

});

});

// Recalculer les résultats une fois les valeurs initiales définies

calculateFinalResults();

}

renderCategories();

getDateDefault();

</script>

</body>