

TP : Développement en HTML, CSS et JS

Exercice 1: Calcul de la Moyenne des Notes

Un professeur souhaite calculer la moyenne des notes de ses élèves. Écrivez un programme en **JavaScript** qui :

1. Stocke les notes dans un **tableau**.
2. Utilise une **boucle** pour calculer la moyenne.
3. Affiche la moyenne dans la console et dans une page HTML.

Exercice 2: Trouver le Nombre Maximum dans un Tableau

Écrivez un programme en **JavaScript** qui :

1. Stocke une liste de nombres dans un **tableau**.
2. Utilise une **boucle** pour trouver le **nombre le plus grand**.
3. Affiche ce nombre dans la **console** et sur la **page HTML**.

Exercice 3: Trouver le Nombre Maximum dans un Tableau avec saisie

Écrivez un programme en **JavaScript** qui permet à l'utilisateur de **saisir une liste de nombres** séparés par des virgules. Le programme doit ensuite :

1. **Extraire et convertir** ces valeurs en un tableau de nombres.
2. **Déterminer le nombre maximum** parmi ces valeurs.
3. **Afficher les nombres sous forme de tableau HTML**, en indiquant clairement la valeur maximale en rouge.

L'affichage doit être soigné avec un **tableau structuré et stylisé en CSS**.

Exercice 4: Création et manipulation de classes

Créez une classe `Student` qui représente un étudiant avec un **nom**, un **âge**, et une **liste de notes**.

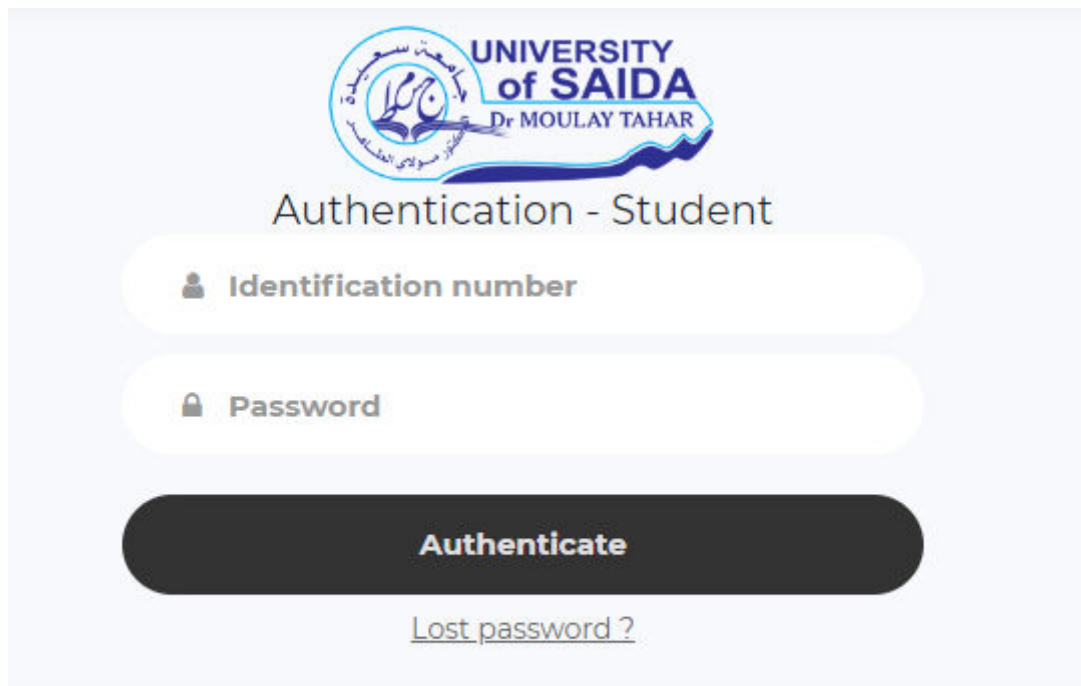
Implémentez les méthodes suivantes :

- **addNote (note)** : Ajouter une note à l'étudiant.
- **calculateAverage ()** : Calculer la moyenne des notes.
- **displayInfo ()** : Afficher les informations de l'étudiant dans un **container HTML** sous forme de tableau.

L'utilisateur pourra saisir les informations d'un étudiant (nom, âge, notes) via un formulaire et voir les résultats affichés dynamiquement.

Exercice 5: universco

Ecrire le code HTML, CSS et JS qui crée ces 2 pages.



Correction

Exercice 1: Calcul de la Moyenne des Notes

```
<!DOCTYPE html>
<html lang="fr">
<head>
  <meta charset="UTF-8">
  <meta name="viewport" content="width=device-width, initial-scale=1.0">
  <title>Calcul de la Moyenne</title>
</head>
<body>

  <h2>Calcul de la Moyenne des Notes</h2>
  <p id="result"></p>

  <script>
    // Déclaration du tableau des notes
    let notes = [15, 12, 18, 10, 14, 16, 19];

    // Initialisation de la somme des notes
    let somme = 0;

    // Utilisation d'une boucle pour calculer la somme
    for (let i = 0; i < notes.length; i++) {
      somme += notes[i];
    }

    // Calcul de la moyenne
    let moyenne = somme / notes.length;
```

```

// Affichage du résultat

console.log("La moyenne est : " + moyenne);

document.getElementById("result").innerHTML = "La moyenne des notes est : " +
moyenne.toFixed(2);

</script>

</body>

</html>

```

Exercice 2: Trouver le Nombre Maximum dans un Tableau

```

<!DOCTYPE html>

<html lang="fr">

<head>

  <meta charset="UTF-8">

  <meta name="viewport" content="width=device-width, initial-scale=1.0">

  <title>Nombre Maximum</title>

</head>

<body>

  <h2>Nombre Maximum dans un Tableau</h2>

  <p id="result"></p>

  <script>

    // Déclaration du tableau de nombres

    let nombres = [45, 78, 12, 34, 99, 56, 89, 102, 23];

    // Supposons que le premier élément est le plus grand

    let max = nombres[0];

    // Utilisation d'une boucle pour trouver le maximum

    for (let i = 1; i < nombres.length; i++) {

```

```

        if (nombres[i] > max) {
            max = nombres[i];
        }
    }

    // Affichage du résultat
    console.log("Le nombre maximum est : " + max);
    document.getElementById("result").innerHTML = "Le plus grand nombre est : " + max;
</script>

</body>
</html>

```

Exercice 3: Trouver le Nombre Maximum dans un Tableau avec saisie

```

<!DOCTYPE html>
<html lang="fr">
<head>
    <meta charset="UTF-8">
    <meta name="viewport" content="width=device-width, initial-scale=1.0">
    <title>Nombre Maximum</title>
    <style>
        body {
            font-family: Arial, sans-serif;
            background-color: #f4f4f4;
            text-align: center;
            padding: 20px;
        }
        .container {
            background: white;
            padding: 20px;

```

```

    border-radius: 10px;
    box-shadow: 0px 0px 10px rgba(0, 0, 0, 0.2);
    display: inline-block;
}
input, button {
    padding: 10px;
    margin: 10px;
    font-size: 16px;
}
table {
    width: 100%;
    border-collapse: collapse;
    margin-top: 20px;
}
th, td {
    border: 1px solid #ddd;
    padding: 10px;
    text-align: center;
}
th {
    background-color: #4CAF50;
    color: white;
}
.highlight {
    font-weight: bold;
    color: red;
}
</style>
</head>
<body>

```

```

<div class="container">

  <h2>Nombre Maximum dans un Tableau</h2>

  <p>Entrez une série de nombres séparés par des virgules :</p>

  <input type="text" id="inputNumbers" placeholder="Ex: 12, 45, 78, 23">

  <button onclick="trouverMaximum()">Trouver le Maximum</button>


  <h3>Résultat</h3>

  <table id="resultTable">

    <thead>

      <tr>

        <th>Index</th>

        <th>Valeur</th>

      </tr>

    </thead>

    <tbody>

    </tbody>

  </table>


  <p id="maxResult"></p>

</div>


<script>

function trouverMaximum() {

  let input = document.getElementById("inputNumbers").value;

  let numbers = input.split(",").map(num => parseFloat(num.trim()));

  if (numbers.some(isNaN)) {

    alert("Veuillez entrer uniquement des nombres séparés par des virgules.");

    return;

  }

}

```

```

let max = Math.max(...numbers);

let tableBody = document.querySelector("#resultTable tbody");
tableBody.innerHTML = ""; // Vider le tableau avant de le remplir

numbers.forEach((num, index) => {
    let row = tableBody.insertRow();
    let cellIndex = row.insertCell(0);
    let cellValue = row.insertCell(1);

    cellIndex.innerText = index + 1;
    cellValue.innerText = num;

    // Mettre en évidence la valeur maximale
    if (num === max) {
        cellValue.classList.add("highlight");
    }
});

document.getElementById("maxResult").innerHTML = `<strong>Le nombre maximum est :
<span class="highlight">${max}</span></strong>`;
}
</script>

</body>
</html>

```

Exercice 4: Création et manipulation de classes

Exo4.html

```
<!DOCTYPE html>
```



```

<html lang="fr">
<head>
  <meta charset="UTF-8">
  <meta name="viewport" content="width=device-width, initial-scale=1.0">
  <title>Gestion des Étudiants</title>
  <link rel="stylesheet" href="exo4.css">
</head>
<body>

  <div class="container">
    <h2>Ajouter un Étudiant</h2>

    <form id="studentForm">
      <label for="name">Nom :</label>
      <input type="text" id="name" required>

      <label for="age">Âge :</label>
      <input type="number" id="age" required>

      <label for="notes">Notes (séparées par des virgules) :</label>
      <input type="text" id="notes" required>

      <button type="submit">Ajouter</button>
    </form>

    <h2>Liste des Étudiants</h2>
    <table>
      <thead>
        <tr>
          <th>Nom</th>
          <th>Âge</th>

```

```

        <th>Notes</th>

        <th>Moyenne</th>

    </tr>

</thead>

    <tbody id="studentList"></tbody>

</table>

</div>

<script src="exo4.js"></script>

</body>

</html>

```

Exo4.css

```

body {

    font-family: Arial, sans-serif;

    background-color: #f5f5f5;

    text-align: center;

    margin: 0;

    padding: 20px;

}

.container {

    width: 50%;

    margin: auto;

    background-color: white;

    padding: 20px;

    box-shadow: 0px 4px 10px rgba(0, 0, 0, 0.1);

    border-radius: 10px;

}

```

```
h2 {  
  color: #333;  
}  
  
form {  
  display: flex;  
  flex-direction: column;  
  gap: 10px;  
}  
  
input {  
  padding: 8px;  
  border: 1px solid #ccc;  
  border-radius: 5px;  
}  
  
button {  
  padding: 10px;  
  border: none;  
  background-color: #28a745;  
  color: white;  
  font-size: 16px;  
  cursor: pointer;  
  border-radius: 5px;  
}  
  
button:hover {  
  background-color: #218838;  
}  
  
table {
```

```
width: 100%;  
margin-top: 20px;  
border-collapse: collapse;  
}
```

```
th, td {  
    border: 1px solid #ccc;  
    padding: 10px;  
}
```

```
th {  
    background-color: #28a745;  
    color: white;  
}
```

Exo4.js

```
// Définition de la classe Student  
class Student {  
    constructor(name, age) {  
        this.name = name;  
        this.age = age;  
        this.notes = [];  
    }  
  
    // Ajouter une note  
    addNote(note) {  
        this.notes.push(parseFloat(note));  
    }  
  
    // Calculer la moyenne
```

```

calculateAverage() {
  if (this.notes.length === 0) return 0;

  let sum = this.notes.reduce((a, b) => a + b, 0);

  return (sum / this.notes.length).toFixed(2);
}

// Récupérer les informations sous forme de tableau HTML
getRowHTML() {
  return `
    <tr>
      <td>${this.name}</td>
      <td>${this.age}</td>
      <td>${this.notes.join(", ")}</td>
      <td>${this.calculateAverage()}</td>
    </tr>
  `;
}

}

// Sélection des éléments du DOM
const studentForm = document.getElementById("studentForm");
const studentList = document.getElementById("studentList");

// Tableau pour stocker les étudiants
let students = [];

// Gestion de la soumission du formulaire
studentForm.addEventListener("submit", function(event) {
  event.preventDefault();

  // Récupération des valeurs

```

```

let name = document.getElementById("name").value.trim();
let age = document.getElementById("age").value.trim();
let notes = document.getElementById("notes").value.trim().split(",");

// Création d'un nouvel étudiant
let student = new Student(name, age);

// Ajout des notes
notes.forEach(note => {
    if (!isNaN(note) && note !== "") {
        student.addNote(note);
    }
});

// Ajout de l'étudiant au tableau
students.push(student);

// Mise à jour de l'affichage
updateStudentTable();

// Réinitialisation du formulaire
studentForm.reset();
});

// Fonction pour mettre à jour le tableau des étudiants
function updateStudentTable() {
    studentList.innerHTML = ""; // Effacer l'ancien contenu

    students.forEach(student => {
        studentList.innerHTML += student.getRowHTML();
    });
}

```

}