

TP1

Exercice 1

- 1- Créer un dossier nommé **devweb**
- 2- Ouvrir le dossier **devweb** dans l'éditeur **ATOM** (*File->Open folder*)
- 3- Créer, dans ce dossier, un fichier **tp1.html**

tp1.html	Questions - Qu'est-ce qu'une balise (tag) html ? - Citer la liste des balises utilisées dans tp1.html - Qu'est-ce qu'un élément HTML ? - Citer la liste des éléments de la page tp1.html - Combien d'éléments, des types suivants, peut une page HTML avoir ? html, head, title, body, et p - Quel est le rôle de l'élément body ?
4- Ajouter à la page tp1.html les éléments suivants : - <code><h1> Heading level 1</h1></code> - <code><h2> Heading level 2</h2></code> - <code><h3> Heading level 3</h3></code> - <code><h4> Heading level 4</h4></code>	Questions - Combien de niveaux de <i>heading</i> existe-il dans l'HTML ?

Exercice 2

- Ajouter à la page **tp1.html** (de l'exercice 1) la table suivante
- Expliquer les différents éléments **table**, **tr**, **th**, et **td**
- Ajouter les attribues **border = "1"** et **cellspacing="0"** à la balise **<table>**, que remarquer vous ?
- C'est quoi un **attribut** ?
- Ajouter à cette table, la ligne suivante

Ahmed	02/02/2002
-------	------------

```

<table>
  <tr>
    <th>Nom</th>
    <th>Date de naissance</th>
  </tr>
  <tr>
    <td>Mohamed</td>
    <td>01/01/2001</td>
  </tr>
</table>

```

Exercice 3 (Ajouter la liste imbriquée suivante à la page **tp1.html**)

```

<ol>
  <li>Mercedes Benz
    <ul>
      <li>E350</li>
      <li>C220</li>
    </ul>
  </li>
</ol>

```

Questions

- C'est quoi une liste imbriquée ?
- Définir les éléments **ol**, **li**, et **ul**
- Modifier la liste précédente pour obtenir la forme de la **Figure 1**
- Quelle est la relation entre **ol**, **li**, et **ul**

1. Mercedes Benz
 - E350
 - C220
 2. Audi
 - A4
 - Q5
 3. Volkswagen
 - Golf
 - Passat

Figure 1

TP2

Exercice 1

- 1- En utilisant les éléments `<table>`, `<th>`, `<tr>` et `<td>`, créer la table montrée dans la figure suivante.

Column 1 heading	Column 2 heading	Column 3 heading
Row 1, cell 1	Two cells خليتان من سطر واحد	
Two cells خليتان من عمود واحد	Row 2, cell 2	Row 2, cell 3
	Row 3, cell 2	Row 3, cell 3

- 2- Ajouter le style suivant

```
<style> td , th { border: 1px solid #999; padding: 0.1em 1em; text-align: center;} </style>
```

Exercice 2 (continuer dans la même page précédente)

1. Ajouter le formulaire suivant

```
<form method="get" action="tp2.htm">
  Nom: <input name="nom">
  Prenom: <input name="prenom">
  Mois de naissance:
  <select name="month">
    <option value="1">Janvier</option>
    <option value="2">Février</option>
    ...
    <option value="12">Décembre</option>
  </select>
  <button type="submit">Envoyer</button>
</form>
```

2. Insérer des sauts de ligne dans les emplacements appropriés.

3. Organiser le contenu de `<form>` en utilisant les balise `<table>`, `<tr>`, `<td>` pour qu'il soit similaire au formulaire montré dans la figure suivante.

Nom:	
Prenom:	
Mois de naissance:	Janvier v
Envoyer	

4. Changer l'attribut méthode à `method="post"`, que remarquer vous ?

Exercice 3

- 1- Créer, dans le même dossier, les deux pages `p1.html` et `p2.html` suivantes.

```
<html>
<head>
  <title>Page 1</title>
</head>
<body>
  <p>you are in <strong>P1</strong></p>
  <a href="p2.html">go to p2</a>
</body>
</html>
```

```
<html>
<head>
  <title>Page 2</title>
</head>
<body>
  <p>you are in <strong>P2</strong></p>
</body>
</html>
```

- 2- Ajouter un lien à la page « `p2.html` » permettant de se retourner à la page « `p1.html` »

TP3

1. Ecrire dans le *body* les éléments suivants

```
<body>
  <div class="container">
    <div class="top-bar">
      <ul class="top-menu">
        <li><a href="#">Home</a></li>
        <li><a href="#">About</a></li>
        <li><a href="#">Services</a></li>
        <li><a href="#">Contact</a></li>
      </ul>
    </div>
    <div class="content">
      <div class="sidebar">
        <a href="#">Link 1</a>
        <a href="#">Link 2</a>
        <a href="#">Link 3</a>
        <a href="#">Link 4</a>
      </div>
      <div class="main-content">
        Main Content Area - Page Content Goes Here
      </div>
    </div>
    <div class="bottom-bar">
      Bottom Bar - Footer, Copyright Information, etc.
    </div>
  </div>
</body>
```

2. Définir les styles CSS suivants (dans un éléments style puis dans un fichier séparé **style.css**)

body {	.sidebar {
margin: 0;	background-color: #f4f4f4;
padding: 0;	width: 200px;
font-family: Arial, sans-serif;	padding: 10px;
}	}
.container {	.sidebar a {
display: flex;	display: block;
flex-direction: column;	margin-bottom: 10px;
height: 100vh;	color: #333;
}	text-decoration: none;
	}
	.main-content {

```
.top-bar {
    background-color: #333;
    color: white;
    padding: 10px;
    text-align: center;
}
.top-menu {
    list-style-type: none;
    padding: 0;
    margin: 0;
    float: right;
}
.top-menu li {
    display: inline;
    margin-right: 20px;
}
.top-menu li a {
    color: white;
    text-decoration: none;
}
.content {
    display: flex;
    flex: 1;
}
flex: 1;
padding: 10px;
}
.bottom-bar {
    background-color: #333;
    color: white;
    padding: 10px;
    text-align: center;
}
```

TP4**Exercice 1**

- 1- Taper le code HTML suivant dans un fichier html **tp4.html** puis tester la page sur le navigateur

```
<html>
<body>
<h2>My First JavaScript</h2>
<button onclick="document.getElementById('demo').innerHTML = 'hello world!' ">
Click me</button>
<p id="demo"></p>
</body>
</html>
```

- 2- Séparer le code JS dans une fonction **afficher()** dans l'élément **<script>**

- 3- Ajouter les fonctions suivantes

Elément	Événement	Fonction	Description
demo	onmouseover	changer1()	Changer le background-color au red
demo	onmouseout	changer2()	Changer le background-color au transparent

Exercice 2

- 1- Ajouter le code html suivant dans la page **tp4.html**

```
<input type="text" id="input_x">
<input type="text" id="input_y">
<button onclick="addxy()">Ajouter</button>
<p id="output"></p>
```

- 2- Ajouter la fonction JS **addxy()** suivante

```
function addxy() {
    var x = document.getElementById('input_x').value;
    var y = document.getElementById('input_y').value;
    var z = x + y;
    document.getElementById('output').innerHTML = z;
}
```

- 3- Tester le code dans le navigateur. Que remarque-tu ?

- 4- Modifier la fonction de telle sorte qu'elle retourne la somme de deux nombres entiers.

Exercice 3

- 1- Créer un nouveau fichier HTML **test.html**

```
<html>
<body>
<p id="output"></p>
<script type="text/javascript">
var s=0;
for(var i=0; i<10; i++){
    s+=i;
}
document.getElementById('output').innerHTML = s;
</script>
</body>
</html>
```

- 2- Modifier la page précédente en ajoutant un **bouton** et un **input**. Lorsqu'on clique sur le bouton, on lit une valeur entière **n** à partir de l'**input** et on affiche la somme **s = 1 + 2 + + n** dans le paragraphe **output**

Exercice 4

1- Ajouter le formulaire suivant ainsi que le paragraphe ci-dessous à votre page

```
<form>
  <input type="text" id="textInput" value="Hello World"><br>
  <input type="number" id="numberInput" name="numberInput" value="42"><br>
  <input type="checkbox" id="checkboxInput" name="checkboxInput" checked><br>
  <select id="selectInput" name="selectInput">
    <option value="1">Option 1</option>
    <option value="2">Option 2</option>
    <option value="3">Option 3</option>
  </select><br>
  <textarea id="textareaInput" name="textareaInput">Lorem ipsum dolor sit amet.</textarea><br>
  <button type="button" onclick="readFormValues()">Read Values</button>
</form>
```

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

2- Complétez la fonction **readFormValues()** qui lit les valeurs des éléments de saisie du formulaire précédent, puis les affiche concaténées dans le paragraphe 'result'.