

# TECHNOLOGIE WEB

## HTML5 CSS3 JAVASCRIPT

Pr. Faouzia Benabbou (faouzia.benabbou@univh2c.ma)  
Département de mathématiques et Informatique  
Laboratoire TIM

**FSBM : 2020-2021**



# LANGAGE CASCADING STYLE SHEETS CSS

2

# MISE EN PAGE AVEC FLEXBOX

- Le concept de Flex Box en CSS permet la disposition des éléments dans une page web plus facilement.
- Ce standard permet d'avoir un design adaptatif à l'écran sans se soucier de positionnement ou de marges.
- Les éléments peuvent être réagencés selon la taille de l'écran.

- Exemple

```
<div class="container">  
  <div>1</div>  
  <div>2</div>  
  <div>3</div>  
</div>
```

```
<style>  
.container {  
  display: flex;  
  background-color: SkyBlue;  
}
```

```
.container > div {  
  background-color: pink;  
  margin: 20px;  
  padding: 20px;  
  font-size: 2em;  
}  
</style>
```

# PROPRIÉTÉS DE FLEXBOX

- Le conteneur flexbox a les propriétés suivantes:
  - flex-direction
  - flex-wrap
  - flex-flow
  - justify-content
  - align-items
  - align-content
- (<https://css-tricks.com/snippets/css/a-guide-to-flexbox/>)

# PROPRIÉTÉ FLEX-DIRECTION

- La propriété "flex-direction" définit dans quelle direction le conteneur veut empiler les articles flexflex-wrap,
- Elle peut prendre les valeurs suivantes :
  - Column
  - Row
  - Column-reverse
  - Row-reverse

# EXEMPLE

`flex-direction:row;`



`flex-direction:row-reverse;`



# EXEMPLE

`flex-direction:Column;`

bloc 1

bloc 2

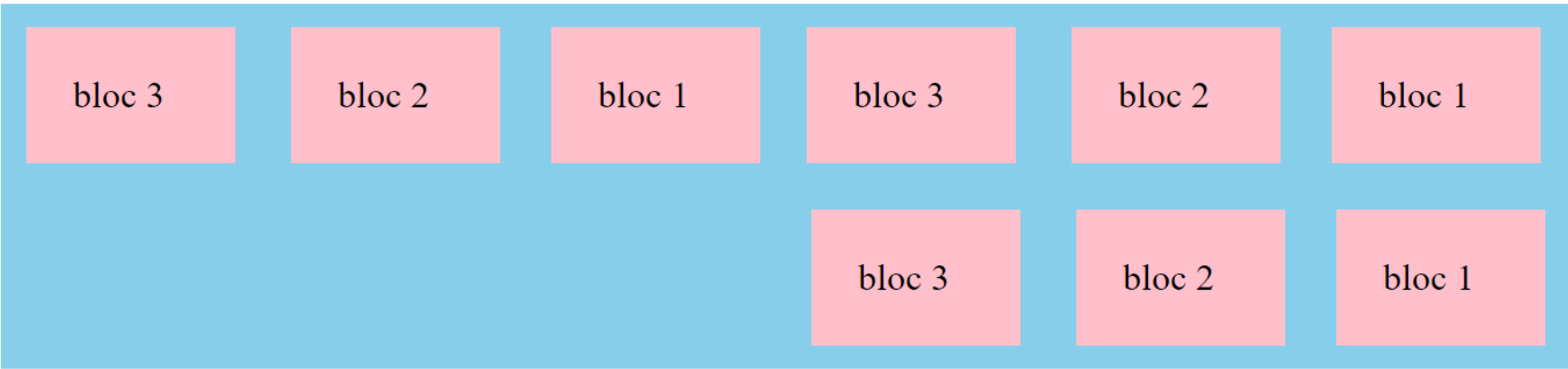
bloc 3



# PROPRIÉTÉ FLEX-WRAP

- La propriété flex-wrap précise si les articles flex doivent être encapsulés ou non sur la m<sup>e</sup>me ligne.
- Elle peut prendre les valeurs suivantes :
  - Wrap : si ligne est pleine on revient à la ligne
  - Nowrap: mem si la ligne est pleine on ne revient pas à la ligne





**`flex-direction: row-reverse;`**  
**`Flex-wrap: wrap;`**

# LA PROPRIÉTÉ FLEX-FLOW

- La propriété flex-flow est un raccourci pour flex-direction et flex-wrap properties.

Exemple

```
.container {  
  display: flex;  
  flex-flow: row wrap;  
}
```

# LA PROPRIÉTÉ JUSTIFY-CONTENT

- La propriété justify-content est utilisée pour aligner les éléments flex
- Elle a les valeurs suivantes :
  - **flex-start** : agence les élément du début du conteneur
  - **flex-end** : agence les élément depuis la fin du conteneur
  - **Center** : au milieu
  - **space-around** : agence les élément en distribuant l'espace du conteneur entre les éléments flex
  - **space-between** : les articles sont répartis uniformément dans la ligne ; le premier article est sur le début de ligne de départ, le dernier sur la fin de
  - **space-evenly**: les éléments sont répartis de manière à ce que l'espacement avant et après entre deux éléments adjacents soit le même

bloc 1

bloc 2

bloc 3

flex-start

flex-end

bloc 1

bloc 2

bloc 3

space-around

bloc 1

bloc 2

bloc 3

space-between

bloc 1

bloc 2

bloc 3

space-evenly

bloc 1

bloc 2

bloc 3

# RETOUR À HTML : PARTIE 3

13

# LES CADRES

- Nous avons déjà vu comment on peut agencer les éléments HTML dans une page à l'aide des balises `<div>` et `<span>`.
- Nous avons pu aussi réaliser des menus intéressants.
- Supposons qu'on désire que ce menu soit repris sur toutes les pages du site:
  - si on réécrit le code nous allons le dupliquer et ce n'est pas intéressant.
- Supposons aussi qu'on clique sur un lien du menu et qu'on veuille que la page web correspondante s'affiche au centre.
- Le seul moyen de le faire en HTML est d'utiliser la balise **`<iframe>`**.

# LES CADRES

- L'utilité de cette balise est de se comporter comme un conteneur pouvant afficher une page web locale ou distante.
- Pour les navigateurs ne supportant pas cette balise on peut ajouter du texte pour l'indiquer.
- On ne peut mettre que du texte entre les deux balise `<iframe>` et `</iframe>`.

# BALISE <IFRAME>

- **Code HTML :**

<iframe

**src="demo.html"**

name="centre"

class= " nom du cadre">

Votre navigateur ne supporte pas les iframes

</iframe>

- Pour afficher une page locale dans ce cadre il suffit de mettre dans la propriété **target** le **nom** du cadre.



# BALISE <IFRAME>

- Première étape c'est de créer un fichier index.html où nous allons définir les deux cadres :

```
<body>
<iframe src="entete.html" scrolling="no" class="menu" name="menu">
</iframe>
<iframe src="accueil.html" class="central" name="centre">
</iframe>
</body>
```

# MISE EN FORME DES CADRES



# PAGE INDEX

```
1 <!DOCTYPE html>
2 <html>
3   <head>
4     <meta charset="utf-8">
5     <link rel="stylesheet" href="cadre.css"/>
6     <title>exemple 61</title>
7   </head>
8   <body>
9     <iframe src="entete.html" scrolling="no" class="menu" name="menu"></iframe>
10    <iframe src="corps.html" scrolling="yes" class="central" name="centre"></iframe>
11  </body>
12 </html>
```

# PAGE ENTETE

```
1  <!DOCTYPE html>
2  <html>
3  <head>
4      <meta charset="utf-8">
5      <title>page entete</title>
6      <link rel="stylesheet" href="cadre.css"/>
7  </head>
8  <body>
9  <table>
10 <tr>
11 <td><a href="accueil.html" target="centre">Accueil</a></td>
12 <td><a href="infos.html" target="centre">Formations</a></td>
13 <td><a href="recherche.html" target="centre">Recherche</a></td>
14 <td><a href="contact.html" target="centre">Actualité</a></td>
15 <td><a href="login.html" target="centre">Contact</a></td><td ></td>
16 </tr>
17 </table>
18 </body>
19 </html>
```

# PAGE CORPS

```
1  <!DOCTYPE html>
2  <html>
3  <head>
4      <meta charset="utf-8"/>
5      <title>page corps</title>
6  </head>
7  <body>
8      <h1> Bienvenue sur mon site</h1>
9  </body>
10 </html>
11
```

# PAGE ACCUEIL

```
1 <!DOCTYPE HTML>
2 <html>
3 <head>
4   <link rel="stylesheet" href="cadre.css" />
5   <!--<link rel="stylesheet" media="screen and (max-width: 1280px)" href="mystyles.css" />-->
6   <meta charset="utf-8"/>
7 </head>
8 <body>
9   <h1>Page d'accueil</h1>
10  <p>
11    c'est le contenu de la page d'accueil qui s'affiche dans
12    le cadre nommé "centre"
13    quand je clique sur accueil
14  </p>
15 </body>
16 </html>
```

# CADRE.CSS

```
1  .menu{
2      position:absolute;
3      text-align:center;
4      background-color:rgb(210, 180, 222);
5      border-radius:10px 10px 10px 10px;
6      padding-bottom:6px;
7      left:10%;
8      width:80%;
9      height:6%;
10     border:none;
11 }
12  .central{
13     position:absolute;
14     left:10%;
15     top:10%;
16     width:80%;
17     height:100%;
18     background-color:rgb(215, 189, 226 );
19     border-radius:10px;
20     padding:2px;
21     border:none;
22 }
23  td{  width:20%;padding:5px;  }
24  a{text-decoration:none; color : white; font-weight:bold; }
25  p{text-align:justify;}
```

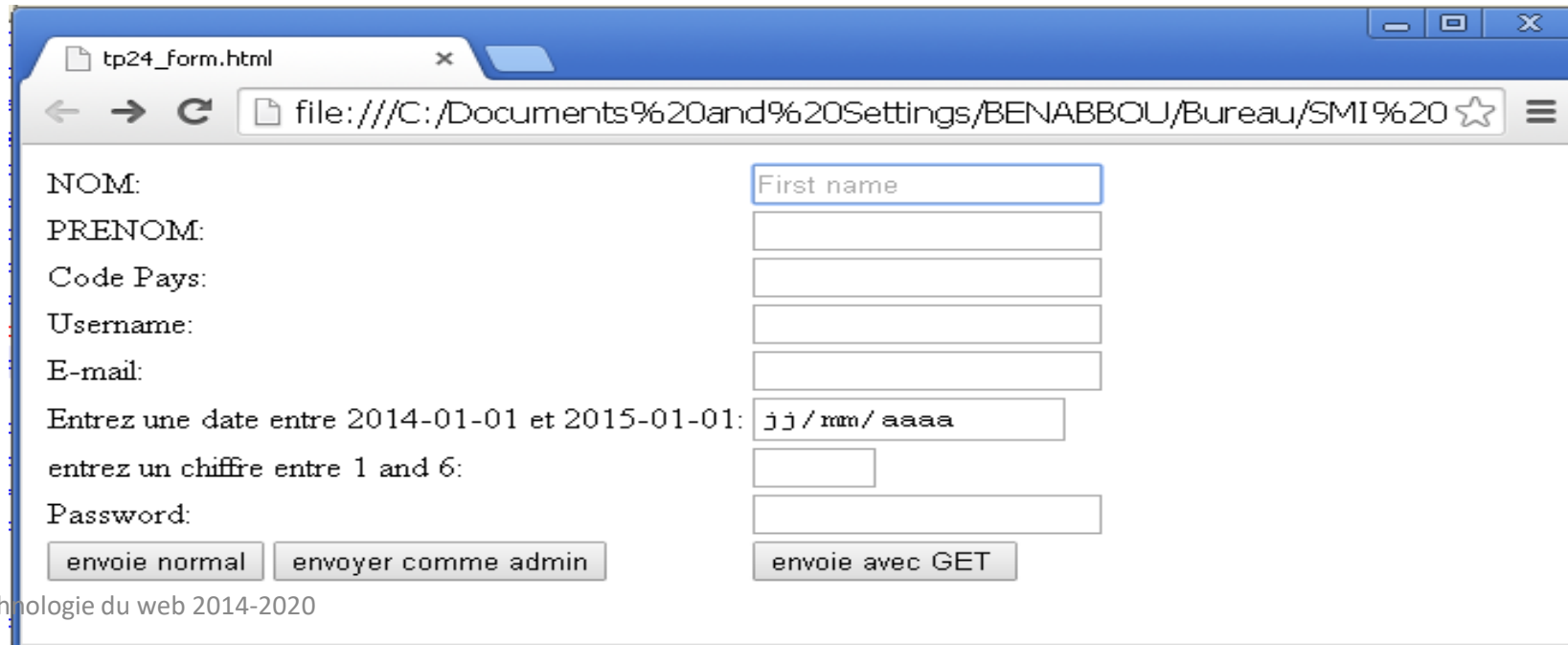
# AUTRES ATTRIBUTS

loading	eager lazy	Specifies whether a browser should load an iframe immediately or to defer loading of iframes until some conditions are met
referrerpolicy	no-referrer no-referrer-when-downgrade origin origin-when-cross-origin same-origin strict-origin-when-cross-origin unsafe-url	Cet attribut permet de définir une politique de fonctionnalité pour notre iframe. choisir quelles fonctionnalités de l'iframe activer ou désactiver.
sandbox	allow-forms allow-pointer-lock allow-popups allow-same-origin allow-scripts allow-top-navigation	Il nous permet de limiter de ce qu'on peut faire à l'intérieur de e notre iframe Question de sécurité



# FORMULAIRES

- Un formulaire HTML est une section d'un document pouvant contenir des balises et des contrôles
- Les contrôles sont des éléments spéciaux qui permettent à l'utilisateur de saisir des données, ex: les cases à cocher, les boutons radio, les labels, les menus,... etc
- Ils permettent alors d'établir un véritable dialogue entre l'utilisateur et le serveur contenant le site.



The screenshot shows a web browser window with the title 'tp24\_form.html'. The address bar shows the file path: 'file:///C:/Documents%20and%20Settings/BENABBOU/Bureau/SMI%20'. The form contains the following elements:

- NOM:
- PRENOM:
- Code Pays:
- Username:
- E-mail:
- Entrez une date entre 2014-01-01 et 2015-01-01:
- entrez un chiffre entre 1 and 6:
- Password:
- Buttons: 'envoi normal', 'envoyer comme admin', and 'envoi avec GET'

# BALISE <FORM>

- **Code HTML**

```
<form name= " nom "  
  action= " script.php "  
  method="get/post "  
  novalidate  
  autocomplete="on/off" >
```

Ici on Définit les éléments du formulaire

```
</form>
```

- Les formulaires sont surtout utilisés conjointement avec des éléments tels que le langage JavaScript ou les programmes PHP.

# ATTRIBUTS DE <FORM>

- **Le nom:** Il s'agit du nom du formulaire, utile pour y faire référence lors de l'utilisation du Javascript.
- **L'action :** c'est le nom du script qui va traiter le formulaire
- **Method :** Spécifie le type de requête *http* qui sera utilisé pour soumettre les données avec GET ou POST.
  - méthode *post* : Données transmises dans le corps du formulaire
  - méthode *get* : Données transmises dans l'*URL* (Concaténation dans la chaîne *HTTP*)
  - GET constitue la méthode par défaut lorsque celle-ci n'est pas spécifiée.

# ATTRIBUTS DE <FORM>

- **ENCTYPE**

- ENCTYPE spécifie comment les données sont envoyées au serveur (type MIME).
- Ce paramètre ne peut être utilisé qu'avec par la méthode POST.

- ENCTYPE peut prendre plusieurs valeurs:

- ENCTYPE="application/x-www-form-urlencoded"
  - encode tout le contenu du formulaire avant son envoie
- ENCTYPE="text/plain"
  - le contenu du formulaire sera envoyé en format texte lisible au serveur
- ENCTYPE="multipart/form-data"
  - Utilisé lors de l'envoi d'un fichier (<input type=" file"/>

# ATTRIBUTS DE <FORM>

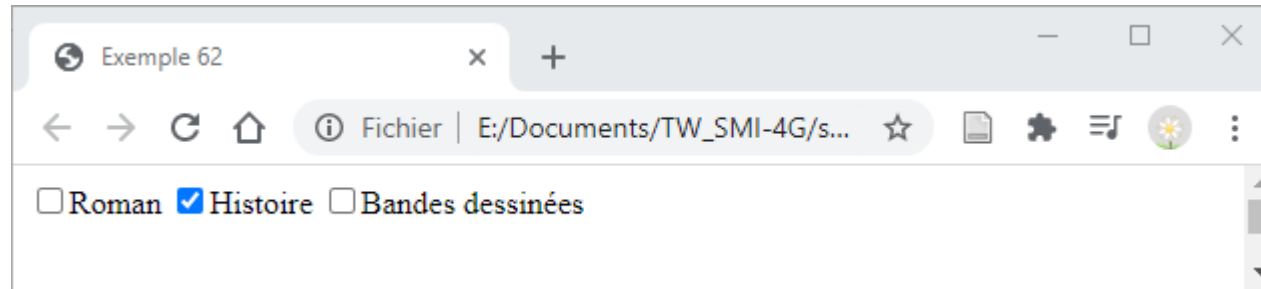
- **novalidate** c'est un nouveau attribut qui permet l'envoi sans validation des champs du formulaire
- **autocomplete="on/off"** c'est un nouveau attribut, son rôle est au moment de la saisi c'est de compléter le champs ou nom par des valeurs que l'utilisateur a déjà saisi
- La mise en page des formulaires se fait usuellement à l'aide de tableau (<TABLE>) dont on ne fait pas apparaître les bordures.

# LES ÉLÉMENTS D'UN FORMULAIRE

- On peut trouver l'élément input qui peut être :
  - case à cocher
  - case radio
  - boîte à texte
  - mot de passe et entrée cachée
  - entrée fichier
  - **nouveau en HTML 5** : color, date, datetime, datetime-local, email, month, time, number, range, tel, search, url, week
- Listes
  - boîte liste
  - boîte liste déroulante
  - boîte texte multilignes
- bouton
  - bouton soumettre
  - bouton recommencer
  - image de formulaire

# LES CASES À COCHER

- Les cases à cocher présentent une liste d'items à l'utilisateur. Celui-ci peut alors cocher un ou plusieurs items afin d'indiquer ses choix.
- Exemple:



- Les cases à cocher sont appropriées dans ce cas-ci, puisque l'utilisateur **peut avoir plus** d'un choix.

# FORMULAIRE : LES CASES À COCHER

- Code HTML

```
1  <!DOCTYPE HTML>
2  <html>
3  <head>
4    <link rel="stylesheet" href="styles62.css" />
5    <meta charset="utf-8" />
6    <title>Exemple 62</title>
7  </head>
8  <body>
9    <form>
10     <input type="checkbox" name="livre1" value="Roman"/>Roman
11     <input type="checkbox" name="livre2" value="Histoire" checked />Histoire
12     <input type="checkbox" name="livre3" value="Bandes dessinées"/>Bandes dessinées
13     <input type="password"/>
14   </form>
15 </body>
16 </html>
```

- **CHECKED** est optionnel et sa présence indique que la case est **cochée**.

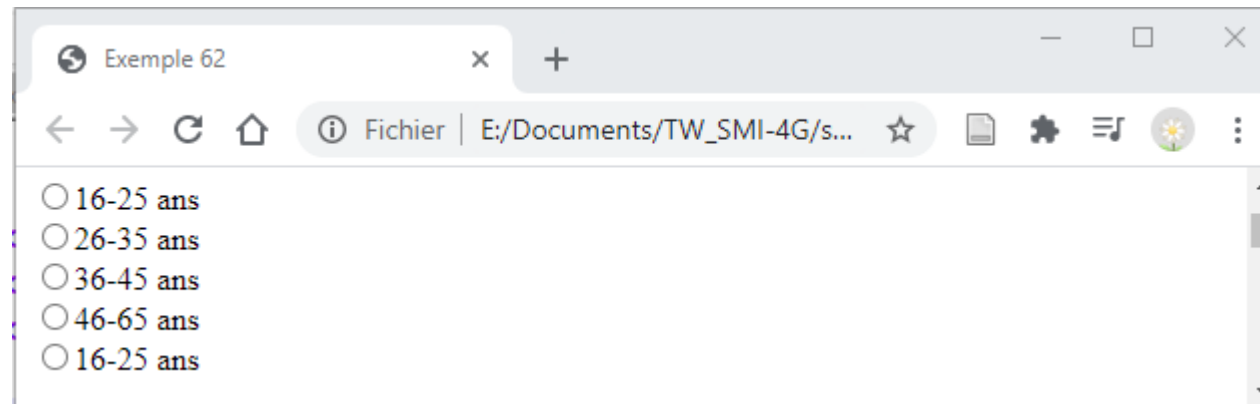


# FORMULAIRE : LES CASES RADIO

- **Code HTML**

`<input type="radio" name="nom" value="valeur " checked /> Texte`

- Remarquez dans l'exemple ci-haut que toutes les cases radio portent le même nom identificateur.
- CHECKED est optionnel et permet de spécifier la case radio cochée.



The screenshot shows a web browser window titled 'Exemple 62'. The address bar displays 'Fichier | E:/Documents/TW\_SMI-4G/s...'. The main content area contains a vertical list of five radio buttons, each followed by the text 'ans':

- ☐ 16-25 ans
- ☐ 26-35 ans
- ☐ 36-45 ans
- ☐ 46-65 ans
- ☐ 16-25 ans

# FORMULAIRE : LES CASES RADIO

- Les cases radio présentent également une liste de choix à l'utilisateur.
- Cependant, il ne peut effectuer qu'un seul choix parmi la liste.
- Supposons que l'on désire connaître dans quel groupe d'âge se trouve le visiteur, on obtiendrait les cases radio suivantes:

```
30
31 <form>
32 <input type="radio" name="choix" value="1"/>16-25 ans<br/>
33 <input type="radio" name="choix" value="2"/>26-35 ans<br/>
34 <input type="radio" name="choix" value="3"/>36-45 ans<br/>
35 <input type="radio" name="choix" value="4"/>46-65 ans<br/>
36 <input type="radio" name="choix" value="5"/>16-25 ans<br/>
37 </form>
38
```

# FORMULAIRE: LISTE DÉROULANTE

- La boîte à liste représente une liste de choix
- Un élément seul est apparent lorsque attribut size n'est pas défini.
- Code html :

```
<select name="livres" >
```

```
<option value="Apprendre HTML"> Apprendre HTML
```

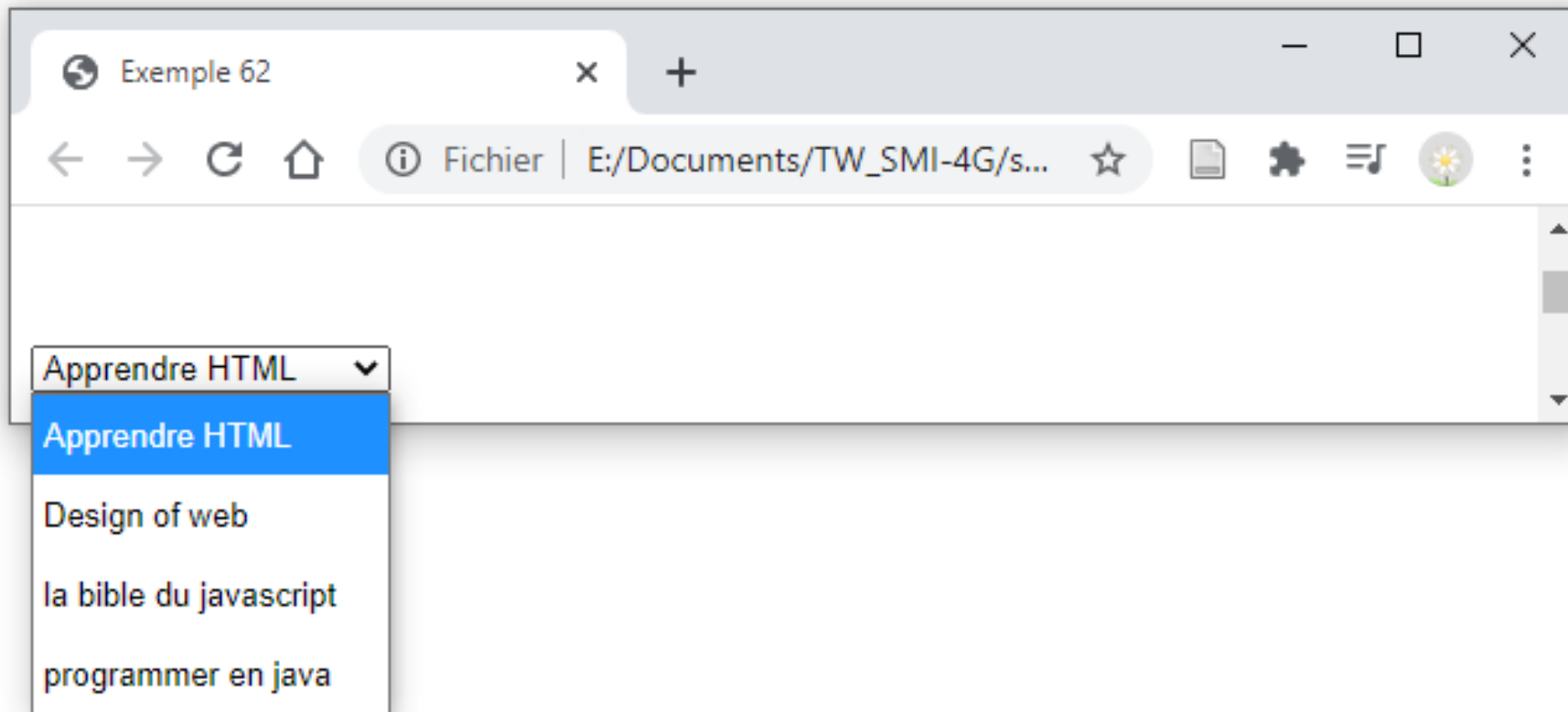
```
<option value="Design of web"> Design of web
```

```
<option value="la bible du javascript"> la bible du javascript
```

```
<option value="programmer en java"> programmer en java
```

```
</select>
```

# FORMULAIRE: LISTE DÉROULANTE



# FORMULAIRE: LISTE DÉROULANTE

```
42 <form>
43 <select name="livres" >
44 <option value="Apprendre HTML"> <label>Apprendre HTML</label>
45 <option value="Design of web"> <label>Design of web</label>
46 <option value="la bible du javascript"><label>la bible du javascript</label>
47 <option value="programmer en java"><label>programmer en java</label>
48 </select>
49 </form>
```

# ATTRIBUT DE LA LISTE DÉROULANTE

- ◆ **Name**: nom identificateur
- ◆ **Multiple**: s'il est présent, permet à l'utilisateur d'effectuer plus d'une sélection.
- ◆ **Size** : indique le nombre de lignes qui seront visibles dans la liste.
- ◆ Attribut de la balise <option>
  - **Value**: valeur de la sélection
  - **Selected** : option sélectionnée par défaut au chargement de la page
- ◆ On peut sélectionner plus d'un items à l'aide de la touche SHIFT ou CTRL.

# EXEMPLE

```
55 <form>
56 <select name="livres" size="3" multiple>
57 <option value="apprendre à naviguer avec netscape">
58   <label> apprendre à naviguer avec netscape</label>
59 <option value="créez vos graphiques pour le web">
60   <label> créez vos graphiques pour le web</label>
61 <option value="la bible du javascript">
62   <label>la bible du javascript</label>
63 <option value="le livre du html">
64   <label>le livre du html</label>
65 <option value="programmer en java">
66   <label>programmer en java</label>
67 </select>
68
```

# DES SUGGESTIONS DE CHOIX

- Nouvelle balise dans HTML5, **<datalist>** est un élément de formulaire permettant de lier **une liste de choix** à un ou plusieurs éléments **<input>**.
- Fourni des fonctionnalités d'auto-complétion ou d'auto-suggestion, dans un formulaire classique ou un champ de recherche...
- Par défaut, **<datalist>** demeure **invisible**.
- Lors de la frappe, ou en appuyant sur la touche ↓ (flèche bas) , la liste de choix apparaît et auto-complète le champ de formulaire rattaché.



# EXEMPLE

- **Code HTML :**

```
<label for="recherche">Indiquez votre Moteur de recherche préféré:</label>
```

```
<input list="moteurs" type="text" name="recherche ">
```

```
  <datalist id=" moteurs ">
```

```
    <option value="Google">
```

```
    <option value="Bing">
```

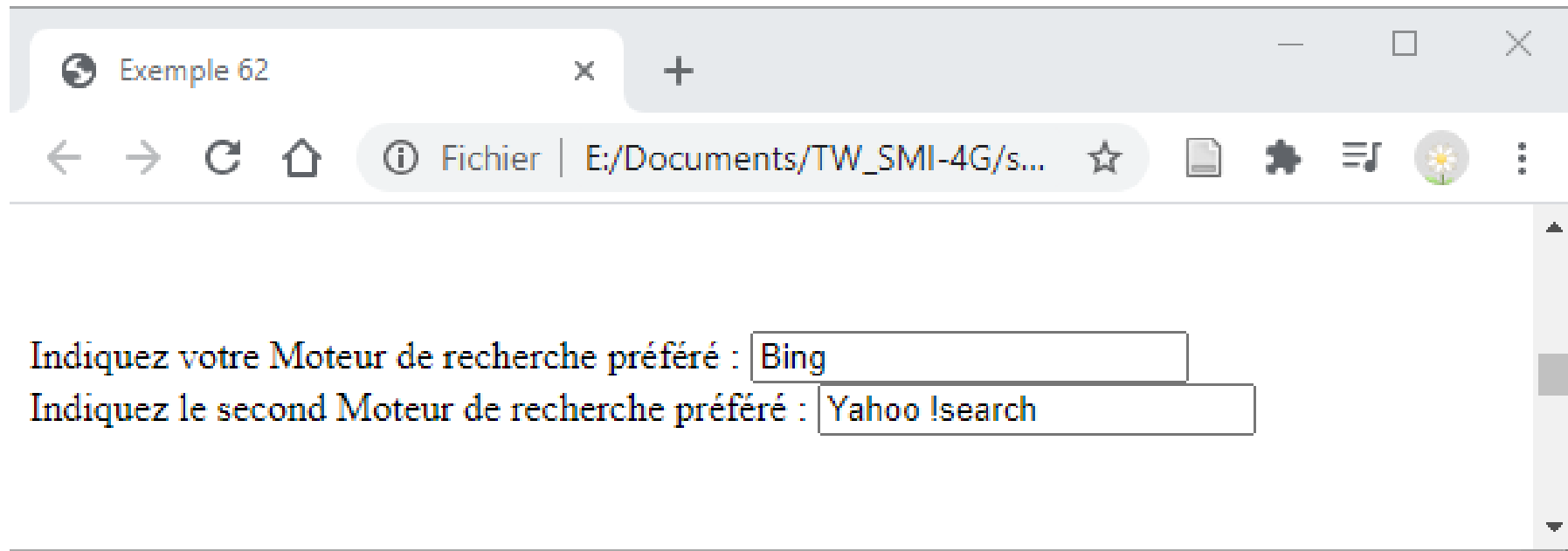
```
    <option value="Ask">
```

```
    <option value="Yahoo !search">
```

```
  <option value="chrome">
```

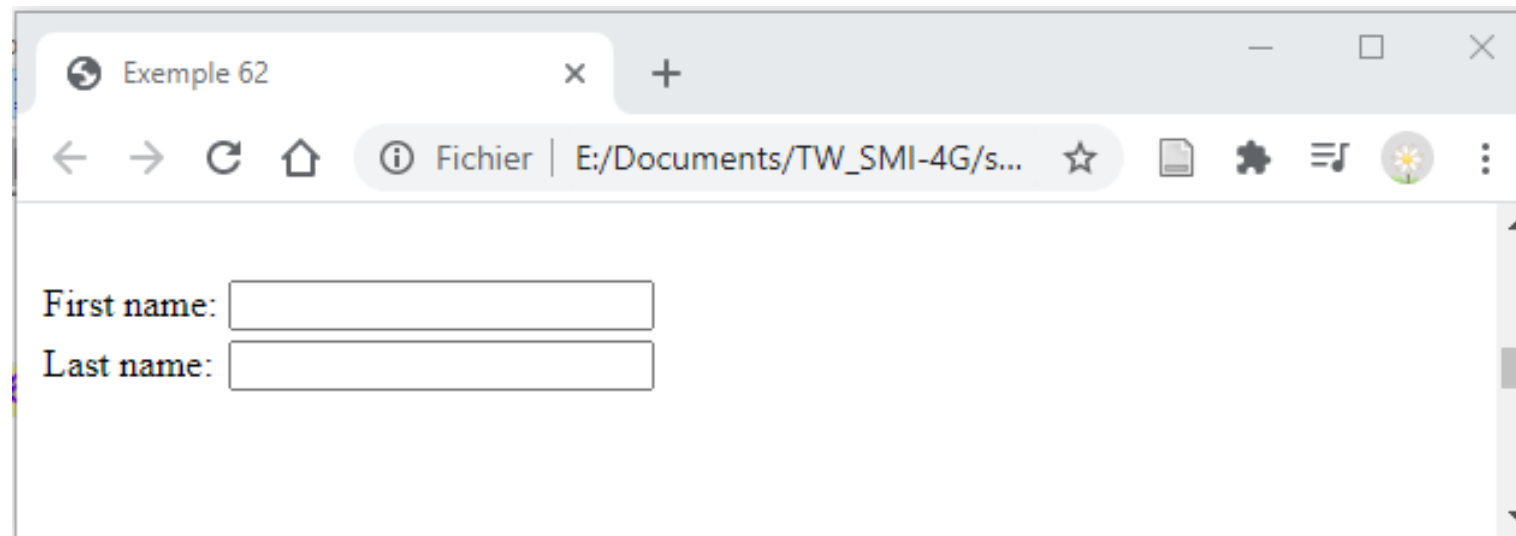
```
</datalist>
```

# RÉSULTAT



# BOÎTE À TEXTE

- La boîte texte permet de saisir du texte.
- On s'en sert pour recueillir une entrée au clavier de la part de l'utilisateur, comme le nom, prénom... de l'utilisateur par exemple.



The screenshot shows a web browser window titled 'Exemple 62'. The address bar displays 'Fichier | E:/Documents/TW\_SMI-4G/s...'. The main content area contains a form with two text input fields. The first field is labeled 'First name:' and the second field is labeled 'Last name:'. Both fields are empty and have a light gray border.

# INPUT DE TYPE TEXT

- Code HTML:

**<label>NOM:</label>**

**<input type="text" name="fname"**

***autofocus***

***placeholder="First name"***

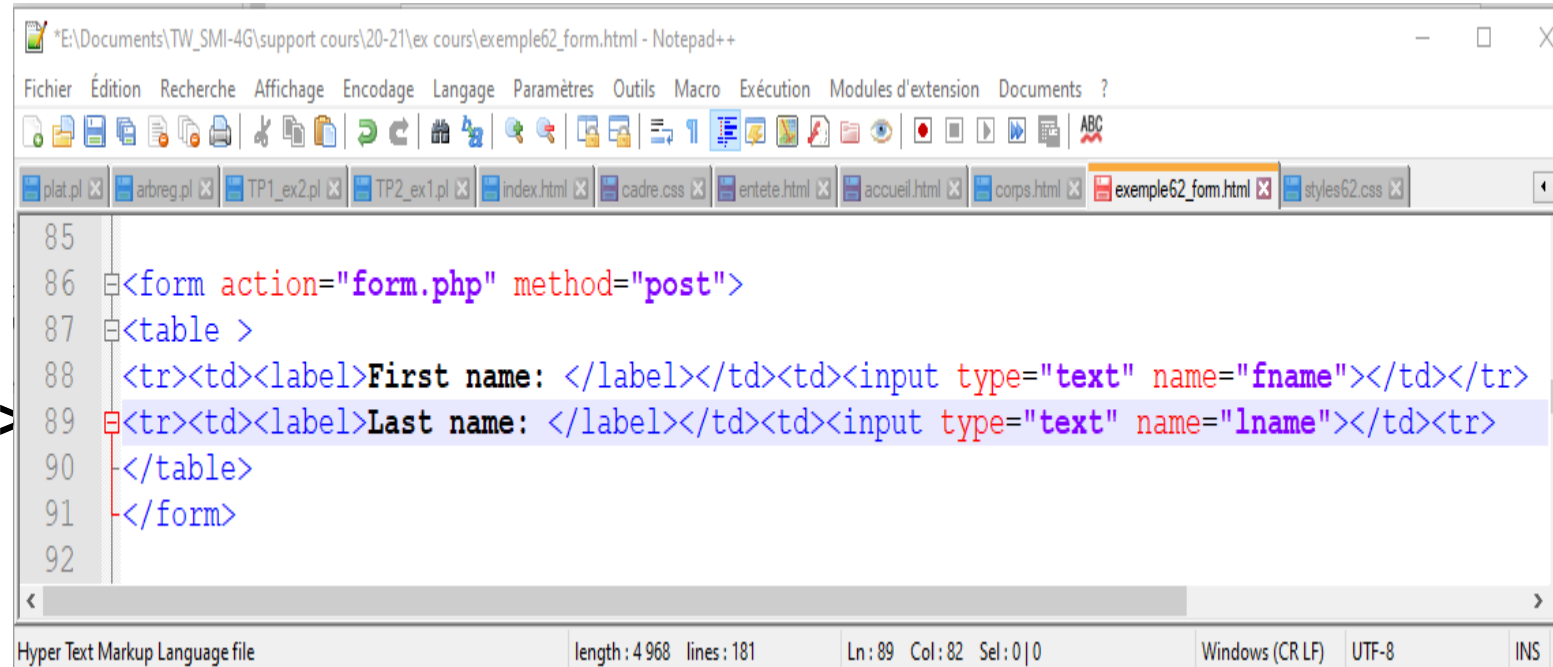
***autocomplete="off"***

**size= "10"**

**maxlength= " 20 "**

***required***

**pattern="[A-Za-z]{5}"/>**



```
85
86 <form action="form.php" method="post">
87 <table >
88 <tr><td><label>First name: </label></td><td><input type="text" name="fname"></td></tr>
89 <tr><td><label>Last name: </label></td><td><input type="text" name="lname"></td></tr>
90 </table>
91 </form>
92
```

# INPUT DE TYPE TEXT: ATTRIBUTS

- Cet input permet de saisir du texte, ses attributs sont :
  - **name**: le nom que porte le champ, qui permettra de l'identifier par la suite.
  - **value**: la valeur du champ.
  - **disabled**: permet de désactiver le champ, qui devient alors grisé et inutilisable dans la page.
  - **readonly**: de la même façon que "disabled" rend un champ inopérant, "readonly" permet d'avoir un champ avec une valeur mais non modifiable.
  - **Placeholder** donne une indication sur ce qui va être saisi ex: "First name"
  - l'attribut **autofocus** permet de placer le curseur en premier dans ce champ
  - **size** détermine le nombre de caractères maximal permis
  - **maxlength** fixe le nombre maximal de caractères visible à l'écran.
  - **pattern** définit une expression régulière que doit être respectée lors de la saisi( ex. 5 caractère alphabétiques minuscule ou majuscule).
  - **required** veut dire que ce champ doit être obligatoirement par l'utilisateur.

Exemple 62

← → ↻ 🏠 ⓘ Fichier | E:/Documents/TW\_SMI-4G/s... ☆ 📄 ⚙️ 🎵 🌻 ⋮

First name:

Last name:

User: \*

Order:

Country:

# INPUT DE TYPE SUBMIT ET RESET

- Un formulaire n'est transmis au serveur que lorsque l'utilisateur clique sur le bouton soumettre à l'aide de la souris.
- La plupart du temps, il est accompagné du bouton recommencer (reset), permettant ainsi à l'utilisateur d'effacer toutes les entrées du formulaire d'un seul coup.

- **Code HTML**

```
<form action="form.asp" method="post">  
<label>First name: </label><input type="text" name="fname"><br>  
<label>Last name: </label> <input type="text" name="lname"><br>  
  <input type="submit" value="Envoyer"/>  
  <input type="reset" value="Annuler"/>  
</form>
```

```

94
95
96
97 <form action="form.php" method="post" >
98 <table >
99 <tr><td><label>First name: </label></td>
100 <td><input type="text" name="fname" placeholder="Your First name" size="15" maxlength="8"/></td>
101 </tr>
102 <tr><td><label>Last name: </label></td>
103 <td><input type="text" name="lname" placeholder="Your last name" size="15"/></td>
104 </tr>
105 <tr><td><label>User: *</label></td>
106 <td><input type="text" name="matricule" pattern="[A-Za-z1-9]{5}" required/></td>
107 </tr>
108 <tr><td><label>Order: </label></td>
109 <td><input type="text" name="seq" value="124" readonly/></td>
110 </tr>
111 <tr><td><label>Country: </label></td>
112 <td><input type="text" name="origin" value="NA" disabled/></td>
113 </tr>
114 </table>
115 <input type="submit" value="Envoyer">
116 <input type="reset" value="Annuler">
117 </form>
118
119

```