

Les formulaires

Plan

- 1. Principe de fonctionnement d'un formulaire**
- 2. Création des formulaires**
 - A- Méthode de transmission et script de réception**
 - B-Création des champs de formulaire**
- 1. Les éléments de formulaire html**
- 2. Les types d'entrée de l'élément input**
- 3. Les attributs de l'élément input**

1-Principe de fonctionnement d'un formulaire

- Un formulaire HTML est utilisé pour collecter les entrées de l'utilisateur qui sont souvent envoyées à un serveur pour traitement.
- Les données d'un formulaire sont validées syntaxiquement.

Client

Serveur

index.HTML

formulaire

.....

.....

.....

.....

Valider

Client

Serveur

index.HTML

formulaire

.....

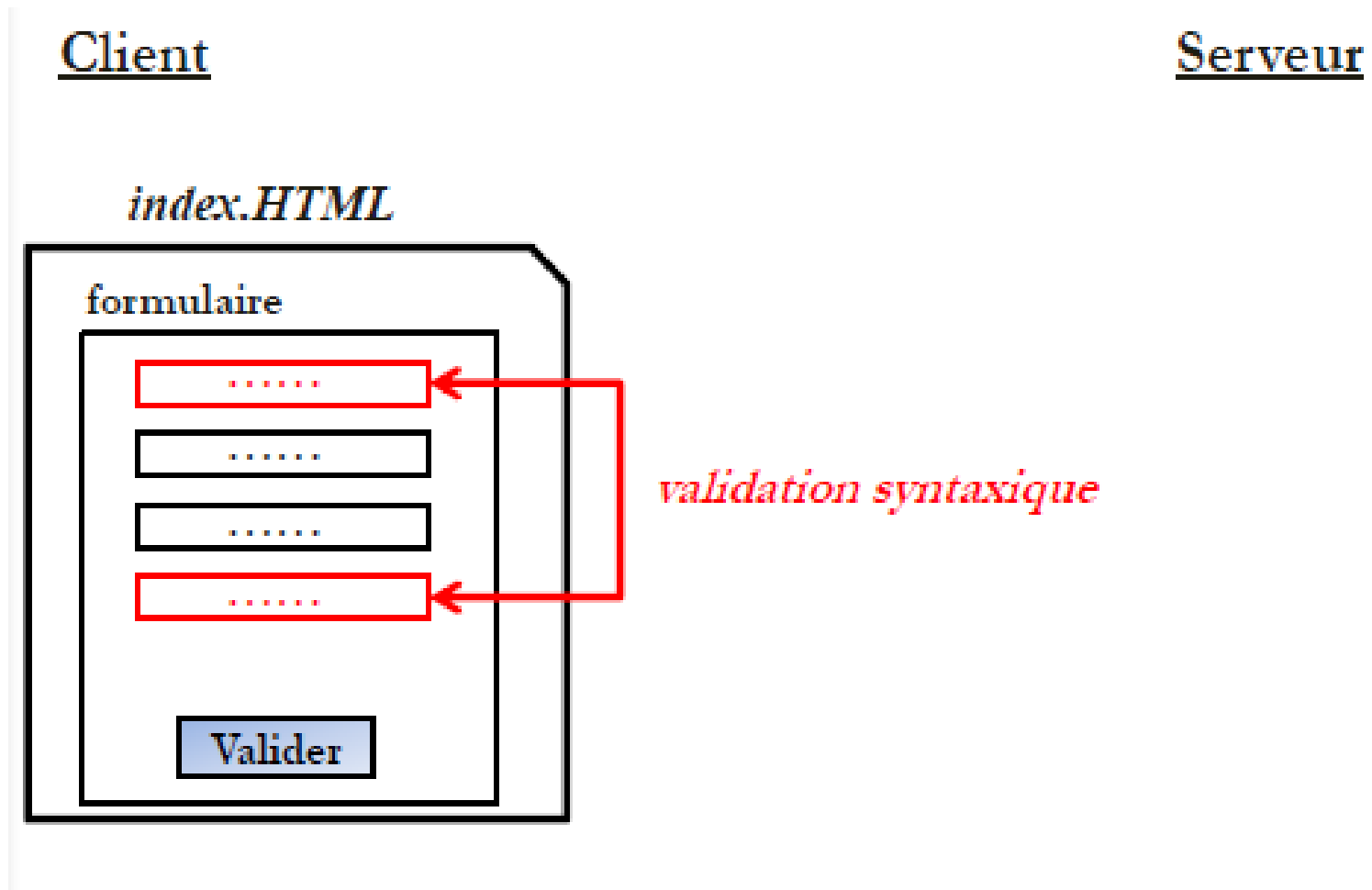
.....

.....

.....

Valider

validation syntaxique



Client

index.HTML

formulaire

.....
.....
.....
.....

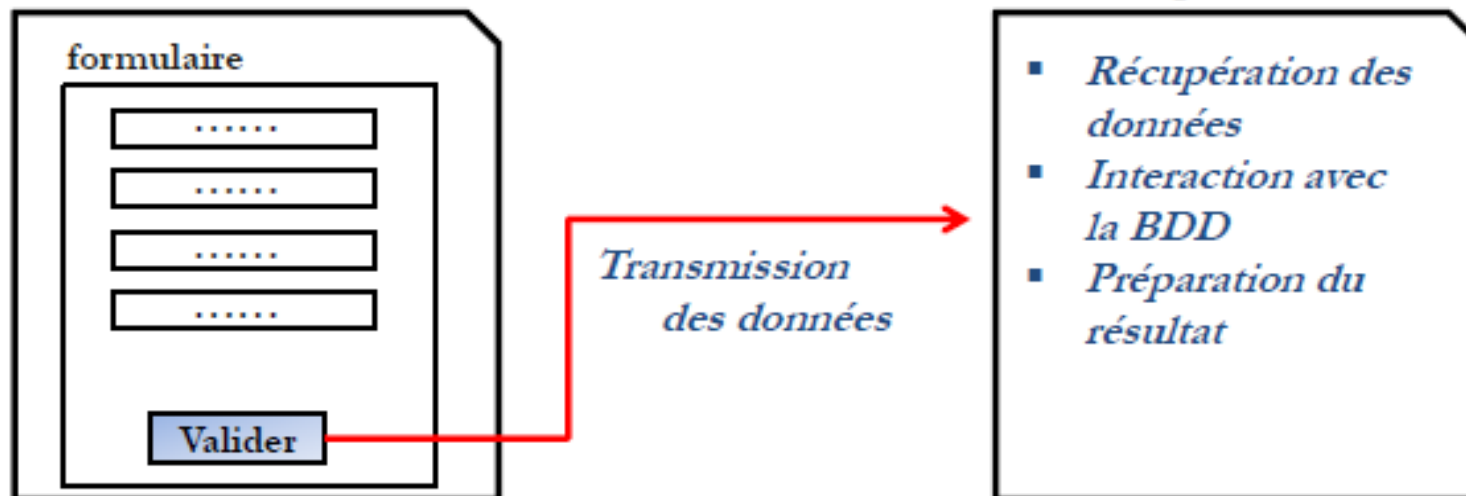
Valider

Serveur

script.PHP

- *Récupération des données*
- *Interaction avec la BDD*
- *Préparation du résultat*

*Transmission
des données*



Client

index.HTML

formulaire

Valider

*HTML + CSS
+ JavaScript*

Serveur

script.PHP

- *Récupération des données*
- *Interaction avec la BDD*
- *Préparation du résultat*

*Transmission
du résultat*



2- Création des champs du formulaire

- Pour créer un formulaire il faut:

A. Préciser la méthode de transmission des données ,
et spécifier le script de réception

B. Créer les champs du formulaire

C. Définir les règles de validation syntaxique des données

} Du côté client

D. Programmer le script de réception des données de formulaire

} Du côté serveur

A- Préciser la méthode de transmission
et le script de réception

- Pour créer un formulaire on utilise l'élément `<form> </form>`
- `<form method = " GET ou POST " action=" traitement.php">`
.....
- `</form>`
- La méthode de transmission est spécifiée avec l'attribut **method** de l'élément form.
- Le script de réception est spécifié avec l'attribut **action** de l'élément form

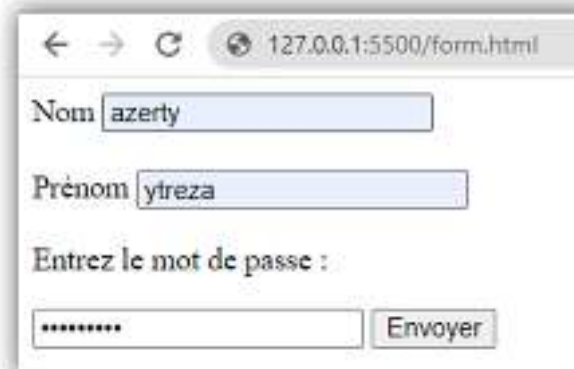
L'attribut **method**

- Spécifie la méthode HTTP à utiliser lors de la soumission des données du formulaire.
- Les données de formulaire peuvent être envoyées sous deux formes : **GET** et **POST**.
- La méthode HTTP par défaut lors de la soumission des données de formulaire est GET.

GET:

- Avec method = “get”, les données de formulaire sont envoyés sous forme de **variables URL** et par paires **nom/valeur**.
- Get n’est JAMAIS utilisée pour envoyer des données sensibles Car elles seront visibles dans l'URL
- La longueur d'une URL est limitée 2048 caractères

```
<form action="" method="get">  
  <label for="nm" >Nom</label>  
  <input type="text" name="nm"> <br>  
  
  <br><label for="prnm" >Prénom</label>  
  <input type="text" name="prnm"> <br>  
  
  <p>Entrez le mot de passe :</p>  
  
  <input type="password" name="pwd">  
  <input type="submit" value="Envoyer">  
  
</form>
```



← → ↻ 127.0.0.1:5500/form.html

Nom

Prénom

Entrez le mot de passe :

127.0.0.1:5500/form.html?nm=azerty&prnm=ytreza&pwd=123456789

POST: .

- Post Ajoute les données du formulaire dans le **corps de la requête HTTP** et donc elles ne sont pas affichées dans l'URL
- POST n'a aucune limitation de taille et peut être utilisé pour envoyer de grandes quantités de données.

L'attribut **action**

- Définit l'action à effectuer lors de la soumission du formulaire.
- Habituellement, les données du formulaire sont envoyées vers un fichier sur le serveur lorsque l'utilisateur clique sur le bouton Soumettre.
- Exemple : `<form action=" traitement.php" =>` Lors de la soumission, envoyez les données du formulaire à " traitement.php"
- Si l'attribut action est omis, l'action est définie sur **la page actuelle**

- L'élément <form> peut contenir d'autres attributs.
- Les attributs les plus utilisés sont: **target, autocomplete et novalidate...**

L'attribut **Target**

- L' attribut Target spécifie où afficher la réponse reçue après la soumission du formulaire.
- L' attribut Target peut avoir l'une des valeurs suivantes :
- **_blank** (nouvel onglet), **_self** (même onglet),etc
- La valeur par défaut est **_self** ce qui signifie que la réponse s'ouvrira dans la fenêtre actuelle

Attribut Autocomplete

- spécifie si un formulaire doit avoir la saisie semi-automatique activée ou désactivée.
- Lorsque la saisie semi-automatique est activée, le navigateur complète automatiquement les valeurs en fonction des valeurs que l'utilisateur a saisie auparavant.

```

<!DOCTYPE html>
<html>
<body>

<h1>The form autocomplete attribute</h1>

<p>Fill in and submit the form, then reload the page,
start to fill in the form again - and see how autocomplete
works.</p>

<p>Then, try to set autocomplete to "off".</p>

<form action="/action_page.php" autocomplete="on">
  <label for="fname">First name:</label>
  <input type="text" id="fname" name="fname"><br><br>
  <label for="email">Email:</label>
  <input type="text" id="email" name="email"><br><br>
  <input type="submit">
</form>

</body>
</html>

```

The form autocomplete attribute

Fill in and submit the form, then reload the page, start to fill in the form again - and see how autocomplete works.

Then, try to set autocomplete to "off".

First name:
azerty

Email:

Attribut **Novalidate**

- Attribut booléen, Lorsqu'il est présent, il spécifie que les données de formulaire (entrée) ne doivent pas être validées lors de la soumission.
- (désactiver la validation automatique du côté navigateur pour la faire avec du JavaScript ou du côté serveur)

```
<!DOCTYPE html>
<html>
<body>

<h1>The form novalidate attribute</h1>

<p>The novalidate attribute indicates that the form input
is not to be validated on submit:</p>

<form action="/action_page.php" novalidate>
  <label for="email">Enter your email:</label>
  <input type="email" id="email" name="email"><br><br>
  <input type="submit">
</form>

</body>
</html>
```

The form novalidate attribute

The novalidate attribute indicates that the form input is not to be validated on submit.

Enter your email:

Envoyer

B- Créer les champs du formulaire

- `<input>`
- `<label>`
- `<select>`
- `<textarea>`
- `<button>`
- `<fieldset>`
- `<legend>`
- `<datalist>`
- `<output>`
- `<option>`
- `<optgroup>`

L'élément input

- <Input> est l'élément le plus utilisé.
- Il peut être affiché de plusieurs manières selon l'attribut **type**.

L'élément label

- L'élément <label> Définit une étiquette pour plusieurs éléments de formulaire.
- L'attribut **for** de l'élément label doit être égal à l'attribut **id** de l'élément input pour les lier ensemble.

```
<!DOCTYPE html>
<html>
<body>

<h2>L'élément Input</h2>

<form action="Traitement.php">
  <label for="fname">First name:</label><br>
  <input type="text" id="fname" name="fname"><br><br>
  <input type="submit" value="Submit">
</form>

</body>
</html>
```

L'élément Input

First name:

L'élément select

- L'élément **<select>** Définit une liste déroulante
- Les éléments **<option>** définissent une option qui peut être sélectionnée.
- Par défaut, le premier élément de la liste déroulante est sélectionné.

```
<!DOCTYPE html>
<html>
<body>

<h2>The select Element</h2>

<p>The select element defines a drop-down list:</p>

<form action="/action_page.php">
  <label for="cars">Choose a car:</label>
  <select id="cars" name="cars">
    <option value="volvo">Volvo</option>
    <option value="saab">Saab</option>
    <option value="fiat">Fiat</option>
    <option value="audi">Audi</option>
  </select>
  <input type="submit">
</form>

</body>
</html>
```

The select Element

The select element defines a drop-down list:

Choose a car:

- Pour définir une option présélectionnée, ajoutez l'attribut **selected** à l'option

```
<!DOCTYPE html>
<html>
<body>

<h2>Pre-selected Option</h2>

<p>You can preselect an option with the selected attribute:</p>

<form action="/action_page.php">
  <label for="cars">Choose a car:</label>
  <select id="cars" name="cars">
    <option value="volvo">Volvo</option>
    <option value="saab">Saab</option>
    <option value="fiat" selected>Fiat</option>
    <option value="audi">Audi</option>
  </select>
  <input type="submit">
</form>

</body>
</html>
```

Pre-selected Option

You can preselect an option with the selected attribute:

Choose a car:

- Utilisez l'attribut **size** pour spécifier le nombre de valeurs visibles

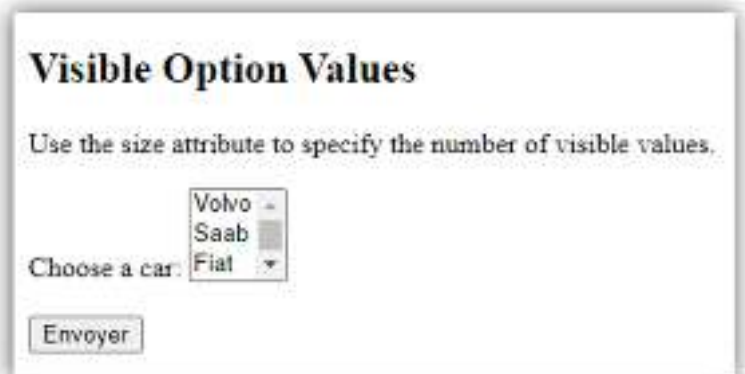
```
<!DOCTYPE html>
<html>
<body>

<h2>Visible Option Values</h2>

<p>Use the size attribute to specify the number of visible values.</p>

<form action="/action_page.php">
  <label for="cars">Choose a car:</label>
  <select id="cars" name="cars" size="3">
    <option value="volvo">Volvo</option>
    <option value="saab">Saab</option>
    <option value="fiat">Fiat</option>
    <option value="audi">Audi</option>
  </select><br><br>
  <input type="submit">
</form>

</body>
</html>
```



Visible Option Values

Use the size attribute to specify the number of visible values.

Choose a car: Volvo
Saab
Fiat

- Utilisez l'attribut **multiple** pour permettre à l'utilisateur de sélectionner plusieurs valeurs

```
<!DOCTYPE html>
<html>
<body>

<h2>Allow Multiple Selections</h2>

<p>Use the multiple attribute to allow the user to select more than one
value.</p>


<form action="/action_page.php">
  <label for="cars">Choose a car:</label>
  <select id="cars" name="cars" size="4" multiple>
    <option value="volvo">Volvo</option>
    <option value="saab">Saab</option>
    <option value="fiat">Fiat</option>
    <option value="audi">Audi</option>
  </select><br><br>
  <input type="submit">
</form>

<p>Hold down the Ctrl (windows) / Command (Mac) button to select multiple
options.</p>

</body>
</html>
```

Allow Multiple Selections

Use the multiple attribute to allow the user to select more than one value.

Choose a car: 

Hold down the Ctrl (windows) / Command (Mac) button to select multiple options.

L'élément textarea

- L'élément <textarea> définit un champ de saisie multiligne (une zone de texte)
- L'attribut **rows** spécifie le nombre de lignes visibles dans une zone de texte.
- L'attribut **cols** spécifie la largeur visible d'une zone de texte.

```
<!DOCTYPE html>
<html>
<body>

<h2>Textarea</h2>
<p>The textarea element defines a multi-line input field.</p>

<form action="/action_page.php">
  <textarea name="message" rows="10" cols="30">The cat was playing in the
garden.</textarea>
  <br><br>
  <input type="submit">
</form>

</body>
</html>
```

Textarea

The textarea element defines a multi-line input field.

The cat was playing in the
garden.

Envoyer

L'élément button

- L'élément `<button>` définit un bouton cliquable
- Spécifier toujours L'attribut **type** de l'élément `<button>`

```
<!DOCTYPE html>
<html>
<body>

<h2>The button Element</h2>

<button type="button" >Click Me!</button>

</body>
</html>
```

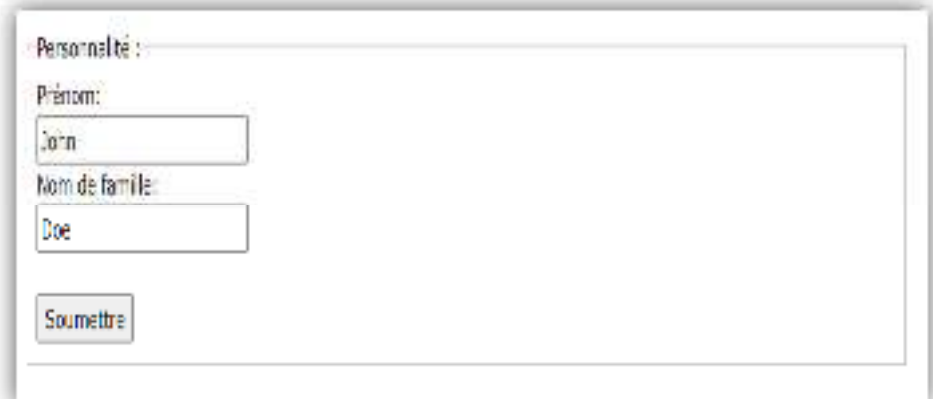
The button Element

Click Me!

Les éléments fieldset et legend

- L'élément `<fieldset>` est utilisé pour regrouper les données associées dans un formulaire.
- L'élément `<legend>` définit une légende pour L'élément `<fieldset>`

```
<form action="/action_page.php">
  <fieldset>
    <legend>Personalia:</legend>
    <label for="fname">First name:</label><br>
    <input type="text" id="fname" name="fname" value="John"><br>
    <label for="lname">Last name:</label><br>
    <input type="text" id="lname" name="lname" value="Doe"><br><br>
    <input type="submit" value="Submit">
  </fieldset>
</form>
```



Personalia :

Prénom:

Nom de famille:

L'élément `datalist`

- L'élément `<datalist>` spécifie une liste d'options prédéfinies pour un élément `<input>`
- Les utilisateurs verront une liste déroulante des options prédéfinies lorsqu'ils saisiront des données..
- L'attribut **list** de l'élément `<input>` doit faire référence à l'attribut **id** de l'élément `<datalist>`.

```

<!DOCTYPE html>
<html>
<body>

<h2>The datalist Element</h2>

<p>The datalist element specifies a list of pre-defined options for an input
element.</p>

<form action="/action_page.php">
  <input list="browsers" name="browser">
  <datalist id="browsers">
    <option value="Internet Explorer">
    <option value="Firefox">
    <option value="Chrome">
    <option value="Opera">
    <option value="Safari">
  </datalist>
  <input type="submit">
</form>

<p><b>Note:</b> The datalist tag is not supported in Safari prior version
12.1.</p>

</body>
</html>

```

The datalist Element

The datalist element specifies a list of pre-defined options for an input element.



Internet Explorer

Firefox

Chrome

Opera

Safari

supported in Safari prior version 12.1.

3- Types d'entrée de l'élément input

- L'attribut type de L'élément <input> peut avoir plusieurs valeurs possibles
- Sa valeur par défaut est **text**

- `<input type="button">`
- `<input type="checkbox">`
- `<input type="color">`
- `<input type="date">`
- `<input type="datetime-local">`
- `<input type="email">`
- `<input type="file">`
- `<input type="hidden">`
- `<input type="image">`
- `<input type="month">`
- `<input type="number">`
- `<input type="password">`
- `<input type="radio">`
- `<input type="range">`
- `<input type="reset">`
- `<input type="search">`
- `<input type="submit">`
- `<input type="tel">`
- `<input type="text">`
- `<input type="time">`
- `<input type="url">`
- `<input type="week">`

Type d'entrée : text

- `<input type="text">` définit un champ de saisie de texte sur une seule ligne :

```
<form>
  <label for="fname">First name:</label><br>
  <input type="text" id="fname" name="fname"><br>
  <label for="lname">Last name:</label><br>
  <input type="text" id="lname" name="lname">
</form>
```



Prénom:

Nom de famille:

Type d'entrée : password (mot de passe)

- `<input type="password">` : définit un champ mot de passe

```
<form>
  <label for="username">Username:</label><br>
  <input type="text" id="username" name="username"><br>
  <label for="pwd">Password:</label><br>
  <input type="password" id="pwd" name="pwd">
</form>
```



Nom d'utilisateur:

Mot de passe:

Type de d'entrée : submit (soumettre)

- `<input type="submit">` définit un bouton pour **soumettre des** données de formulaire à un gestionnaire de formulaire spécifié dans l'attribut action du formulaire.

```
<form action="/action_page.php">  
  <label for="fname">First name:</label><br>  
  <input type="text" id="fname" name="fname" value="John"><br>  
  <label for="lname">Last name:</label><br>  
  <input type="text" id="lname" name="lname" value="Doe"><br><br>  
  <input type="submit" value="Submit">  
</form>
```



Prénom:

Nom de famille:

Type d'entrée: reset (réinitialiser)

- `<input type="reset">` définit un bouton de réinitialisation qui réinitialisera toutes les valeurs du formulaire à leurs valeurs par défaut :

```
<form action="/action_page.php">  
  <label for="fname">First name:</label><br>  
  <input type="text" id="fname" name="fname" value="John"><br>  
  <label for="lname">Last name:</label><br>  
  <input type="text" id="lname" name="lname" value="Doe"><br><br>  
  <input type="submit" value="Submit">  
  <input type="reset">  
</form>
```



Prénom:

Nom de famille:

Type d'entrée : radio

- `<input type="radio">` : définit un **bouton radio** .
- Les boutons radio permettent à un utilisateur de sélectionner UNIQUEMENT UN parmi un nombre limité de choix :

```
<p>Choose your favorite Web language:</p>

<form>
  <input type="radio" id="html" name="fav_language" value="HTML">
  <label for="html">HTML</label><br>
  <input type="radio" id="css" name="fav_language" value="CSS">
  <label for="css">CSS</label><br>
  <input type="radio" id="javascript" name="fav_language"
value="JavaScript">
  <label for="javascript">JavaScript</label>
</form>
```



☐ HTML
☐ CSS
☐ JavaScript

Type d'entrée : checkbox (case à cocher)

- `<input type="checkbox">` définit une case à cocher .
- Les cases à cocher permettent à un utilisateur de sélectionner les options ZÉRO ou PLUS parmi un nombre limité de choix.

```
<form>
  <input type="checkbox" id="vehicle1" name="vehicle1" value="Bike">
  <label for="vehicle1"> I have a bike</label><br>
  <input type="checkbox" id="vehicle2" name="vehicle2" value="Car">
  <label for="vehicle2"> I have a car</label><br>
  <input type="checkbox" id="vehicle3" name="vehicle3" value="Boat">
  <label for="vehicle3"> I have a boat</label>
</form>
```



☐ j'ai un vélo
☐ j'ai une voiture
☐ j'ai un bateau

Type d'entrée : button (bouton)

- `<input type="button">` définit un **bouton**

```
<!DOCTYPE html>
<html>
<body>

<h2>Input Button</h2>

<input type="button" value="Click Me!">

</body>
</html>
|
```

Input Button

Click Me!

Type d'entrée : couleur

- `<input type="color">` utilisée pour les champs de saisie qui doivent *contenir une couleur* .
- Selon la prise en charge du navigateur, un sélecteur de couleurs peut apparaître dans le champ de saisie.

```
<!DOCTYPE html>
<html>
<body>

<h2>Show a Color Picker</h2>

<p>The <strong>input type="color"</strong> is used for input fields that should
contain a color.</p>

<form action="/action_page.php">
  <label for="favcolor">Select your favorite color:</label>
  <input type="color" id="favcolor" name="favcolor" value="#ff0000">
  <input type="submit" value="Submit">
</form>

</body>
</html>
```

Show a Color Picker

The `input type="color"` is used for input fields that should contain a color.

Select your favorite color:



The image shows a web browser rendering of the HTML code. It features a heading "Show a Color Picker", a paragraph explaining the `input type="color"` attribute, and a form with a label "Select your favorite color:". The form contains a color input field showing a red color swatch and a "Submit" button. Below the color swatch, a color picker interface is visible, showing a gradient bar, a circular color selector, and RGB input fields with values 255, 0, and 0.

Type d'entrée : Date

- Le `<input type="date">` est utilisé pour les champs de saisie qui doivent contenir une date.
- Selon la prise en charge du navigateur, un sélecteur de date peut apparaître dans le champ de saisie.

```
<!DOCTYPE html>
<html>
<body>

<h2>Date Field</h2>

<p>The <strong>input type="date"</strong> is used for input fields that should
contain a date.</p>

<form action="/action_page.php">
  <label for="birthday">Birthday:</label>
  <input type="date" id="birthday" name="birthday">
  <input type="submit" value="Submit">
</form>

</body>
</html>
```

Date Field

The `input type="date"` is used for input fields that should contain a date.

Birthday:

janvier 2022 ▾ ↑ ↓

sa	di	lu	ma	me	je	ve
1	2	3	4	5	6	7
8	9	10	11	12	13	14
15	16	17	18	19	20	21
22	23	24	25	26	27	28
29	30	31	1	2	3	4
5	6	7	8	9	10	11

Effacer Aujourd'hui

- Vous pouvez également utiliser les attributs min et max pour ajouter des restrictions aux dates :

```
<!DOCTYPE html>
<html>
<body>

<h2>Date Field Restrictions</h2>

<p>Use the min and max attributes to add restrictions to dates:</p>

<form action="/action_page.php">
  <label for="datemin">Enter a date after 2000-01-01:</label>
  <input type="date" id="datemin" name="datemin" min="2000-01-02"><br><br>

  <label for="datemax">Enter a date before 1980-01-01:</label>
  <input type="date" id="datemax" name="datemax" max="1979-12-31"><br><br>

  <input type="submit" value="Submit">
</form>

</body>
</html>
```

Date Field Restrictions

Use the min and max attributes to add restrictions to dates:

Enter a date after 2000-01-01:

Enter a date before 1980-01-01:

Type d'entrée: email (courriel)

- Le `<input type="email">` est utilisé pour les champs de saisie qui doivent contenir une adresse e-mail.
- Selon la prise en charge du navigateur, l'adresse e-mail peut être automatiquement validée lors de la soumission.

```
<!DOCTYPE html>
<html>
<body>

<h2>Email Field</h2>

<p>The <strong>input type="email"</strong> is used for input fields that should
contain an e-mail address:</p>

<form action="/action_page.php">
  <label for="email">Enter your email:</label>
  <input type="email" id="email" name="email">
  <input type="submit" value="Submit">
</form>

</body>
</html>
```

Email Field

The input type="email" is used for input fields that should contain an e-mail address:

Enter your email:

Type d'entrée: File (fichier)

- Le `<input type="file">` définit un champ de sélection de fichier et un bouton "Parcourir" pour les téléchargements de fichiers.

```
<!DOCTYPE html>
<html>
<body>

<h1>File upload</h1>

<p>Show a file-select field which allows a file to be chosen for upload:</p>
<form action="/action_page.php">
  <label for="myfile">Select a file:</label>
  <input type="file" id="myfile" name="myfile"><br><br>
  <input type="submit" value="Submit">
</form>

</body>
</html>
```

File upload

Show a file-select field which allows a file to be chosen for upload:

Select a file: Aucun fichier choisi

Type d'entrée: hidden (masqué)

- Le `<input type="hidden">` définit un champ de saisie masqué (non visible pour un utilisateur).
- Un champ masqué permet aux développeurs Web d'inclure des données qui ne peuvent pas être vues ou modifiées par les utilisateurs lorsqu'un formulaire est soumis.
- Remarque: Attention, la valeur est visible et peut être modifiée à l'aide des outils de développement ou Afficher la source.

```
<!DOCTYPE html>
<html>
<body>

<h1>A Hidden Field (look in source code)</h1>

<form action="/action_page.php">
  <label for="fname">First name:</label>
  <input type="text" id="fname" name="fname"><br><br>
  <input type="hidden" id="custId" name="custId" value="3487">
  <input type="submit" value="Submit">
</form>
</body>
</html>
```

A Hidden Field (look in source code)

First name:

Type d'entrée Month (mois)

- Le `<input type="month">` permet à l'utilisateur de sélectionner un mois et une année.
- Selon la prise en charge du navigateur, un sélecteur de date peut apparaître dans le champ de saisie.

```
<!DOCTYPE html>
<html>
<body>

<h2>Month Field</h2>

<p>The <strong>input type="month"</strong> allows the user to select a month
and year.</p>

<form action="/action_page.php">
  <label for="bdaymonth">Birthday (month and year):</label>
  <input type="month" id="bdaymonth" name="bdaymonth">
  <input type="submit" value="Submit">
</form>
</body>
</html>
```

Month Field

The `input type="month"` allows the user to select a month and year.

Birthday (month and year):

2022

janv.	févr.	mars	avr.
mai	juin	juil.	août
sept.	oct.	nov.	déc.
Effacer		Ce mois	

Type d'entrée: Number (numéro)

- Le `<input type="number">` définit un champ de saisie numérique .
- Vous pouvez également définir des restrictions sur les numéros acceptés.

```
<!DOCTYPE html>
<html>
<body>

<h2>Number Field</h2>

<p>The <input type="number"> defines a numeric input field.</p>

<p>You can use the min and max attributes to add numeric restrictions in the
input field:</p>

<form action="/action_page.php">
  <label for="quantity">Quantity (between 1 and 5):</label>
  <input type="number" id="quantity" name="quantity" min="1" max="5">
  <input type="submit" value="Submit">
</form>

</body>
</html>
```

Number Field

The input type="number" defines a numeric input field.

You can use the min and max attributes to add numeric restrictions in the input field:

Quantity (between 1 and 5):

Type d'entrée : range (plage de valeurs)

- Le `<input type="range">` définit un contrôle pour entrer un nombre dont la valeur exacte n'est pas importante. Plage par défaut est de 0 à 100. Toutefois, vous pouvez définir des restrictions sur les nombres acceptés avec les attributs `min`, `max` et `step`

```
<!DOCTYPE html>
<html>
<body>

<h2>Range Field</h2>

<p>Depending on browser support: The input type "range" can be displayed as a
slider control.</p>

<form action="/action_page.php" method="get">
  <label for="vol">Volume (between 0 and 50):</label>
  <input type="range" id="vol" name="vol" min="0" max="50">
  <input type="submit" value="Submit">
</form>

</body>
</html>
```

Range Field

Depending on browser support: The input type "range" can be displayed as a slider control.

Volume (between 0 and 50): 

Type d'entrée: Search (Recherche)

- Le `<input type="search">` est utilisé pour les champs de recherche (un champ de recherche se comporte comme un champ de texte normal).

```
<!DOCTYPE html>
<html>
<body>

<h2>Search Field</h2>
<p>The <strong>input type="search"</strong> is used for search fields (behaves
like a regular text field):</p>

<form action="/action_page.php">
  <label for="gsearch">Search Google:</label>
  <input type="search" id="gsearch" name="gsearch">
  <input type="submit" value="Submit">
</form>

</body>
</html>
```

Search Field

The input type="search" is used for search fields (behaves like a regular text field):

Search Google:

Type d'entrée: Tel (téléphone)

- Le `<input type="tel">` est utilisé pour les champs de saisie qui doivent contenir un numéro de téléphone.

```
<!DOCTYPE html>
<html>
<body>

<h2>Telephone Field</h2>

<p>The <strong>input type="tel"</strong> is used for input fields that should
contain a telephone number:</p>

<form action="/action_page.php">
  <label for="phone">Enter a phone number:</label><br><br>
  <input type="tel" id="phone" name="phone" placeholder="123-45-678" pattern="
[0-9]{3}-[0-9]{2}-[0-9]{3}" required><br><br>
  <small>Format: 123-45-678</small><br><br>
  <input type="submit" value="Submit">
</form>

</body>
</html>
```

Telephone Field

The `input type="tel"` is used for input fields that should contain a telephone number:

Enter a phone number:

Format: 123-45-678

Type d'entrée: Time

- Le `<input type="time">` permet à l'utilisateur de sélectionner une heure (pas de fuseau horaire).

Selon la prise en charge du navigateur, un sélecteur de temps peut apparaître dans le champ de saisie.

```
<!DOCTYPE html>
<html>
<body>

<h1>Show a Time Input Control</h1>

<p>The <strong>input type="time"</strong> allows the user to select a time (no
time zone):</p>

<p>If the browser supports it, a time picker pops up when entering the input
field.</p>

<form action="/action_page.php">
  <label for="appt">Select a time:</label>
  <input type="time" id="appt" name="appt">
  <input type="submit" value="Submit">
</form>
</body>
</html>
```

Show a Time Input Control

The `input type="time"` allows the user to select a time (no time zone):

If the browser supports it, a time picker pops up when entering the input field.

Select a time:

11	05	AM
12	06	PM
01	07	
02	08	
03	09	
04	10	
05	11	

Type d'entrée : URL

- Le `<input type="url">` est utilisé pour les champs de saisie qui doivent contenir une adresse URL.
- Selon la prise en charge du navigateur, le champ URL peut être automatiquement validé lors de la soumission.

```
<!DOCTYPE html>
<html>
<body>

<h1>Display a URL Input Field</h1>

<p>The <strong>input type="url"</strong> is used for input fields that should
contain a URL address:</p>

<form action="/action_page.php">
  <label for="homepage">Add your homepage:</label>
  <input type="url" id="homepage" name="homepage">
  <input type="submit" value="Submit">
</form>

</body>
</html>
```

Display a URL Input Field

The `input type="url"` is used for input fields that should contain a URL address:

Add your homepage:

Type d'entrée: Week (semaine)

- Le `<input type="week">` permet à l'utilisateur de sélectionner une semaine et une année.
- Selon la prise en charge du navigateur, un sélecteur de date peut apparaître dans le champ de saisie.

```
<!DOCTYPE html>
<html>
<body>

<h1>Display a Week Input Control</h1>

<p>The <strong>input type="week"</strong> allows the user to select a week and
year.</p>

<p>If the browser supports it, a date picker pops up when entering the input
field.</p>

<form action="/action_page.php">
  <label for="week">Select a week:</label>
  <input type="week" id="week" name="week">
  <input type="submit" value="Submit">
</form>
</body>
</html>
```

Display a Week Input Control

The `input type="week"` allows the user to select a week and year.

If the browser supports it, a date picker pops up when entering the input field.

Select a week: Semaine --, ---- ☐

janvier 2022 ▾ ↑ ↓

Semaine	sa	di	lu	ma	me	je	ve
1	1	2	3	4	5	6	7
2	8	9	10	11	12	13	14
3	15	16	17	18	19	20	21
4	22	23	24	25	26	27	28
5	29	30	31	1	2	3	4
6	5	6	7	8	9	10	11

Effacer Cette semaine

4- Les attributs de l'élément input

Value, Readonly

- **Value:** spécifie une valeur initiale pour un champ de saisie
- **Readonly:** spécifie qu'un champ de saisie est en lecture seule.

```
<form>  
  <label for="fname">First name:</label><br>  
  <input type="text" id="fname" name="fname" value="John" readonly><br>  
  <label for="lname">Last name:</label><br>  
  <input type="text" id="lname" name="lname" value="Doe">  
</form>
```

The input readonly attribute

The readonly attribute specifies that an input field should be read-only (cannot be changed):

First name:

Last name:

Disabled

- **Disabled:** spécifie qu'un champ de saisie doit être désactivé.
Un champ de saisie désactivé est inutilisable et non cliquable.
La valeur d'un champ de saisie désactivé ne sera pas envoyée lors de la soumission du formulaire

```
<!DOCTYPE html>
<html>
<body>

<h1>The input disabled attribute</h1>

<p>The disabled attribute specifies that an input field should be disabled
(unusable and un-clickable):</p>

<form action="/action_page.php">
  <label for="fname">First name:</label><br>
  <input type="text" id="fname" name="fname" value="John" disabled><br>
  <label for="lname">Last name:</label><br>
  <input type="text" id="lname" name="lname" value="Doe"><br><br>
  <input type="submit" value="Submit">
</form>
```

The input disabled attribute

The disabled attribute specifies that an input field should be disabled (unusable and un-clickable):

First name:

Last name:

Size

- **Size:** spécifie la largeur visible, en caractères, d'un champ de saisie. Fonctionne avec les types d'entrée suivants: text, search, tel, url, email et password
- La valeur par défaut pour size est 20

```
<form>  
  <label for="fname">First name:</label><br>  
  <input type="text" id="fname" name="fname" size="50"><br>  
  <label for="pin">PIN:</label><br>  
  <input type="text" id="pin" name="pin" size="4">  
</form>
```

The input size attribute

The size attribute specifies the width (in characters) of an input field:

First name:

PIN:

Submit

Maxlength

- **Maxlength**: spécifie le nombre maximum de caractères autorisés dans un champ de saisie.

```
<input type="text" id="name" name="name" value="John">  
<input type="text" id="pin" name="pin" value="1234" maxlength="4" value="1234">  
</form>
```

The input maxlength attribute

The maxlength attribute specifies the maximum number of characters allowed in an input field:

First name:

PIN:

Submit

Min, Max

- spécifient les valeurs minimale et maximale d'un champ d'entrée.
- Les attributs min et max fonctionnent avec les types d'entrée suivants : nombre, plage, date, date-heure locale, mois, heure et semaine.

```
<input type="date" id="before" name="before" max="1980-01-01" />
<input type="date" id="after" name="after" min="2000-01-01" />
<input type="number" id="quantity" name="quantity" min="1" max="5" />
```

The input min and max attributes

The min and max attributes specify the minimum and maximum values for an input element.

Enter a date before 1980-01-01:

Enter a date after 2000-01-01:

Quantity (between 1 and 5):

Submit

Multiple

- L'attribut d'entrée multiples spécifie que l'utilisateur est autorisé à saisir plusieurs valeurs dans un champ de saisie.
- L'attribut Multiple fonctionne avec les types d'entrée suivants : e-mail et fichier.

Un champ de téléchargement de fichier qui accepte plusieurs valeurs :

```
<input type="file" id="files" multiple />  
<input type="file" id="files" name="files" multiple />
```

The input multiple attributes

The multiple attribute specifies that the user is allowed to enter more than one value in an input field.

Select files: Aucun fichier sélectionné.

To select multiple files, hold down the CTRL or SHIFT key while selecting.

Required

- spécifie qu'un champ de saisie doit être rempli avant de soumettre le formulaire.
- L'attribut required fonctionne avec les types d'entrée suivants : texte, recherche, URL, tél, e-mail, mot de passe, sélecteurs de date, numéro, case à cocher, radio et fichier.

```
<input type="text" id="username" name="username" required>
```

The input required attribute

The required attribute specifies that an input field must be filled out before submitting the form.

Username:

Note: The required attribute of the input tag is not supported in Safari prior version 10.1.

Step

- **Step:** spécifie l'intervalle légal de nombres pour un champ de saisie. Fonctionne avec les types input suivants : number, range, date, month, time and week.

Exemple : si step="3", les nombres pourraient être -3, 0, 3, 6,....etc.

```
<input type="number" id="points" min="0" step="3">
```

The input step attribute

The step attribute specifies the legal number intervals for an input element.

Points:

Exemple : affiche un champ de saisie numérique, dans lequel vous pouvez saisir une valeur de 0 à 100, par pas de 10.
La valeur par défaut est 30

```
<!DOCTYPE html>
<html>
<body>

<h2>Numeric Steps</h2>

<p>Depending on browser support: Fixed steps will apply in the input field.</p>

<form action="/action_page.php">
  <label for="quantity">Quantity:</label>
  <input type="number" id="quantity" name="quantity" min="0" max="100"
step="10" value="30">
  <input type="submit" value="Submit">
</form>

</body>
</html>
```

Numeric Steps

Depending on browser support: Fixed steps will apply in the input field.

Quantity:

Autofocus

- **Autofocus:** spécifie qu'un champ de saisie doit automatiquement obtenir le focus lors du chargement de la page.

```
<input type="text" id="first" name="first" autofocus>  
<input type="text" id="last" name="last" autofocus>  
<input type="text" id="first" name="first" autofocus>  
<input type="text" id="last" name="last" autofocus>
```

The input autofocus attribute

The autofocus attribute specifies that the input field should automatically get focus when the page loads.

First name:

Last name:

Pattern

- À voir dans la partie validation formulaire

Placeholder

- placeholder : spécifie un bref indice qui décrit la valeur attendue d'un champ d'entrée (un exemple de valeur ou une brève description du format attendu).
- Le petit conseil s'affiche dans le champ de saisie avant que l'utilisateur n'entre une valeur.
- L'attribut placeholder fonctionne avec les types d'entrée suivants : texte, recherche, URL, tél, e-mail et mot de passe.


```
<form>
  <label for="phone">Enter a phone number:</label>
  <input type="tel" id="phone" name="phone"
    placeholder="123-45-678"
    pattern="[0-9]{3}-[0-9]{2}-[0-9]{3}">
</form>
```

The input placeholder attribute

The placeholder attribute specifies a short hint that describes the expected value of an input field.

Enter a phone number:

Submit

Autocomplete

- Autocomplete : spécifie si un formulaire ou un champ de saisie doit avoir la saisie semi-automatique activée ou désactivée.
- Lorsqu'un utilisateur commence à taper dans un champ, le navigateur doit afficher des options pour remplir le champ, en fonction des valeurs saisies précédemment.
- L'attribut autocomplete fonctionne avec <form>les <input>types suivants : text, search, url, tel, email, password, range

Validation syntaxique des
formulaire

- Avant d'envoyer des données dans un formulaire, il faut s'assurer qu'elles sont dans un format correct pour pouvoir être traitées correctement.
- Exemple: si vous n'entrez pas les données dans le format attendu, vous aurez des messages similaires

- « Ce champ est obligatoire » (vous ne pouvez pas le laisser vide)
- « Veuillez entrer votre numéro de téléphone au format xxx-xxxx » (il attend trois chiffres suivis d'un tiret, suivi de quatre chiffres).
- « Veuillez entrer une adresse e-mail valide » (ce que vous avez saisi ne ressemble pas à une adresse e-mail valide).
- « Votre mot de passe doit comporter entre 8 et 30 caractères et contenir une majuscule, un symbole et un chiffre » (sérieusement ?).

Les types de validation de formulaire

1- validation coté client

- est effectuée dans le navigateur, avant que les données n'aient été soumises au serveur. Elle donne une réponse instantanée.
- Il est possible de la subdiviser encore avec:
 - - Validation Javascript.
 - - Fonctions de validation de formulaire HTML 5

2- Validation coté Serveur

- Après que les données soient soumises, le code serveur est utilisé pour valider les données avant de les mettre dans la base de données.
- Nécessite un aller-retour vers le serveur

- Dans le monde réel, les développeurs ont tendance à utiliser une combinaison de validations côté client et côté serveur, pour être du côté sûr.

Fonctions de validation HTML 5

Validation selon une expression régulière

- la syntaxe HTML5 permet la validation des données saisies dans un formulaire grâce à l'attribut **Pattern** qui attend une expression régulière comme valeur.
- L'attribut pattern peut être placé uniquement sur l'élément <input> ayant pour types text, search, password, url, tel, email.
- La valeur saisie doit respecter la forme imposée par l'expression régulière. Et si ce n'est pas le cas le formulaire est bloqué et les données ne sont pas envoyées vers le serveur.

Exemples

- 1- Code postal avec 5 chiffres entre 0-9:

```
<input pattern="[0-9]{5}" type="text" placeholder="ex : 37000"/>
```

Code postal ex : 37000

- 2- Valeur numérique d'au moins 5 chiffres précédée d'une lettre majuscule comprise entre A et F.

```
<input type="text" pattern="[A-F][0-9]{5}">
```

- Si le format n'est pas respecté, le navigateur en informe l'utilisateur.

Votre def Alpha (ex : A12345)



Veuillez respecter le format requis.

Quelques expressions régulières

- `[a-z]` : un caractère minuscule de la plage `[a-z]`
- `[A-Z]` : un caractère majuscule de la plage `[A-Z]`
- `[a-zA-Z]`: un caractère minuscule ou majuscule
- `A{5}`: correspond à `a`, 5 fois
- `a{5,7}` — correspond à `a`, 5 à 7 fois, mais ni plus, ni moins.
- `[0-9]` — correspond à tout nombre compris entre 0 et 9
- `[-]` — correspond à un espace ou un tiret.
- `[0-9]{3}[-][0-9]{3}[-][0-9]{4}`

- $[\text{^A-Z}]$: aucune lettre majuscule
- $(a \mid b)$: le a ou le b
- (ab) : le a et le b
-