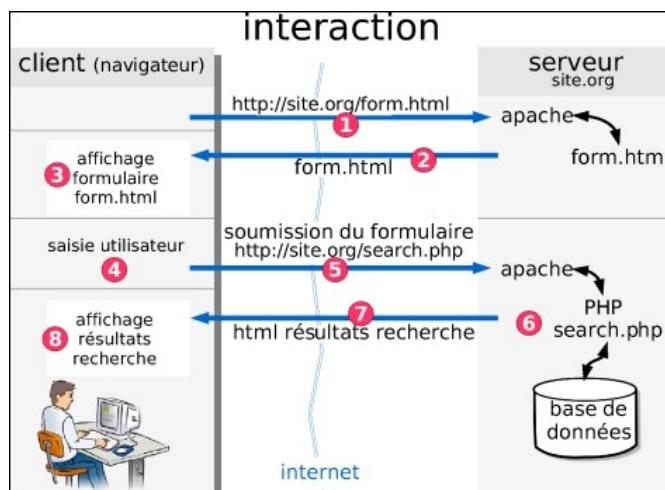


Les formulaires HTML

Schéma de fonctionnement



Syntaxe de base

```
<form>
    Login :
    <input type="text" name="login">
    <input type="submit" value="Envoyer">
</form>
```

type submit : 2 syntaxes

```
<input type="submit" value="Envoyer">

<button type="submit">Envoyer</button>
```

Attribut « action »

Attribut « action »

Il indique à quelle url la requête va être envoyée.

```
<form action="login.php">
```

Par défaut, c'est le même fichier qui est appelé.

Attribut «method»

Il précise comment sont envoyées les données du formulaire : **GET** ou **POST**.

Avec la méthode POST, les données sont transmises dans l'en-tête de la requête. Avec la méthode GET, les données sont transmises dans l'url.

Syntaxe :

```
<form method="get">  
ou  
<form method="post">
```

Attributs des champs de formulaire

name

L'attribut name est l'identifiant par lequel la données est envoyée

L'attribut name est obligatoire

value

Valeur par défaut.

placeholder

L'attribut placeholder est une chaîne de caractères fournissant une courte indication à l'utilisatrice ou l'utilisateur quant à l'information attendue dans le champ.

readonly

Lorsqu'il est présent, indique que le champ ne peut pas être édité par l'utilisatrice ou l'utilisateur

autofocus

Lorsqu'il est présent, indique que le champ reçoit le focus au chargement de la page.

Champs de formulaire

type "submit"

Il est l'élément par lequel le formulaire est envoyé. Il est possible de désactiver l'envoi d'un formulaire en ajoutant l'attribut « disabled ».

```
<input type="submit" value="Envoyer" disabled>
```

type "reset"

les input de type reset permettent de réinitialiser un formulaire

```
<input type="reset">
```

<input type="text">

Apparence : Zone de texte sur une seule ligne

```
<input type="text" name="login">
```

<textarea>

Apparence : Zone de texte sur plusieurs lignes

```
<textarea name="story"></textarea>
```

Personnalisations : il est possible de définir les dimensions avec **rows** et **cols** et d'ajouter un texte par défaut :

```
<textarea name="story" rows="5" cols="33">  
    It was a dark and stormy night...  
</textarea>
```

<input type="password">

Apparence : champs de saisi dont le contenu n'est pas lisible à l'écran.

<input type="checkbox">

Apparence : cases à cocher, pour sélectionner une ou plusieurs valeurs dans un formulaire.

Exemple avec une seule case :

```
<form>  
    <input type="checkbox" id="subscribe" name="subscription">  
    <label for="subscribe">Souhaitez-vous vous abonner à la newsletter ?</label>  
    <button type="submit">S'abonner</button>  
</form>
```

Attribut « **checked** » : affiche la case cochée par défaut.

Agrandir la zone de clic : balise « **label** » avec l'attribut « **for** » correspondant à l'id de la checkbox.

Lorsqu'une case est cochée, la valeur envoyée est la chaîne de caractère « **on** ».

Exemple avec plusieurs cases :

```
<form>
  <input type="checkbox" name="sport[]" value="foot"> Foot
  <input type="checkbox" name="sport[]" value="tennis"> Tennis
  <button type="submit">S'abonner</button>
</form>
```

Syntaxe particulière de l'attribut « name », avec **un couple de crochets**.

L'attribut « **value** » indique la valeur qui sera envoyée, lorsqu'une case est cochée.

```
<input type="radio">
```

Groupe d'options parmi lesquelles on ne peut choisir qu'une valeur.

```
<input type="radio" name="language" value="fr" checked>
<label for="language">Français</label>
<input type="radio" name="language" value="en">
<label for="language">Anglais</label>
```

L'attribut « **value** » indique la valeur qui sera envoyée.

Attribut « **checked** » : affiche la case cochée par défaut.

Agrandir la zone de clic : balise « **label** » avec l'attribut « **for** » correspondant à l'id de la checkbox.

```
<select>
```

Liste d'options parmi lesquelles l'utilisateur pourra choisir.

```
Moyen de transport :
<select name="transport">
  <option value="foot">Piéton</option>
  <option value="bike" selected>Vélo</option>
  <option value="train">Train</option>
</select>
```

Attribut « **multiple** » : possibilité de sélectionner plusieurs options parmi celles offertes dans le contrôle.

Attribut « **selected** » : fournit une valeur par défaut.

```
<input type="range">
```

Curseur

```
<label>Volume</label>
<input type="range" name="volume" min="0" max="10" step="0.1" >
```

<input type="hidden">

Champs invisible sur la page

```
<input name="product" type="hidden" value="xm234jq">
```

<input type="file">

Widget de sélection de fichiers.

```
<form action="upload.php" method="post" enctype="multipart/form-data">  
    Choisissez une photo de profil  
    <input type="file" name="avatar">  
    <input type="submit" value="Enregistrer">  
</form>
```

Attribut « **multiple** » : permet la sélection de plusieurs fichiers

Pour l'envoi de fichier, le formulaire doit être configuré avec la méthode « **post** » et l'attribut « **enctype="multipart/form-data"** ».

<input type="color">

Widget de sélection de couleur, dont l'apparence est nativement prise en charge par chaque OS.

```
<input name="favorite_color" type="color">
```

Attribut « **value** » permet de fournir une valeur par défaut (au format hexadécimal).

<input type="date">

Widget de sélection de date, dont l'apparence est nativement prise en charge par chaque OS.

```
<input name="partie" type="date">
```

Attribut « **value** » permet de fournir une valeur par défaut (au format yyyy-mm-dd).

<input type="time">

Widget de selection heure/minute, dont l'apparence est nativement prise en charge par chaque OS.

```
<input type="time" name="meeting">
```

Attribut « **value** » permet de fournir une valeur par défaut (au format hh:mm).

```
<input type="search">
```

Widget de saisi de recherche, dont l'apparence est nativement prise en charge par chaque OS.

```
<input type="search" name="q">
```

Attribut « **value** » permet de fournir une valeur par défaut

```
<input type="tel">
```

Widget de saisi de recherche ; les navigateurs mobiles proposent automatiquement un clavier adapté à la saisie de numéros de téléphone.

```
<input type="tel" name="phone">
```

Attribut « **value** » permet de fournir une valeur par défaut

Contraintes de validation

En HTML

Par type

<input type="email">	Vérifie que la valeur est bien un email.
<input type="url">	Vérifie que la valeur est bien une url commençant par http:// ou https://
<input type="number">	Vérifie que la valeur est bien numérique.

Par attributs

Attribut « **required** »

sur tous les input

Attribut « **min-length** » & « **max-length** »

sur les input de type text, password, ...

Attribut « **min** » & « **max** »

sur les input de type number, date, ...

Attribut « **accept** »

sur les input de type file

```
<input type="file" name="soundFile" accept="audio/*">
<input type="file" name="videoFile" accept="video/*">
<input type="file" name="imageFile" accept="image/*">
```

```
<input type="file" name="pdfFile" accept=".pdf">
```

Attribut « pattern »

L'attribut pattern permet de définir un « motif » que la saisie de l'utilisateur doit respecter. La syntaxe du motif est une « expression régulière »

Exemple :

La règle pour la saisie d'un code postal français valide : 5 caractères numériques :

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

La règle pour la saisie d'un prix valide : un ou plusieurs chiffres suivi d'une virgule puis une ou deux décimales

```
<input type="text" name="price" pattern="[0-9]+([,][0-9]{1,2})?">
```

En javascript

Nous verrons au cours des différentes séances que le langage Javascript nous donne des moyens pour lire les valeurs saisie dans le formulaire, les contrôler, effectuer la validation et d'autres traitements.

En fonction de nos besoins, le code Javascript pourra s'exécuter

- Au moment de l'envoi du formulaire
- Au moment de la saisie par l'utilisateur

Feedback CSS

Pour donner du feedback aux utilisateurs, penser à utiliser les pseudo-classes CSS :

:valid, :invalid, :required

Exemples :

```
:valid{  
    border-color:green ;  
}  
:invalid{  
    border-color:red;  
}  
:required {  
    background-color:pink ;  
}
```

Astuce pour afficher un astérisque rouge après le nom d'un champ obligatoire :

<https://jsfiddle.net/x5m4suf3/2/>