

HTML
CSS



Créez des formulaires


Toute page HTML peut être enrichie de formulaires interactifs, qui invitent vos visiteurs à renseigner des informations : saisir du texte, sélectionner des options, valider avec un bouton... tout est possible !

Créer un formulaire

Lorsqu'il vous prend subitement l'envie d'insérer un formulaire dans votre page HTML, vous devez pour commencer écrire une balise

`<form> </form>`

C'est la balise principale du formulaire, elle permet d'en indiquer le début et la fin.



```
1 <p>Texte avant le formulaire</p>
2
3 <form>
4   <p>Texte à l'intérieur du formulaire</p>
5 </form>
6
7 <p>Texte après le formulaire</p>
```

On va prendre un exemple pour que les choses soient claires. Supposons que votre visiteur vienne de taper un commentaire dans votre formulaire, par exemple un message qu'il aimerait publier sur vos forums. Ce message doit être *envoyé* pour que vous puissiez le recevoir (logique, non ?) et l'afficher pour vos autres visiteurs.

Eh bien c'est là le problème, ou plutôt les problèmes, que l'on va se poser :

Problème n° 1 : comment envoyer le texte saisi par le visiteur ? Par quel moyen ?

Problème n° 2 : une fois que les données ont été envoyées, comment les traiter ? Souhaitez-vous recevoir le message automatiquement par mail ou préférez-vous qu'un programme se charge de l'enregistrer quelque part, puis de l'afficher sur une page visible par tout le monde ?

Pour fournir les réponses à ces deux problèmes, vous devez ajouter deux attributs à la balise `<form>`



Method cet attribut indique par quel moyen les données vont être envoyées (réponse au **problème n° 1**). Il existe deux solutions pour envoyer des données sur le Web :

method=« get » :

: c'est une méthode en général assez peu adaptée car elle est limitée à 255 caractères. La particularité vient du fait que les informations seront envoyées dans l'adresse de la page (http://...), mais ce détail ne nous intéresse pas vraiment pour le moment. La plupart du temps, je vous recommande d'utiliser l'autre méthode : post

method="post" : c'est la méthode la plus utilisée pour les formulaires car elle permet d'envoyer un grand nombre d'informations. Les données saisies dans le formulaire ne transitent pas par la barre d'adresse ;

Action : c'est l'adresse de la page ou du programme qui va *traiter* les informations (réponse au **problème n° 2**). Cette page se chargera de vous envoyer un e-mail avec le message si c'est ce que vous voulez, ou bien d'enregistrer le message avec tous les autres dans une base de données.

Cela ne peut pas se faire en HTML et CSS ; on utilisera en général un autre langage dont vous avez peut-être entendu parler : PHP.



On va donc maintenant compléter la balise `<form>` avec les deux attributs qu'on vient de voir.

Pour `method` , vous l'aurez deviné, je vais mettre la valeur `post` .

Pour `action` , je vais taper le nom d'une page fictive en PHP (`traitement.php`). C'est cette page qui sera appelée lorsque le visiteur cliquera sur le bouton d'envoi du formulaire.

```
1 <p>Texte avant le formulaire</p>
2
3 <form method="post" action="traitement.php">
4   <p>Texte à l'intérieur du formulaire</p>
5 </form>
6
7 <p>Texte après le formulaire</p>
```

Pour le moment, on ne sait pas ce qu'il se passe à l'intérieur de la

page **traitement.php** : je vous demande de me faire confiance et d'imaginer que cette page existe et fonctionne.

Notre priorité, pour le moment, est de découvrir en HTML/CSS comment faire pour insérer des zones de texte, des boutons et des cases à cocher dans votre page web. C'est ce que nous allons voir maintenant.

Les zones de saisie basiques

Nous allons passer en revue les différentes balises HTML permettant de saisir du texte dans un formulaire.

Il faut savoir qu'il y a deux zones de texte différentes :

- **la zone de texte monoligne** : comme son nom l'indique, on ne peut y écrire qu'une seule ligne.

Elle sert à saisir des textes courts, par exemple un pseudo ;

- **la zone de texte multiligne** : cette zone de texte permet d'écrire une quantité importante de texte sur plusieurs lignes, par exemple une dissertation sur l'utilité du HTML dans le développement des pays d'Asie du Sud-Est (ce n'est qu'une suggestion, hein...).



Zone de texte monoligne

La figure suivante montre à quoi ressemble une zone de texte monoligne.

Votre pseudo :

Pour insérer une zone de texte dans une ligne, on va utiliser la balise

`<input />`

On retrouvera cette balise plusieurs fois dans la suite de ce chapitre. À chaque fois, c'est la valeur de son attribut **`type`** qui va changer.

```
1 <input type="text" />
```


Ce n'est pas encore suffisant : il faut donner un nom à votre zone de texte. Ce nom n'apparaît pas sur la page, mais il vous sera indispensable par la suite. En effet, cela vous permettra (en PHP par exemple) de reconnaître d'où viennent les informations : vous saurez que tel texte est le pseudo du visiteur, tel texte est le mot de passe qu'il a choisi, etc.

Pour donner un nom à un élément de formulaire, on utilise l'attribut **name** . Ici, on va supposer qu'on demande au visiteur de rentrer son pseudo

```
1 <input type="text" name="pseudo" />
```

Essayons donc de créer un formulaire très basique avec notre champ de texte :

```
1 <form method="post" action="traitement.php">
2   <p><input type="text" name="pseudo" /></p>
3 </form>
```

Comme d'habitude, je vous invite fortement à essayer ce code chez vous afin d'observer le résultat.

Les libellés

Cette zone de texte est bien jolie mais si votre visiteur tombe dessus, il ne sait pas ce qu'il doit écrire.

C'est justement le rôle de la balise `<label>` :

```
1 <form method="post" action="traitement.php">
2   <p>
3     <label>Votre pseudo</label> : <input type="text" name="pseudo" />
4   </p>
5 </form>
```

Mais cela ne suffit pas. Il faut lier le label à la zone de texte.

Pour ce faire, on doit donner un nom à la zone de texte, non pas avec l'attribut `name` mais avec l'attribut `id` (que l'on peut utiliser sur toutes les balises).

Pour lier le label au champ, il faut lui donner un attribut `for` qui a la même valeur que l' `id` du champ... Le mieux est de le voir sur un exemple :

```
1 <form method="post" action="traitement.php">
2   <p>
3     <label for="pseudo">Votre pseudo</label> : <input type="text" name="pseudo" id="pseudo" />
4   </p>
5 </form>
```

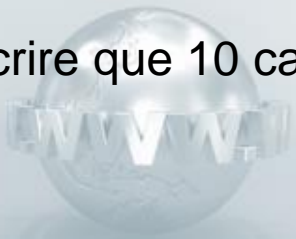
Essayez de cliquer sur le texte « Votre pseudo » : vous allez voir que le curseur se place automatiquement dans la zone de texte correspondante.



On peut ajouter un certain nombre d'autres attributs à la balise `<input />` pour **personnaliser son fonctionnement** :

- on peut agrandir le champ avec `size` ;
- on peut limiter le nombre de caractères que l'on peut saisir avec `maxlength` ;
- on peut préremplir le champ avec une valeur par défaut à l'aide de `value` ;
- on peut donner une indication sur le contenu du champ avec `placeholder`. Cette indication disparaîtra dès que le visiteur aura cliqué à l'intérieur du champ.

Dans l'exemple suivant, la zone de texte contient une indication permettant de comprendre ce qu'il faut saisir ; le champ fait 30 caractères de long mais on ne peut écrire que 10 caractères maximum à l'intérieur :



```
1 <form method="post" action="traitement.php">
2     <p>
3         <label for="pseudo">Votre pseudo :</label>
4         <input type="text" name="pseudo" id="pseudo" placeholder="Ex : Zozor" size="30"
5         maxlength="10" />
6     </p>
7 </form>
```

Votre pseudo :

Zone de mot de passe

Vous pouvez facilement faire en sorte que la zone de texte se comporte comme une « zone de mot de passe », c'est-à-dire une zone où on ne voit pas à l'écran les caractères saisis. Pour créer ce type de zone de saisie, utilisez l'attribut `type="password"` .

Je complète mon formulaire. Il demande maintenant au visiteur son pseudo *et* son mot de passe :



Votre pseudo :

Votre mot de passe :

Zone de texte multiligne

Pour créer une zone de texte multiligne, on change de balise : nous allons utiliser **`<textarea>`** **`</textarea>`**

Comme pour tout autre élément du formulaire, il faut lui donner un nom avec `name` et utiliser un `label` qui explique de quoi il s'agit

```
1 <form method="post" action="traitement.php">
2   <p>
3     <label for="ameliorer">Comment pensez-vous que je pourrais améliorer mon site ?</label><br
4     <textarea name="ameliorer" id="ameliorer"></textarea>
5   </p>
6 </form>
```

Comment pensez-vous que je pourrais améliorer mon site ?

- **En CSS** : il suffit d'appliquer les propriétés CSS `width` et `height` au `<textarea>` .
- **Avec des attributs** : on peut ajouter les attributs `rows` et `cols` à la balise `<textarea>` . Le premier indique le nombre de lignes de texte qui peuvent être affichées simultanément, et le second le nombre de colonnes.

```
1 <form method="post" action="traitement.php">
2   <p>
3     <label for="ameliorer">
4       Comment pensez-vous que je puisse améliorer mon site ?
5     </label>
6     <br />
7
8     <textarea name="ameliorer" id="ameliorer" rows="10" cols="50">
9       Améliorer ton site ?!
10      Mais enfin ! Il est tellement génialissime qu'il n'est pas nécessaire de l'améliorer !
11    </textarea>
12  </p>
13 </form>
```

Comment pensez-vous que je puisse améliorer mon site ?


Améliorer ton site ?!
Mais enfin ! Il est tellement génialissime qu'il
n'est pas nécessaire de l'améliorer !

Les zones de saisie enrichies

E-mail

Vous pouvez demander à saisir une adresse e-mail :

```
1 <input type="email" />
```




dsqfdsdfsp

Une URL

Avec le type `url` , on peut demander à saisir une adresse absolue (commençant généralement par `http://`)

```
1 <input type="url" />
```

Numéro de téléphone



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



```
1 <input type="number" />
```



Vous pouvez personnaliser le fonctionnement du champ avec les attributs suivants :

- **min** : valeur minimale autorisée
- **max** : valeur maximale autorisée
- **step** : c'est le « pas » de déplacement. Si vous indiquez un pas de 2, le champ n'acceptera que des valeurs de 2 en 2 (par exemple 0, 2, 4, 6...).

Un curseur

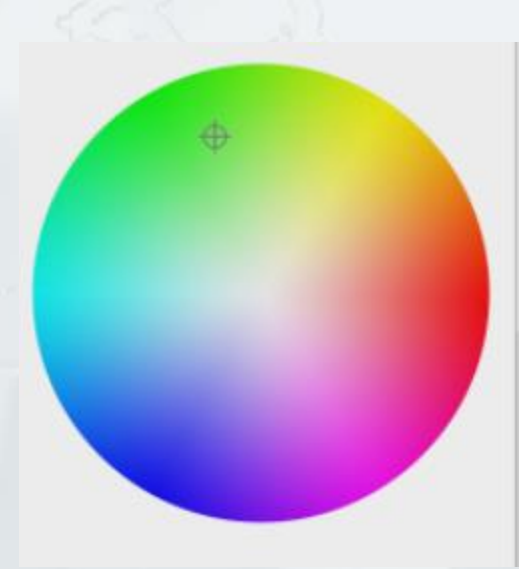
```
1 <input type="range" />
```



Vous pouvez utiliser là aussi les attributs `min` , `max` et `step` pour restreindre les valeurs disponibles.

Couleur

```
1 <input type="color" />
```

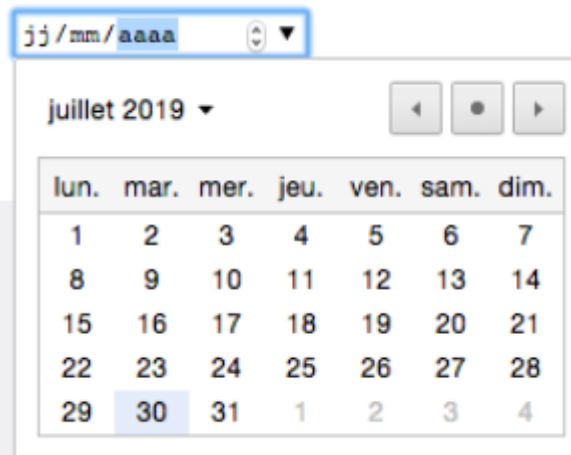


Date

Différents types de champs de sélection de date existent :

- `date` : pour la date (05/08/1985 par exemple) ;
- `time` : pour l'heure (13:37 par exemple) ;
- `week` : pour la semaine ;
- `month` : pour le mois ;
- `datetime` : pour la date et l'heure (avec gestion du décalage horaire) ;
- `datetime-local` pour la date et l'heure (sans gestion du décalage horaire).

```
1 <input type="date" />
```



The image shows a date picker interface. At the top, there is a text input field containing "jj/mm/aaaa" with a dropdown arrow. Below it, the text "juillet 2019" is displayed with a dropdown arrow. To the right of the text are three navigation buttons: a left arrow, a central dot, and a right arrow. Below these is a calendar grid with days of the week as headers: "lun.", "mar.", "mer.", "jeu.", "ven.", "sam.", "dim.". The grid contains numbers 1 through 31. The number 30 is highlighted in blue.

lun.	mar.	mer.	jeu.	ven.	sam.	dim.
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

Recherche

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



Les éléments d'options

HTML vous offre une ribambelle d'éléments d'options à utiliser dans votre formulaire. Ce sont des éléments qui demandent au visiteur de faire un choix parmi une liste de possibilités. Nous allons passer en revue :

- les cases à cocher ;
- les zones d'options ;
- les listes déroulantes.

Les cases à cocher

```
1 <input type="checkbox" name="choix" />
```



```
1 <form method="post" action="traitement.php">
2   <p>
3     Cochez les aliments que vous aimez manger :<br />
4     <input type="checkbox" name="frites" id="frites" /> <label for="frites">Frites</label><br
5     />
6     <input type="checkbox" name="steak" id="steak" /> <label for="steak">Steak haché</label><br
7     />
8     <input type="checkbox" name="epinards" id="epinards" /> <label
9     for="epinards">Epinards</label><br />
10    <input type="checkbox" name="huitres" id="huitres" /> <label for="huitres">Huitres</label>
11  </p>
12 </form>
```

Cochez les aliments que vous aimez manger :

- ☒ Frites
- ☒ Steak haché
- ☐ Epinards
- ☐ Huitres



Enfin, sachez que vous pouvez faire en sorte qu'une case soit cochée par défaut, avec l'attribut checked

```
1 <input type="checkbox" name="choix" checked />
```

Les zones d'options

```
1 <form method="post" action="traitement.php">
2   <p>
3     Veuillez indiquer la tranche d'âge dans laquelle vous vous situez :<br />
4     <input type="radio" name="age" value="moins15" id="moins15" /> <label for="moins15">Moins
de 15 ans</label><br />
5     <input type="radio" name="age" value="medium15-25" id="medium15-25" /> <label
for="medium15-25">15-25 ans</label><br />
6     <input type="radio" name="age" value="medium25-40" id="medium25-40" /> <label
for="medium25-40">25-40 ans</label><br />
7     <input type="radio" name="age" value="plus40" id="plus40" /> <label for="plus40">Encore
plus vieux que ça ?!</label>
8   </p>
9 </form>
```

Veuillez indiquer la tranche d'âge dans laquelle vous vous situez :

- ☐ Moins de 15 ans
- ☐ 15-25 ans
- ☒ 25-40 ans
- ☐ Encore plus vieux que ça ?!



si vous avez deux zones d'options différentes, il faut donner un name unique à chaque groupe, comme ceci :

```
1 <form method="post" action="traitement.php">
2   <p>
3     Veuillez indiquer la tranche d'âge dans laquelle vous vous situez :<br />
4     <input type="radio" name="age" value="moins15" id="moins15" /> <label for="moins15">Moins
de 15 ans</label><br />
5     <input type="radio" name="age" value="medium15-25" id="medium15-25" /> <label
for="medium15-25">15-25 ans</label><br />
6     <input type="radio" name="age" value="medium25-40" id="medium25-40" /> <label
for="medium25-40">25-40 ans</label><br />
7     <input type="radio" name="age" value="plus40" id="plus40" /> <label for="plus40">Encore
plus vieux que ça ?!</label>
8   </p>
9   <p>
10    Sur quel continent habitez-vous ?<br />
11    <input type="radio" name="continent" value="europe" id="europe" /> <label
for="europe">Europe</label><br />
12    <input type="radio" name="continent" value="afrique" id="afrique" /> <label
for="afrique">Afrique</label><br />
13    <input type="radio" name="continent" value="asie" id="asie" /> <label
for="asie">Asie</label><br />
14    <input type="radio" name="continent" value="amerique" id="amerique" /> <label
for="amerique">Amérique</label><br />
15    <input type="radio" name="continent" value="australie" id="australie" /> <label
for="australie">Australie</label>
16  </p>
17 </form>
```

Les listes déroulantes

es listes déroulantes sont un autre moyen élégant de faire un choix parmi plusieurs possibilités. Le fonctionnement est un peu différent. On va utiliser la balise **<select>** **</select>** qui indique le début et la fin de la liste déroulante. On ajoute l'attribut `name` à la balise pour donner un nom à la liste.

Puis, à l'intérieur du **<select>** **</select>** , nous allons placer plusieurs balises `<option>` `</option>` (une par choix possible). On ajoute à chacune d'elles un attribut `value` pour pouvoir identifier ce que le visiteur a choisi.

Voici un exemple d'utilisation :




```
1 <form method="post" action="traitement.php">
2   <p>
3     <label for="pays">Dans quel pays habitez-vous ?</label><br />
4     <select name="pays" id="pays">
5       <option value="france">France</option>
6       <option value="espagne">Espagne</option>
7       <option value="italie">Italie</option>
8       <option value="royaume-uni">Royaume-Uni</option>
9       <option value="canada">Canada</option>
10      <option value="etats-unis">États-Unis</option>
11      <option value="chine">Chine</option>
12      <option value="japon">Japon</option>
13    </select>
14  </p>
15 </form>
```



Dans quel pays habitez-vous ?

Espagne	▼
Europe	
France	
Espagne	
Italie	
Royaume-Uni	
Amérique	
Canada	
Etats-Unis	
Asie	
Chine	
Japon	

Nous y sommes presque. Il ne nous reste plus qu'à agrémenter notre formulaire de quelques dernières fonctionnalités (comme la validation), puis nous pourrons ajouter le bouton d'envoi du formulaire.

Regrouper les champs

Si votre formulaire grossit et comporte beaucoup de champs, il peut être utile de les regrouper au sein de plusieurs balises `<fieldset>`. Chaque `<fieldset>` peut contenir une légende avec la balise `<legend>`. Regardez cet exemple :



Vos coordonnées

Quel est votre nom ?

Quel est votre prénom ?

Quel est votre e-mail ?

Votre souhait

Faites un souhait que vous voudriez voir exaucé :

- ☐ Etre riche
- ☐ Etre célèbre
- ☐ Etre **encore** plus intelligent
- ☐ Autre...

Les champs sont regroupés



```
1 <form method="post" action="traitement.php">
2
3   <fieldset>
4     <legend>Vos coordonnées</legend> <!-- Titre du fieldset -->
5
6     <label for="nom">Quel est votre nom ?</label>
7     <input type="text" name="nom" id="nom" />
8
9     <label for="prenom">Quel est votre prénom ?</label>
10    <input type="text" name="prenom" id="prenom" />
11
12    <label for="email">Quel est votre e-mail ?</label>
13    <input type="email" name="email" id="email" />
14
15  </fieldset>
16
17  <fieldset>
18    <legend>Votre souhait</legend> <!-- Titre du fieldset -->
19
20    <p>
21      Faites un souhait que vous voudriez voir exaucé :
22
23      <input type="radio" name="souhait" value="riche" id="riche" /> <label for="riche">Etre
riche</label>
24      <input type="radio" name="souhait" value="celebre" id="celebre" /> <label
for="celebre">Etre célèbre</label>
25      <input type="radio" name="souhait" value="intelligent" id="intelligent" /> <label
for="intelligent">Etre <strong>encore</strong> plus intelligent</label>
26      <input type="radio" name="souhait" value="autre" id="autre" /> <label for="autre">Autre...
</label>
27    </p>
28
29    <p>
30      <label for="precisions">Si "Autre", veuillez préciser :</label>
31      <textarea name="precisions" id="precisions" cols="40" rows="4"></textarea>
32    </p>
33  </fieldset>
34 </form>
```

Sélectionner automatiquement un champ

Vous pouvez placer automatiquement le curseur dans l'un des champs de votre formulaire avec l'attribut **autofocus** . Dès que le visiteur chargera la page, le curseur se placera dans ce champ.

```
1 <input type="text" name="prenom" id="prenom" autofocus />
```

Rendre un champ obligatoire

Vous pouvez faire en sorte qu'un champ soit obligatoire en lui donnant l'attribut `required`.

```
1 <input type="text" name="prenom" id="prenom" required />
```



```
1 :required
2 {
3     background-color: red;
4 }
```

Le bouton d'envoi

Il ne nous reste plus qu'à créer le bouton d'envoi. Là encore, la balise `<input />` vient à notre secours. Elle existe en quatre versions :

- `type="submit"` : le principal bouton d'envoi de formulaire. C'est celui que vous utiliserez le plus souvent. Le visiteur sera conduit à la page indiquée dans l'attribut `action` du formulaire
- `type="reset"` : remise à zéro du formulaire
- `type="image"` : équivalent du bouton submit , présenté cette fois sous forme d'image. Rajoutez l'attribut `src` pour indiquer l'URL de l'image
- `type="button"` : bouton générique, qui n'aura (par défaut) aucun effet. En général, ce bouton est géré en JavaScript pour exécuter des actions sur la page.



Envoyer

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

Lorsque vous cliquez sur le bouton « Envoyer », le formulaire vous amène alors à la page indiquée dans l'attribut `action` . Souvenez-vous, nous avons imaginé une page fictive : `traitement.php` .



1.Exercice 01

Créer un formulaire HTML5 avec les éléments suivants :

- Un fieldset avec legend "coordonnées"
- Champ de texte avec l'attribut placeholder
- Champ pour renseigner une URL
- Zone de texte pour renseigner une adresse "textarea"
- Champ pour renseigner une adresse eMail
- Champ réclamant un format numérique
- Champ de date avec ouverture d'un calendrier
- Champ de couleur en code hexadécimal
- Champ de téléchargement d'un fichier
- Un bouton qui sert à envoyer votre formulaire
- Devant chaque champs mettez un texte de renseignement ("label")
- Pour aligner tous les éléments du formulaire. Utilisez un tableau HTML.

1.Exercice 2

- Écrire une page HTML5 avec titre Création des formulaires
- Cette page contient un formulaire avec les contraintes suivantes :
 - ☐ Utilisation de la méthode Get
 - ☐ Le nom de la page cible est défini grâce à l'attribut action.
 - ☐ présente une zone de saisie pour un identifiant (input de type text qui aura automatiquement le focus dès l'ouverture de la page),
 - ☐ présente une zone de saisie pour un mot de passe.
 - ☐ présente un champ caché nommé operations .
 - ☐ présente un champ numéro de téléphone qui commence obligatoirement avec le chiffre 9 et n'accepte les chiffre de zéro à 9 pour le faire consulter [Saisie de numéro de téléphone en HTML](#).
 - ☐ présente un bouton pour télécharger les fichiers,
 - ☐ présente un bouton d'envoi.

1.Exercice 3

- À l'aide des éléments et attributs HTML suivants:
 - `fieldset`, `date`, `radio`, `select`, `textarea`, `url`, `label`, `legend`, `mail`, `text`, `submit`, `label`, ...
- Reproduire le formulaire suivant:

Vos coordonnées

Nom :	<input type="text" value="votre nom"/>
Prénom :	<input type="text" value="votre prénom"/>
Date de naissance :	<input type="text" value="jj/mm/aaaa"/>
Mail :	<input type="text" value="votre mail"/>
Votre site :	<input type="text"/>
Code :	<input type="text"/>
Genre :	<input type="radio"/> Homme <input type="radio"/> Femme
Pays :	<input type="text" value="Votre pays"/>

Vos goûts

☐ Tarte aux fraises à la crème d'amandes

☐ Tarte aux prunes

☐ Tarte rustique aux mirabelles de Lorraine, noisettes et miel

Décrivez vos goûts en détail

Envoyez nous votre photo

Aucun fichier choisi

1.Exercice 4

- À l'aide des éléments et attributs HTML
- Reproduire l'apparence de la page (ex04-Formulaire.html) de la façon la plus fidèle possible.

Coordonnées

Civilité

☐ Mademoiselle ☐ Madame ☒ Monsieur

Nom / Prénom

Adresse

code postal/
Localité /
Gouvernorat

▼

Pays

▼

Plateforme(s)

☒ Windows ☒ Macintosh ☐ Unix

Applications(s)

Bureautique ▲
DAO
Statistiques
SGBD
Internet ▼

1.Exercice 5

Formulaire d'inscription

Nom

Prenom

Email

Sexe Masculin ☒ Féminin ☐

Photo Aucun fichier choisi

Pays ▼

Les langages préférés

Java ▲

C

C++

PHP ▼

 (Pour faire plusieurs choix maintenir la touche CTRL enfoncée)

Domaines d'activités ☒ Informatique ☐ Gestion ☐ Pédagogie