

HTML Les formulaires



ADRAR DIGIT@L ACADEMY

PÔLE NUMERIQUE DU CENTRE DE FORMATION ADRAR
> SUPPORT, ADMINISTRATION SYSTEMES & RESEAUX
> DEVELOPPEMENT D'APPLICATIONS WEB & MOBILES
> TRANSFORMATION NUMERIQUE DES ENTREPRISES

<http://www.adrar-numerique.com>

HTML Les formulaires: Introduction

Sur les différents sites internet que vous avez pu voir, vous avez sûrement vu des dizaines de formulaires !

Ils permettent aux visiteurs de renseigner des champs pour que, nous développeurs, puissions récupérer ces informations dans divers buts.

Ici, nous ne verrons pas comment récupérer les informations, mais juste la manière de les envoyer.

CONTACT & INSCRIPTION

Pour vous inscrire (*) et participer aux évaluations d'entrée en formation, avoir plus d'informations, nous poser vos questions, prendre rdv, etc... c'est très simple, remplissez le formulaire ci-dessous, joignez un CV et vous serez très rapidement contacté par téléphone ou mail. (*) toute inscription doit être préalablement validée et enregistrée par l'ADRAR avant que vous vous présentiez...

CONTACTEZ-NOUS / INSCRIVEZ-VOUS / REJOIGNEZ-NOUS

ces formations métiers sont accessibles à tous, sans prérequis de diplôme, pour peu que vous soyez motivés, et déterminés à réussir l'expérience

Sélectionnez dans la liste ci-dessous la formation et le métier envisagé :

Sélectionnez ici la formation... ▾

Sélectionnez dans la liste ci-dessous le site de formation envisagé :

Sélectionnez le site de formation ▾

VOTRE MESSAGE

Pour postuler, vous inscrire/participer aux évaluations d'entrée en formation, merci de nous communiquer ci-dessous votre C.V. actualisé.

Aucun fichier choisi

(Taille maxi : 2Mo Formats : pdf|doc|docx|odt)

☐ Je ne suis pas un robot

Formulaire de l'ADRAR

HTML Les formulaires: La base

- Pour faire un formulaire, nous aurons besoin de la balise `<form></form>` :

```
<form>  
    <!--Ici mon formulaire-->  
</form>
```

- Il nous faudra aussi préciser dans cette balise, à l'aide d'attributs, la méthode d'envoi et l'adresse du programme qui doit traiter notre demande.

HTML Les formulaires: Méthodes & Actions

Comme dit précédemment, il faudra ajouter une méthode d'envoi, ici deux valeurs possibles pour notre attribut « method » :

1. Méthode « get » => c'est une méthode qui écrit nos informations dans l'URL de notre page. Elle est limitée à 255 caractères.

```
http://www.monSuperSite.com/contact.html?nom=Roger&mdp=pasDuToutSécu
```

2. Méthode « post » => c'est la plus utilisée pour les formulaires, elle permet d'envoyer plus d'informations, de manière un peu moins visible.

ATTENTION !

Les informations envoyées avec la méthode « post » ne sont pas non plus sécurisée !
Elle sont uniquement **MOINS** visible.

HTML Les formulaires: Méthodes & Actions

L'attribut « action » quant à lui, permet de choisir l'adresse à qui envoyer les données (une page HTML, ou un programme, par exemple, en PHP)

Voici ce que ça donne :

```
<form method="POST" action="monProgramme.php">
  <!--Ici mon formulaire-->
</form>
```

HTML Les formulaires: Input Text

LES FORMULAIRES: INPUT TEXT

Commençons avec la balise `<input>` qui contiendra l'attribut « type » qui déterminera.... Son type.

Le premier type qui nous intéressera sera le texte :

```
<form method="POST" action="monProgramme.php">  
  <input type="text"/>  
</form>
```

HTML Les formulaires: Input Text

Ajoutons quelques attributs à notre balise `<input>` pour améliorer tout ça:

- L'attribut « name » => Permet de nommer la valeur du champ (la variable)
- L'attribut « value » => Permet d'imposer une valeur par défaut
- L'attribut « placeholder » => Permet une indication sur le contenu du champ, puis disparaît quand on écrit
- L'attribut « maxlength » => Permet de limiter le nombre de caractères
- L'attribut « minlength » => Permet d'imposer un nombre minimum de caractères
- L'attribut « size » => Permet de choisir la taille du champ
- L'attribut « id » => Vous connaissez déjà ! (permet d'identifier le champ)
- L'attribut « required » => Permet de rendre un champ obligatoire (sans valeur)
- L'attribut « autofocus » => Permet de forcer l'emplacement du curseur

```
<form method="POST" action="monProgramme.php">  
  <input type="text" name="prenom" placeholder="Entrez un prénom" maxlength="20" id="prenom"/>  
</form>
```

HTML Les formulaires: Label

Avant de continuer sur les `<input>`, penchons-nous rapidement sur la balise `<label></label>` qui permet de rajouter un libellé à nos `<input>`

Pour cela, il faut lier notre libellé à notre `<input>` grâce à l'attribut « for » où l'on indique la valeur de l'identifiant de notre `<input>`:

```
<form method="POST" action="monProgramme.php">
  <label for="prenom">Prénom:</label>
  <input type="text" name="prenom" placeholder="Entrez un prénom" maxlength="20" id="prenom"/>
</form>
```

Prénom:

HTML Les formulaires: Input Text Variantes

Il existe plusieurs variantes de texte pour notre balise. Ces types alternatifs ne changent pas l'apparence en mode desktop (PC) mais changent le clavier des mobiles :

- email => Permet de saisir une adresse Email

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

- tel => Permet de saisir un numéro de téléphone

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

- url => Permet de saisir une URL

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

HTML Les formulaires: Mise en pratique (25 min)

- Dans un nouveau répertoire, créez un fichier « formulaire.html » avec le HTML de base.
- Créez un formulaire qui contiendra:
 1. Un champ pour le prénom, limité à 20 caractères, avec au moins 2 caractères minimum et un placeholder
 2. Un champ pour le nom, limité à 20 caractères, avec au moins 2 caractères minimum et un placeholder
 3. Un champ pour l'email, avec un placeholder
 4. Un champ pour le numéro de téléphone, avec un placeholder

Prénom:	<input type="text" value="Ex: Jacques"/>
Nom:	<input type="text" value="Ex: Dubois"/>
E-mail	<input type="text" value="Ex: monMail@FAI.com"/>
Numéro	<input type="text" value="Ex: 00.11.22.33.44"/>

HTML Les formulaires: Input Password

Celle-ci, vous la connaissez bien, elle permet d'ajouter un champ pour un mot de passe, en remplaçant les caractères par des points :

```
<form method="POST" action="monProgramme.php">  
  <input type="password" placeholder="Ex: P@ssWo4d" />  
</form>
```

Note: Il change aussi le clavier sur mobile, pour ne pas mémoriser le mot de passe avec un clavier sécurisé

HTML Les formulaires: Input Number

Ici, cela nous permet de choisir un nombre. Il ajoute des petites flèches par défaut sur plusieurs navigateurs pour permettre d'incrémenter ou de décrémenter la valeur :

```
<form method="POST" action="monProgramme.php">
  <input type="number" placeholder="Ex: 15"/>
</form>
```


HTML Les formulaires: TextArea

Cette nouvelle balise permet d'ajouter un champ de texte, mais cette fois sur plusieurs lignes :

```
<form method="POST" action="monProgramme.php">
  <textarea placeholder="Un très long texte"></textarea>
</form>
```

Cette zone de texte est modifiable par l'utilisateur à l'aide des petits traits en bas à droite.

Nous pouvons aussi ajouter les attributs « cols » et « rows » (respectivement colonnes et lignes) pour spécifier une taille de départ.

Il est aussi possible de mettre du texte entre les deux balises `<textarea>` pour mettre du contenu par défaut (identique à l'attribut « value »).

HTML Les formulaires: Mise en pratique (25 min)

- Reprenez le précédent formulaire
- Ajoutez-y les champs suivants:
 1. Un champ pour le mot de passe, avec un minimum de 8 caractères et un maximum de 30. Il doit contenir un placeholder.
 2. Un champ pour l'âge, avec un placeholder.
 3. Un champ pour laisser un commentaire, avec un placeholder, 70 de longueur et 10 de hauteur.

Prénom:
 Nom:
 E-mail:
 Numéro:
 Mot de passe:
 Age:
 Laissez un commentaire:

HTML Les formulaires: Input date

Cet `<input>` permet à l'utilisateur de saisir une date dans un champ spécifié. L'apparence dépendra du navigateur utilisé :

```
<form method="POST" action="monProgramme.php">  
  <input type="date"/>  
</form>
```

jj/mm/aaaa

18/10/2030

octobre 2030

lu	ma	me	je	ve	sa	di
30	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

Aujourd'hui

Il y a là aussi plusieurs variations possibles :

- `date` => Pour la date
- `time` => Pour l'heure
- `datetime` => Pour la date et l'heure (avec décalage horaire) **/!\ Déprécié /!**
- `datetime-local` => Pour la date et l'heure (sans décalage horaire)
- `month` => Pour le mois
- `week` => Pour la semaine

Il est aussi possible de spécifier un intervalle avec les attributs « `min` » et « `max` » sous le format américain « `aaaa-mm-jj` ».

HTML Les formulaires: Input File

Cet `<input>` permet à l'utilisateur d'*uploader* (téléverser en français) un fichier :

```
<form method="POST" action="monProgramme.php">
  <input type="file"/>
</form>
```

Choisir un fichier Aucun fichier choisi

Nous pouvons spécifier le type de fichier attendu avec l'attribut « accept » :

```
<form method="POST" action="monProgramme.php">
  <input type="file" accept=".png, .jpg, .gif"/>
</form>
```

Nous pouvons aussi autoriser l'envoi de plusieurs fichiers avec l'attribut « multiple », sans valeur.

HTML Les formulaires: Input Checkbox

Les « checkbox », ou case à cocher, sont des éléments qui permettent des choix multiples :

```
<form method="POST" action="monProgramme.php">
  <input type="checkbox" id="checkbox"/><label for="checkbox">Je suis une case à cocher</label>
  <br/>
  <input type="checkbox" id="checkbox2"/><label for="checkbox2">Je suis une checkbox</label>
</form>
```

☐ Je suis une case à cocher
☒ Je suis une checkbox

Vous pouvez ajouter l'attribut « checked », sans valeur, pour cocher une case par défaut.

HTML Les formulaires: Fieldset & Legend

Dans un formulaire, il est possible de délimiter et de regrouper les balises grâce à un `<fieldset></fieldset>`.

Il est aussi possible de mettre une légende à notre zone, grâce à la balise `<legend></legend>` :

```
<fieldset>
  <legend>Je suis une légende</legend>
  <div>
    <label for="champ1">Champ1:</label>
    <input type="text" id="champ1" placeholder="Champ1"/>
  </div>
  <div>
    <label for="champ2">Champ2:</label>
    <input type="text" id="champ2" placeholder="Champ2"/>
  </div>
  <div>
    <label for="champ3">Champ3:</label>
    <input type="text" id="champ3" placeholder="Champ3"/>
  </div>
</fieldset>
```

Champ1:	<input type="text" value="Champ1"/>
Champ2:	<input type="text" value="Champ2"/>
Champ3:	<input type="text" value="Champ3"/>

Je suis une légende	
Champ1:	<input type="text" value="Champ1"/>
Champ2:	<input type="text" value="Champ2"/>
Champ3:	<input type="text" value="Champ3"/>

HTML Les formulaires: Mise en pratique (25 min)

- Reprenez le précédent formulaire.
- Ajoutez y les champs suivants :
 1. Un champ pour la date du commentaire.
 2. Un bouton pour l'envoi de fichier image (.jpg, .png, .gif).
 3. Un ensemble de quatre cases à cocher, dont la première sera cochée par défaut, selon le visuel.
 4. Des `<fieldset>` et des légendes pour rendre notre formulaire un peu plus lisible.

Info personnelles

Prénom: Ex: Jacques

Nom: Ex: Dubois

E-mail: Ex: monMail@FAI.com

Numéro: Ex: 00.11.22.33.44

Mot de passe: Ex: P@ssWo4d

Age: Ex: 15

Laissez un commentaire:

J'adore ce que vous faites !

Date de mon commentaire: jj/mm/aaaa

Ma photo: Choisir un fichier

Aucun fichier choisi

Vous préférez:

☒ Le HTML

☐ Le CSS

☐ Le JavaScript

☐ Ne rien faire

HTML Les formulaires: Input Radio

Les « radio », ou zones d'options, sont des éléments qui ne permettent que des choix simples. Attention, ils doivent avoir le même « name » mais des « value » différentes :

```
<form method="POST" action="monProgramme.php">  
  <input type="radio" id="radio1" name="radio" value="radio1"/><label for="radio1">Radio 1</label>  
  <br/>  
  <input type="radio" id="radio2" name="radio" value="radio2"/><label for="radio2">Radio 2</label>  
</form>
```

☒ Radio 1
☐ Radio 2

Vous pouvez ajouter l'attribut « checked », sans valeur, pour cocher une option par défaut.

HTML Les formulaires: Listes déroulantes

Les listes déroulantes sont composées de deux balises pour fonctionner. La première sera la balise `<select></select>` qui permet d'indiquer le début et la fin de la liste.

La seconde balise sera `<option></option>` qui permet d'indiquer chaque choix possible.

```
<form method="POST" action="monProgramme.php">  
  <label for="liste">Je suis une liste</label>  
  <select name="liste" id="liste">  
    <option value="option1">Option 1</option>  
    <option value="option2">Option 2</option>  
    <option value="option3">Option 3</option>  
    <option value="option4">Option 4</option>  
  </select>  
</form>
```

Je suis une liste

Je suis une liste

Option 1 ▾
Option 1
Option 2
Option 3
Option 4

HTML Les formulaires: Listes déroulantes

Vous pouvez ajouter l'attribut « selected », sans valeur, pour sélectionner une option par défaut.

Les choix, `<option>`, peuvent aussi être rassemblés dans des groupes avec les balises `<optgroup></optgroup>` qui auront comme attribut « label » pour nommer le groupe :

```
<form method="POST" action="monProgramme.php">
  <label for="liste">Je suis une liste</label>
  <select name="liste" id="liste">
    <optgroup label="Groupe 1">
      <option value="option1">Option 1</option>
      <option value="option2">Option 2</option>
    </optgroup>
    <optgroup label="Groupe 2">
      <option value="option3">Option 3</option>
      <option value="option4">Option 4</option>
    </optgroup>
  </select>
</form>
```

Je suis une liste

Option 1	▼
Groupe 1	
Option 1	
Option 2	
Groupe 2	
Option 3	
Option 4	

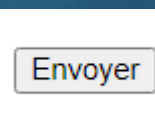
HTML Les formulaires: Les boutons

Ils sont au nombre de quatre qui ont chacun leurs particularités. Il suffit de rajouter l'un des types suivant à la balise `<input>`:

- « submit » => LE bouton du formulaire, permet d'effectuer l'action qui est indiquée dans la balise `<form>` (method + action).
- « image » => Comme pour le bouton « submit », sauf qu'ici, le bouton est sous forme d'image.
- « reset » => Remet le formulaire dans son état d'origine (efface tous les champs).
- « button » => Bouton générique, c'est à nous de lui donner son action (souvent du JavaScript).

Pour lui ajouter un texte, il faut y ajouter l'attribut « value ».

```
<form method="POST" action="monProgramme.php">  
  <input type="submit" value="Envoyer"/>  
</form>
```



HTML Les formulaires: Mise en pratique (30 min)

- Reprenez le précédent formulaire.
- Ajoutez y les champs suivants :
 1. Un ensemble de quatre radio selon le visuel, avec le premier sélectionné par défaut.
 2. Mettez cet ensemble dans un `<fieldset>` avec une légende.
 3. Une liste déroulante suivant le visuel, avec un champ (de votre choix) par défaut.
 4. Deux boutons, un « submit » et un « reset ».
 5. Mettez cet ensemble dans un `<fieldset>` avec une légende.

Info personnelles

Prénom: Ex: Jacques

Nom: Ex: Dubois

E-mail: Ex: monMail@FAI.com

Numéro: Ex: 00.11.22.33.44

Mot de passe: Ex: P@ssWo4d

Age: Ex: 15

Laissez un commentaire:

J'adore ce que vous faites !

Date de mon commentaire: jj/mm/aaaa

Ma photo: Choisir un fichier Aucun fichier choisi

Vous préférez:

☒ Le HTML
 ☐ Le CSS
 ☐ Le JavaScript
 ☐ Ne rien faire

Vous vous levez:

☒ Le matin
 ☐ Le midi
 ☐ Le soir
 ☐ En retard !

Fin du formulaire

Vous en avez assez ? J'en veux plus

Envoyer Reset

Team Pas encore

J'en veux plus

C'est super !

Team Oula

J'en ai marre

Pitié !

HTML Les formulaires: Aller plus loin

Il existe encore plus de choses pour les `<input>` et les formulaires. Je vous invite à aller voir la doc :

<https://developer.mozilla.org/en-US/docs/Web/HTML/Element/Input>

Et bien sûr à chercher sur internet !