

# Activité HTML N° 3 les formulaires

## Présentation

Dans une page HTML, les **formulaires** permettent de **saisir** des données :

- l'utilisateur **saisit** des données,
- ensuite l'utilisateur **valide** sa saisie et **soumet** le formulaire c'est à dire demande l'envoi des informations saisies au serveur Web,
- le serveur Web **reçoit** les données, les traite et **renvoie** ensuite la **réponse**.

Dans ce **mode classique de fonctionnement** des applications Web :

- c'est donc l'utilisateur, le **client**, qui **sollicite** de **temps en temps** le serveur en envoyant des requêtes avec des données ;
- le serveur **attend** uniquement les requêtes de plusieurs clients pour y **apporter** une réponse.

Pour le projet de voiture, nous allons **procéder différemment** en utilisant un **Websocket** qui permet :

- à l'**utilisateur d'envoyer** des données, des requêtes, au serveur pour **commander** par exemples les **moteurs** ;
- mais au **serveur de prendre l'initiative d'envoyer** des données à l'utilisateur comme par exemple les **informations des capteurs** de la voiture.
- il s'agit d'une **communication bidirectionnelle** qui peut être **simultanée**.

Pour cela, il sera **nécessaire d'utiliser le langage Javascript** dans la page HTML pour permettre au navigateur de **dialoguer** avec le serveur dans les **deux sens** : **envoi** de données au serveur et **réception** de données du serveur.

## Créer un formulaire dans une page HTML

Les formulaires sont insérés avec le couple de balise principal `<form> </form>`.

- **Complète** le contenu du **fichier index.html** du **site Web virtuel ISN** pour ajouter le formulaire suivant :

```
<h1>Commande de la voiture</h1>
<form>
  <p>Capteurs de la voiture</p>
  <p>Commande des moteurs</p>
</form>
```

- connecte-toi au site ISN à l'adresse <http://10.3.141.1/> et visualise le résultat de ta page.

La balise **form** utilise les attributs **method** et **action** pour préciser quelle page du serveur doit être **activée** afin de traiter les données envoyées. Mais dans le cadre de la mise en oeuvre d'un **Websocket** géré par javascript, ces attributs **ne seront pas utilisés**.

## Les zones de saisie basiques

La balise `<input />` permet de **saisir** des informations en précisant :

- le **type de données** : texte (**text**), mot de passe (**password**), e-mail (**email**), URL (**url**), numéro de téléphone (**tel**), nombre (**number**), curseur (**range**), couleur (**color**), cases à cocher (**checkbox**), bouton radio (**radio**), etc.
- le **nom** de la balise : `name="nom"`
- son **identifiant** : `id="nomidentifiant"`, ce qui sera très important pour interagir avec javascript.

Le champ de **type nombre** (`number`) peut être personnalisé avec les attributs suivants :

- **min**: valeur minimale autorisée.
- **max**: valeur maximale autorisée.
- **step**: le **pas** d'incrément.

Une **étiquette** est nécessaire pour indiquer la **signification** du champ de saisie. C'est le rôle du couple de balises `<label>...</label>`. L'attribut **for** de cette balise permet de faire le lien avec l'ID du champ input concerné.

- **Complète** le contenu du **fichier index.html** pour ajouter dans le formulaire les champs de saisie de la vitesse des moteur :

```
<h1>Commande de la voiture</h1>
<form>
  <p>Capteurs de la voiture</p>
  <p>Commande des moteurs</p>
  <label for="vitesse1">Vitesse moteur 1 : </label>
  <input type="number" id="vitesse1" name="vitesse1" min="0" max="100" step="10" value="0"/>
</form>
```

L'attribut **value** permet de définir une valeur par défaut.

## Les éléments d'options

### Les cases à cocher

Les cases à cocher permettent de faire **un ou plusieurs choix** parmi une liste de possibilités. \* **Complète** le formulaire pour ajouter une liste de choix des flèches à activer :

```
<p>
  Cochez les flèches à activer :<br />
  <input type="checkbox" name="flecheAvancer" id="flecheAvancer" /> <label
for="flecheAvancer">Avancer</label><br />
  <input type="checkbox" name="flecheReculer" id="flecheReculer" /> <label
for="flecheReculer">Reculer</label><br />
  <input type="checkbox" name="flecheDroite" id="flecheDroite" /> <label
for="flecheDroite">Droite</label><br />
  <input type="checkbox" name="flecheGauche" id="flecheGauche" /> <label
for="flecheGauche">Gauche</label>
</p>
```

### Les zones d'options (boutons radio)

Les zones d'options vous permettent de faire un **choix unique** parmi une liste de possibilités. Pour cela :

- Elles doivent être **organisées en groupes**,
- posséder le **même nom** (name),
- chaque option doit avoir une valeur (value) différente.

\* **Complète** le formulaire pour ajouter indiquer si le moteur est configuré pour avancer ou pour reculer :

```
<p>
  Choisir le sens de fonctionnement du moteur :<br />
  <input type="radio" name="moteurIsens" value="avancer" id="avancer" checked="True" /> <label
for="avancer">Avancer</label><br />
  <input type="radio" name="moteurIsens" value="reculer" id="reculer" /> <label
for="reculer">reculer</label><br />
</p>
```

L'attribut **checked** permet de définir le bouton radio comme coché par défaut.

## Les boutons

Les boutons sont un type particulier de zone de saisie car il sont cliquables. Leur type peut être :

- **submit** pour soumettre traditionnellement un formulaire,

- **reset** pour réinitialiser le contenu du formulaire,
- **button** pour le rendre cliquable et que vous allez utiliser pour le projet de voiture.
- **image** pour utiliser une image cliquable en précisant dans l'attribut **src** l'URL de l'image

L'attribut **value** permet de mettre un libellé sur le bouton.

- **Complète** le formulaire pour ajouter un bouton **Start** et un bouton **Stop** :

```
<p>
<input type="button" name="start" id="start" value="Start"/>
<input type="button" name="stop" id="stop" value="Stop"/>
</p>
<p>
<input type="reset" name="reset" id="reset" value="Reinitialiser"/>
</p>
```

## Regrouper des groupes de champs

Pour plus de lisibilité et davantage d'organisation, il est utile de regrouper ensemble des champs par fonctionnalités en utilisant les balises `<fieldset>` avec une balise légende `<legend>`. Modifiez votre formulaire en ce sens :

```
<form>
<p>Capteurs de la voiture</p>
<fieldset>
  <legend>Cochez les flèches à activer :</legend>
  <input type="checkbox" name="flecheAvancer" id="flecheAvancer" /> <label
for="flecheAvancer">Avancer</label><br />
  <input type="checkbox" name="flecheReculer" id="flecheReculer" /> <label
for="flecheReculer">Reculer</label><br />
  <input type="checkbox" name="flecheDroite" id="flecheDroite" /> <label
for="flecheDroite">Droite</label><br />
  <input type="checkbox" name="flecheGauche" id="flecheGauche" /> <label
for="flecheGauche">Gauche</label>
</fieldset>
<fieldset>
  <legend>Commande du moteur 1</legend>
  <label for="vitesse1">Vitesse moteur 1 : </label>
  <input type="number" id="vitesse1" name="vitesse1" min="0" max="100" step="10" value="0"/>
  <p>
    Choisir le sens de fonctionnement du moteur :<br />
    <input type="radio" name="moteur1sens" value="avancer" id="avancer" checked="True"/> <label
for="avancer">Avancer</label><br />
    <input type="radio" name="moteur1sens" value="reculer" id="reculer" /> <label
for="reculer">reculer</label><br />
  </p>
  <p>
    <input type="button" name="start" id="start" value="Start"/>
    <input type="button" name="stop" id="stop" value="Stop"/>
  </p>
</fieldset>
</form>
```

Travail à faire : compléter le formulaire pour gérer les deux moteurs.

## Le résultat avec Trinket.io

```
<html>
```

```
</html>
```

## Ressources Les premières balises utile pour le HTML

Voici un lien parmi d'autres pour découvrir les balises de formulaire en HTML 5 :

- <https://openclassrooms.com/courses/apprenez-a-creeer-votre-site-web-avec-html5-et-css3/les-formulaires-8>

### Activité Les langages pour créer des sites Web ...

- [Les langages pour créer des sites Web](#)

From:

/ - [Les cours du BTS SIO](#)

Permanent link:

[/doku.php/isn/html3](#)

Last update: **2019/02/14 20:28**

