

A decorative graphic on the left side of the slide, consisting of a network of white lines and small circles on a blue gradient background, resembling a circuit board or a stylized tree structure.

FORMULAIRES HTML

DÉFINITION & RÔLES

- Les formulaires HTML vont permettre à nos visiteurs de nous envoyer des données que nous allons ensuite pouvoir manipuler et / ou stocker.
- Nous allons utiliser les formulaires pour permettre à des utilisateurs de s'inscrire sur notre site (formulaires d'inscription), de se connecter (formulaire de connexion), de nous envoyer des messages (formulaire de contact), de laisser des commentaires, etc.
- Vous l'aurez donc compris : les formulaires se révèlent indispensables et incontournables dans de nombreuses situations.

DÉFINITION & RÔLES

- Les formulaires HTML vont pouvoir être composés de champs de texte (cas d'un champ de formulaire demandant à un utilisateur de renseigner son adresse mail pour se connecter ou pour s'inscrire sur le site par exemple), de listes d'options (choix d'un pays dans une liste de pays par exemple), de cases à cocher, etc.
- Cependant, vous devez bien comprendre ici qu'on touche avec les formulaires aux limites du langage HTML. En effet, nous allons tout à fait pouvoir construire nos formulaires en HTML, mais le HTML ne va nous permettre ni de recueillir les données envoyées par nos visiteurs, ni de les manipuler, ni de les stocker.
- Pour réaliser ces différentes opérations, il faudra utiliser d'autres types de langages comme le PHP (pour la manipulation des données) et le MySQL (pour le stockage) par exemple qui vont s'exécuter côté serveur.

L'ÉLÉMENT FORM ET SES ATTRIBUTS

- Créer un formulaire avec l'élément form. Cet élément va avoir besoin de deux attributs pour fonctionner normalement : les attributs method et action
- L'attribut method indique comment vont être envoyées les données saisies par l'utilisateur. Il peut prendre 2 valeurs GET ou POST
 - GET → les données transit via l'URL (sous forme de paramètres) → envoyées en clair
 - POST → envoyer les données du formulaire via transaction post HTTP → visibles par personne
- L'attribut action va préciser l'adresse de la page qui va traiter les données

CRÉATION D'UN FORMULAIRE SIMPLE HTML

- Pseudo / Mot de passe / Mail → 3 champs html avec l'élément input
- Input : élément qui va permettre à l'utilisateur d'envoyer des données. Il se présente sous la forme d'une balise orpheline et va obligatoirement posséder un attribut type avec différentes valeurs : type="text" type="email" type="date"
- Si le format de données entrées par l'utilisateur ne correspond pas à ce qui est attendu, alors il ne pourra pas envoyer le formulaire et un message d'erreur s'affichera selon le navigateur utilisé.

CRÉATION D'UN FORMULAIRE SIMPLE HTML

```
<!DOCTYPE html>
<html>
  <head>
    <title>Cours HTML et CSS</title>
    <meta charset= "utf-8">
    <link rel="stylesheet" href="cours.css">
  </head>

  <body>
    <form method="post" action="form.php">
      <input type="text">
      <input type="password">
      <input type="email">
    </form>
  </body>
</html>
```

En l'état, notre formulaire n'est cependant pas exploitable pour la simple et bonne raison que l'utilisateur ne possède pas d'indication sur ce qu'il doit rentrer dans chaque champ et également car pour le moment notre formulaire ne possède pas de bouton d'envoi des données !

AJOUT D'INDICATIONS AVEC LABEL

- Pour donner des indications sur les données attendues dans chaque champ aux utilisateurs, nous allons utiliser des éléments label. Cet élément va permettre d'attribuer un libellé (c'est-à-dire une étiquette ou une description) à chaque champ de notre formulaire.
- Pour des raisons d'ergonomie et de cohérence, il est considéré comme une bonne pratique de lier chaque label à son input correspondant.

LIER LABEL ET INPUT AVEC ID ET FOR

```
<form method="post" action="form.php">
  <div class="champ">
    <label for="pseudo">Entrez un pseudonyme : </label>
    <input type="text" id="pseudo">
  </div>
  <div class="champ">
    <label for="pass">Entrez un mot de passe :</label>
    <input type="password" id="pass">
  </div>
  <div class="champ">
    <label for="mail">Entrez un mail :</label>
    <input type="email" id="mail">
  </div>
</form>
```


AJOUT DU BOUTON D'ENVOI DE DONNÉES

- Créer un bouton de soumission de formulaire pour l'envoi vers form.php
- Input type="submit" → avec l'attribut value="envoyer"

```
<form method="post" action="form.php">
  <div class="champ">
    <label for="pseudo">Entrez un pseudonyme : </label>
    <input type="text" id="pseudo">
  </div>
  <div class="champ">
    <label for="pass">Entrez un mot de passe :</label>
    <input type="password" id="pass">
  </div>
  <div class="champ">
    <label for="mail">Entrez un mail :</label>
    <input type="email" id="mail">
  </div>
  <div class="champ">
    <input type="submit" value="Envoyer">
  </div>
</form>
```

IDENTIFICATION DES DONNÉES

- Il faut nommer les données pour pouvoir les manipuler
- Pour identifier les données et pouvoir ensuite les manipuler en PHP ou autre, nous allons devoir ajouter un attribut name qui doit également posséder une valeur unique à chaque élément de formulaire demandant des données à l'utilisateur.

```
<form method="post" action="form.php">
  <div class="champ">
    <label for="pseudo">Entrez un pseudonyme : </label>
    <input type="text" id="pseudo" name="pseudo">
  </div>
  <div class="champ">
    <label for="pass">Entrez un mot de passe :</label>
    <input type="password" id="pass" name="pass">
  </div>
  <div class="champ">
    <label for="mail">Entrez un mail :</label>
    <input type="email" id="mail" name="mail">
  </div>
  <div class="champ">
    <input type="submit" value="Envoyer">
  </div>
</form>
```

LES ÉLÉMENTS DE FORMULAIRES HTML

Élément	Définition
form	Définit un formulaire
input	Définit un champ de données pour l'utilisateur
label	Définit une légende pour un élément input
textarea	Définit un champ de texte long
select	Définit une liste de choix
optgroup	Définit un groupe d'options dans une liste
option	Définit une option dans une liste
fieldset	Permet de regrouper les éléments d'un formulaire en différentes parties
legend	Ajoute une légende à un élément fieldset

L'ÉLÉMENT INPUT ET LES VALEURS DE L'ATTRIBUT TYPE

Type d'input	Définition
text	Définit un champ de saisie monoligne qui accepte du texte
number	Définit un champ qui accepte un nombre décimal
email	Crée un champ qui permet de renseigner une adresse mail
date	Permet à l'utilisateur d'envoyer une date
password	Créer un champ de saisie monoligne dont la valeur va être cachée
checkbox	Permet de créer une case à cocher. L'utilisateur peut cocher une ou plusieurs cases d'un coup
radio	Permet de créer un bouton radio. Par définition, un seul bouton radio peut être coché dans un ensemble
url	Crée un champ qui accepte une URL
file	Permet à un utilisateur de télécharger un fichier
submit	Crée un bouton d'envoi des données du formulaire

- Textarea → laisser la possibilité à un utilisateur d'écrire un texte long
- Select → permettre de créer une liste d'options déroulante
- Option → autant qu'on veut donner à l'utilisateur & à mettre à l'intérieur d'un select
- Optgroup → grouper les différentes options
- Fieldset → grouper plusieurs champs dans un formulaire pour l'organiser en parties.
- Legend → ajouter une légende à une partie de notre formulaire déterminée par fieldset pour expliquer son but par exemple.

- Création d'un formulaire HTML complet

- Section 1 :

- Son nom de famille ;
- Son prénom ;
- Son adresse mail ;
- Son âge ;
- Son sexe ;
- Son pays de résidence.

- Section 2 :

- Ses compétences parmi une liste de choix ;
- Une description d'une expérience professionnelle passée / d'un problème résolu.

- Section 3 :

- Un mot de passe – obligatoire.

ATTRIBUTS DES FORMULAIRES & SÉCURITÉ

- Règle 1 : il ne faut jamais faire confiance aux données envoyées par les utilisateurs.
- Formulaire précédent : envoyer des valeurs aberrantes comme 2000 ans pour son âge ou des données de type invalide comme des éléments HTML à la place de son nom ou encore de renvoyer le formulaire à moitié ou complètement vide.

LES SOLUTIONS

- Ajouter des contraintes avec les attributs :
 - Required / min / max
- tester que les données nous conviennent dès le remplissage d'un champ ou au moment de l'envoi du formulaire grâce au HTML ou au JavaScript (principalement) et bloquer l'envoi du formulaire si des données ne correspondent pas à ce qu'on attend.
→ insuffisant
- Contre les utilisateurs malveillants, nous allons donc également devoir vérifier les données après l'envoi du formulaire et neutraliser les données potentiellement dangereuses. Nous allons effectuer ces vérifications en PHP, côté serveur.

Attribut	Définition
size	Permet de spécifier le nombre de caractères dans un champ
minlength	Permet de spécifier le nombre minimum de caractères dans un champ
maxlength	Permet de spécifier le nombre maximum de caractères dans un champ
min	Permet de spécifier une valeur minimale pour un champ de type number ou date
max	Permet de spécifier une valeur maximale pour un champ de type number ou date
step	Permet de définir un multiple de validité pour un champ acceptant des données de type nombre ou date. En indiquant <code>step="4"</code> , les nombres valides seront -8, -4, 0, 4, 8, etc.
autocomplete	Permet d'activer l'auto complétion pour un champ : si un utilisateur a déjà rempli un formulaire, des valeurs lui seront proposées automatiquement lorsqu'il va commencer à remplir le champ
required	Permet de forcer le remplissage d'un champ. Le formulaire ne pourra pas être envoyé si le champ est vide
pattern	Permet de préciser une expression régulière. La valeur du champ devra respecter la contrainte de la regex pour être valide

MODIFICATIONS DU FORM PRÉCÉDENT

- Reprenons notre formulaire précédent en se concentrant sur certains champs en particulier et ajoutons quelques contraintes sur les données que l'on souhaite recevoir :
 - Les nom, prénom, adresse mail et mot de passe sont désormais obligatoires
 - Les nom et prénom doivent faire au moins 2 caractères et maximum 25 caractères ;
 - L'âge doit être compris entre 16 et 99 ans ;
 - La description de l'expérience professionnelle ne doit pas excéder 500 caractères ;
 - Le mot de passe doit contenir au moins 8 caractères.

A decorative graphic on the left side of the slide, consisting of a network of light blue lines and small circles, resembling a circuit board or a stylized tree structure.

MERCI À VOUS !
AVEZ-VOUS DES QUESTIONS ?