

An abstract geometric design on the left side of the slide. It features a dark blue background with various geometric shapes and patterns. A white circle is positioned near the top left. Below it, a light blue semi-circle is visible. To the right of the semi-circle, there is a pink triangle with diagonal lines. Further down, there is a pink square with a pattern of concentric lines. At the bottom, there is a pink triangle with a pattern of concentric lines. The overall design is modern and minimalist.

HTML FORMULAIRES

FORMULAIRES HTML

INTRODUCTION

Les formulaires sont un élément essentiel du web.

C'est en effet eux qui transforment un site statique ou un site vitrine en véritable application web.

Ils permettent aux utilisateurs de renseigner des informations avec lesquelles nous pourrons ensuite réaliser divers traitements.

Le traitement des données se fera avec un langage de programmation « serveur » (PHP, Java, Python, ...). Pour l'instant nous verrons seulement comment créer un formulaire et envoyer les données.

FORMULAIRES HTML

INTRODUCTION

Voici divers exemples de formulaires :

Un formulaire de recherche

The screenshot shows a search results page for the term 'développeur'. At the top, there is a search bar with the text 'développeur' and an 'OK' button. Below the search bar, the results are categorized: '51 résultats', '23 formations', '28 centres Afpa', and '0 article'. A sidebar on the left allows for refining the search with filters like '23 formations disponibles', 'En alternance (5)', 'Eligible au CPF (6)', and various domains, regions, and project types. The main content area displays 'Entrées rapides en formation' with a link to 'Informatique - Numérique' and a button 'Découvrir la formation'. Below this, there are 'Nos formations' with a link to 'Fonctions supports de l'entreprise' and a button 'Découvrir la formation'.

Un formulaire d'inscription

The screenshot shows a registration form titled 'Remplissez le formulaire en fonction de votre profil' with a note '*champs obligatoires'. The form is divided into sections. The first section, 'Vous êtes*', has two radio buttons: 'Particulier' (selected) and 'Partenaire emploi'. The second section, 'Renseignements', contains several fields: 'Civilité*' with radio buttons for 'Madame' and 'Monsieur'; 'Nom*' and 'Prénom*' text boxes; 'Date de naissance*' with a date picker (format jj/mm/aaaa) and a calendar icon; 'Téléphone*' text box; 'Code Postal*' text box; and 'Ville de résidence*' text box. At the bottom, there is a question 'Êtes-vous actuellement stagiaire ou ancien stagiaire Afpa ?' with radio buttons for 'Oui' and 'Non' (selected).

FORMULAIRES HTML

LES BASES

Pour faire un formulaire, nous utiliserons 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 (verbes HTTP) et l'adresse du programme qui doit traiter le formulaire soumis.

FORMULAIRES HTML

LES BASES

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

- Méthode « GET » => Les paramètres sont envoyés dans l'URI. Le « ? » est le symbole qui définit le début de la liste des paramètres, puis les paramètres sont envoyés de la façon « clé=valeur » et séparés par une esperluette « & ». La méthode GET est utilisée lorsque l'on souhaite accéder à des données en lecture. Le nombre de caractères est limité.

```
http://www.monSuperSite.com/contact.php?nom=Roger&mail=roger@mail.com
```

- Méthode « POST » => Les paramètres sont envoyés dans le « body » de la requête. La méthode POST est utilisée lorsqu'on souhaite exécuter une action qui va ajouter ou modifier des données en base de données.

FORMULAIRES HTML

LES BASES

L'attribut « action » quant à lui, permet de choisir l'adresse où envoyer les données (un programme PHP par exemple).

Voici ce que ça donne :

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

FORMULAIRES HTML

CHAMP DE FORMULAIRE : LES INPUT

La balise la plus courante dans un formulaire est la balise auto-fermante « <input> »

Cette balise prend, au minimum, un attribut : « type » qui permet de définir le type de données envoyés.

La valeur la plus commune de cet attribut est « text ». Ça permet de créer un input recevant une entrée utilisateur de type texte (par exemple pour indiquer son nom ou son prénom)

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

FORMULAIRES HTML

CHAMP DE FORMULAIRE : LES INPUT

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

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

Rappel : Tout ce qui est fait côté front est fait dans un souci d'ergonomie. En aucun cas ça ne devra être la seule façon de vérifier que les données envoyées par l'utilisateur sont conformes à ce que l'on attend.

FORMULAIRES HTML

CHAMP DE FORMULAIRE : 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>
```

Au clic sur un label, le focus est mis dans l'input. Ça permet d'augmenter la hitbox de l'input et donc l'ergonomie du form.

FORMULAIRES HTML

CHAMP DE FORMULAIRE : INPUTS ALTERNATIFS

HTML 5 a ajouté de nouveaux types d'input. Encore une fois, ces nouveaux types ne jouent aucun rôle dans la validation car ils sont très facilement bypassables (en changeant le type d'input grâce à la console du navigateur), mais ils apportent du confort d'utilisation.

On trouve par exemple :

- 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

FORMULAIRES HTML

MISE EN PRATIQUE (25MN)

Dans un nouveau répertoire, créez un fichier « formulaire.html » avec le HTML de base.

Créez un formulaire qui contiendra:

- Un champ pour le prénom, limité à 20 caractères, avec au moins 2 caractères minimum et un placeholder
- Un champ pour le nom, limité à 20 caractères, avec au moins 2 caractères minimum et un placeholder
- Un champ pour l'email, avec un placeholder
- 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"/>

FORMULAIRES HTML

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

Attention, là encore, la sécurité n'est que visuelle. Pour sécuriser correctement un mot de passe, il faut :

- Utiliser la méthode « POST » : évite que le pass soit envoyé dans l'url et donc se retrouve dans l'historique de navigation
- Utiliser le SSL (HTTPS) afin que le contenu du formulaire soit chiffré. Dans le cas contraire, quiconque présent sur le réseau pourrait « sniffer » le réseau et voir passer le pass en clair.

FORMULAIRES HTML

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>
```

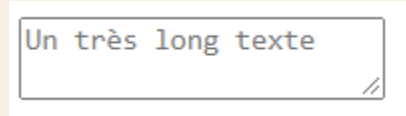


FORMULAIRES HTML

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 »).

FORMULAIRES HTML

MISE EN PRATIQUE (25MN)

Reprenez le précédent formulaire

Ajoutez-y les champs suivants:

- Un champ pour le mot de passe, avec un minimum de 8 caractères et un maximum de 30. Il doit contenir un placeholder.
- Un champ pour l'âge, avec un placeholder.
- 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:

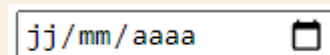
Laissez un commentaire:

FORMULAIRES HTML

INPUT « DATE »

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

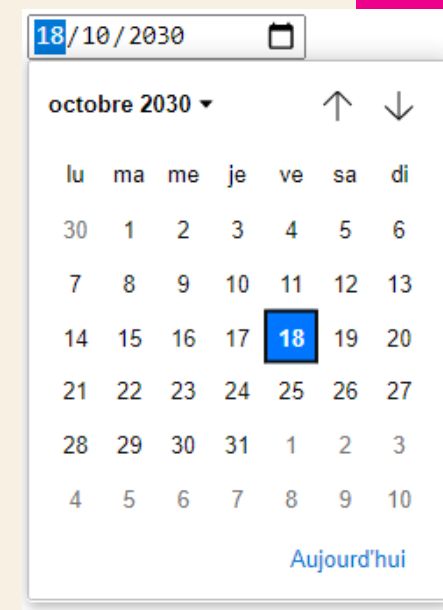
```
<form method="POST" action="monProgramme.php">  
    <input type="date"/>  
</form>
```



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` ».

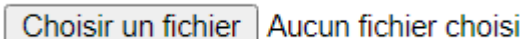


FORMULAIRES HTML

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>
```



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>
```

Vous savez déjà ce que je vais vous dire concernant la sécurité, non ?

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

FORMULAIRES HTML

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.

FORMULAIRES HTML

FIELDSET ET 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"/>

FORMULAIRES HTML

MISE EN PRATIQUE (25MN)

Reprenez le précédent formulaire.

Ajoutez-y les champs suivants :

- Un champ pour la date du commentaire.
- Un bouton pour l'envoi de fichier image (.jpg, .png, .gif).
- Un ensemble de quatre cases à cocher, dont la première sera cochée par défaut, selon le visuel.
- 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

FORMULAIRES HTML

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.

FORMULAIRES HTML

LISTES DÉROULANTES : SELECT LIST

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.

Question : Peut-on se fier à la valeur d'une option reçue dans un formulaire ?

```
<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

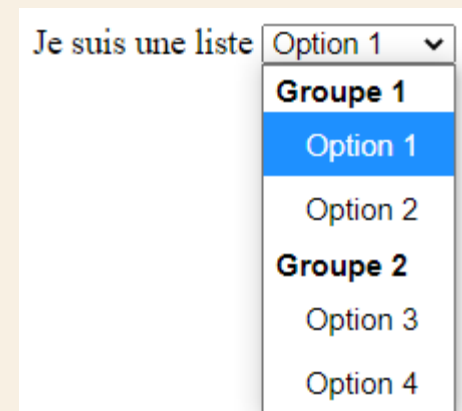
FORMULAIRES HTML

LISTES DÉROULANTES : SELECT LIST

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

FORMULAIRES HTML

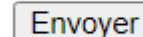
LES BOUTONS

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>
```



FORMULAIRES HTML

MISE EN PRATIQUE (25MN)

Reprenez le précédent formulaire.

Ajoutez-y les champs suivants :

- Un ensemble de quatre radio selon le visuel, avec le premier sélectionné par défaut.
- Mettez cet ensemble dans un <fieldset> avec une légende.
- Une liste déroulante suivant le visuel, avec un champ (de votre choix) par défaut.
- Deux boutons, un « submit » et un « reset ».
- Mettez cet ensemble dans un <fieldset> avec une légende.

The image shows a web form titled "Info personnelles" with several input fields: "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), and "Age:" (Ex: 15). Below these is a "Laissez un commentaire:" section with a text area containing "J'adore ce que vous faites !". A date field "Date de mon commentaire:" shows "jj/mm/aaaa" with a calendar icon. A file upload section "Ma photo:" has a "Choisir un fichier" button and the text "Aucun fichier choisi".

Below the first section is another fieldset titled "Vous préférez:" containing four radio buttons: "Le HTML" (checked), "Le CSS", "Le JavaScript", and "Ne rien faire".

Below that is a fieldset titled "Vous vous levez:" containing four radio buttons: "Le matin" (checked), "Le midi", "Le soir", and "En retard !".

The final section is titled "Fin du formulaire" and contains a dropdown menu "Vous en avez assez ?" with a default value of "J'en veux plus". Below the dropdown are "Envoyer" and "Reset" buttons. The dropdown menu is open, showing options: "Team Pas encore", "J'en veux plus" (highlighted), "C'est super !", "Team Oula", "J'en ai marre", and "Pitié !".

FORMULAIRES HTML

POUR 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 les internet !