

DBWEB 2 :
Architecture Web Dynamique
Licence 1ère Année

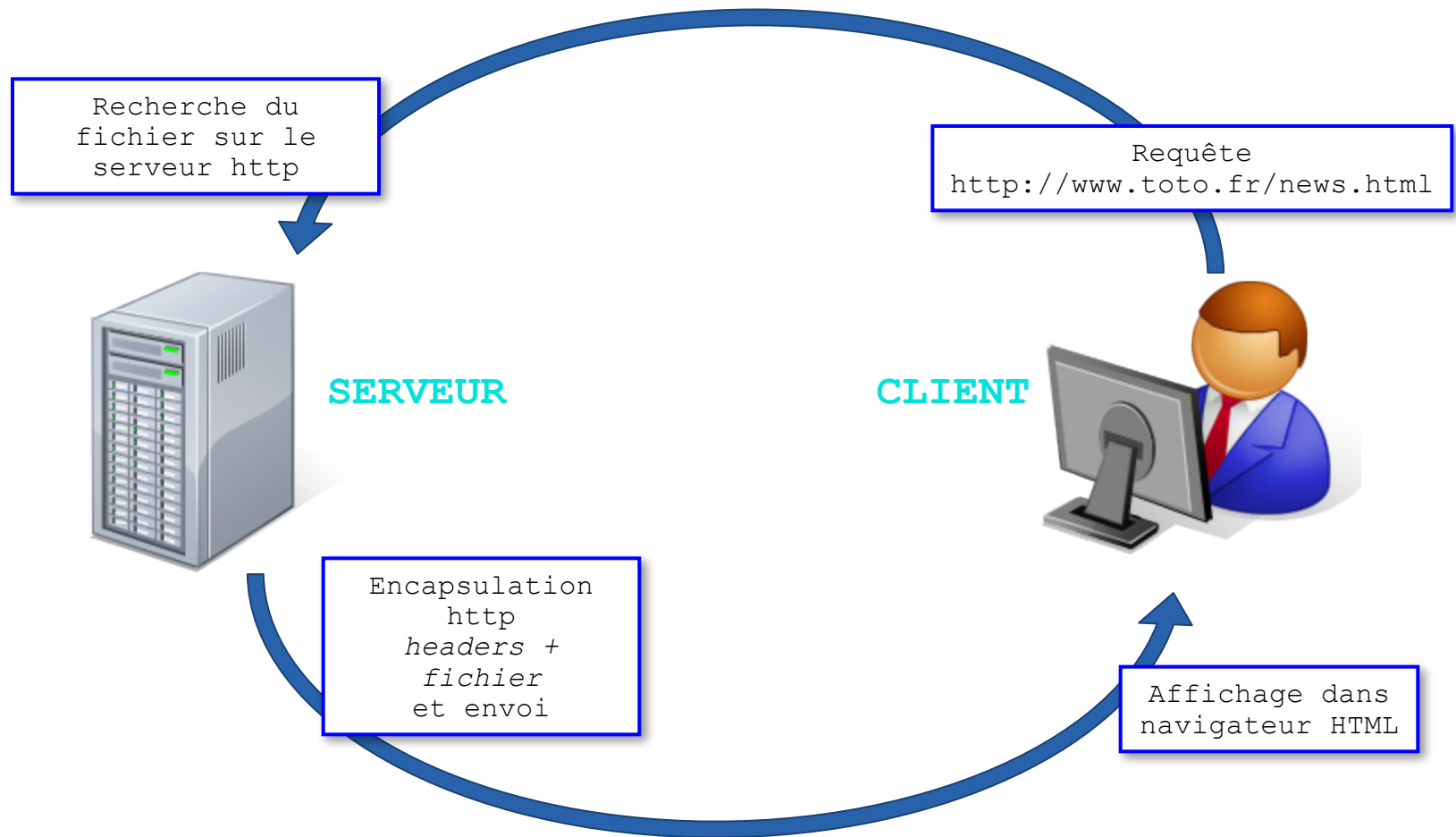
Formulaires HTML

Fabrice Lefèvre

Fabrice.Lefevre@univ-avignon.fr

2021

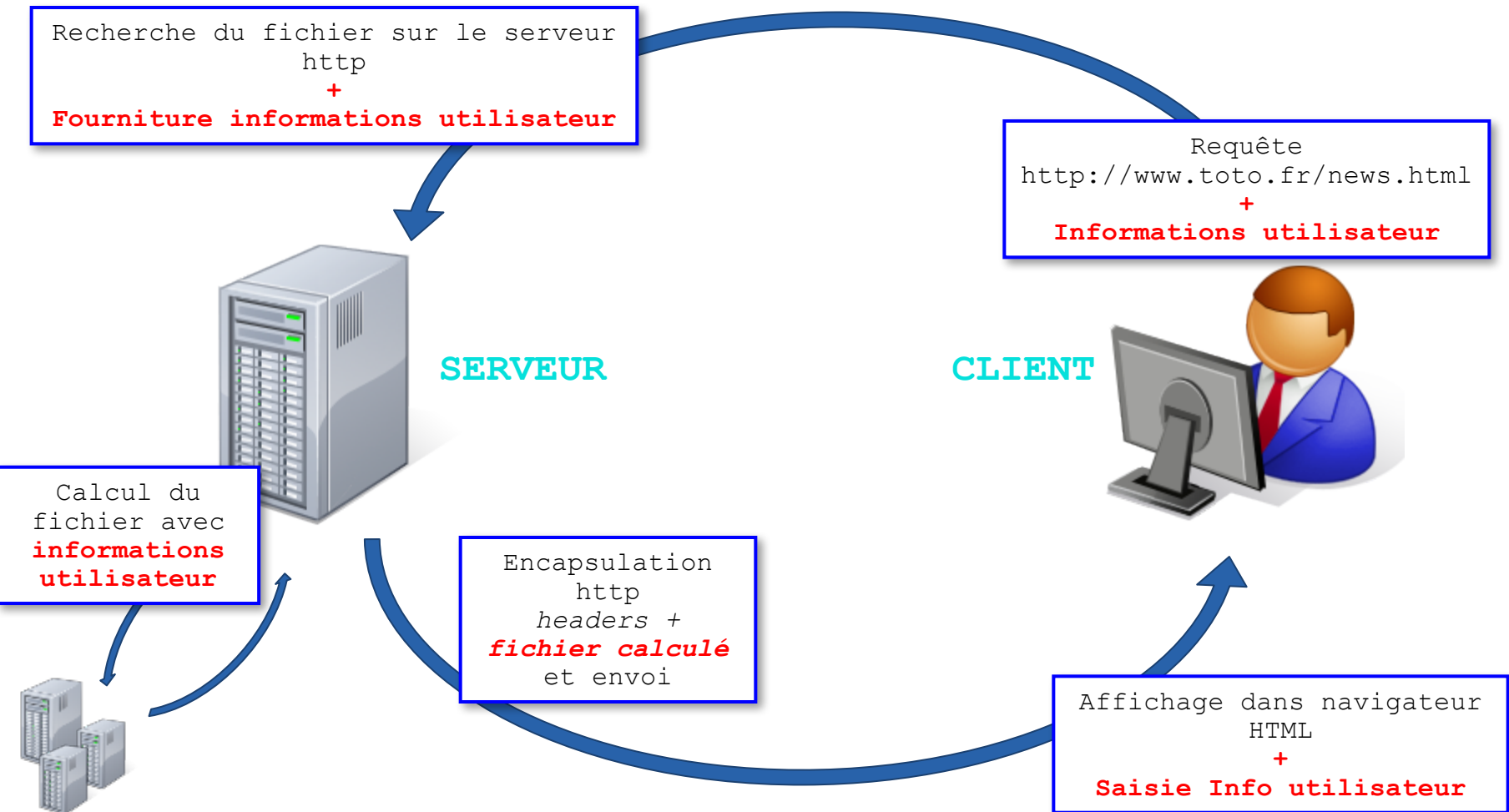
Interaction classique



Web et Contexte (rappel)

- Principes du web dynamique
 - ▣ pages/contenus s'adaptent au contexte d'utilisation du site
- Contexte à plusieurs niveaux
 - ▣ environnement technique (nature et version du navigateur, disponibilité de plugins...)
 - ▣ temporel (actualisation en fonction de la date, l'heure...) et géographique (langue, législation...)
 - ▣ utilisateurs
- **Contexte utilisateur**
 - ▣ **utilisateur doit pouvoir communiquer des informations le concernant**
 - ▣ **permanence des informations**
 - ▣ **confidentialité, sécurisation de la transmission**

Interaction dynamique



Inscriptions

Prénom	<input type="text"/>	Nom	<input type="text"/>
Labo/Institution	<input type="text"/>		
Adresse	<input type="text"/>		
Code Postal	<input type="text"/>	Ville	<input type="text" value="Avignon"/>
Téléphone	<input type="text"/>	Fax	<input type="text"/>
Courriel	<input type="text"/>		
Je ne souhaite pas bénéficier de la nuit d'hôtel incluse dans le forfait (et donc je me débrouillerai seul pour mon logement)		<input type="checkbox"/>	
Je ne souhaite pas partager ma chambre		<input type="checkbox"/>	
Dans le cas contraire, donnez le nom de votre colocataire souhaité		<input type="text"/>	
Je ne souhaite pas participer au repas du 30 au soir (bien qu'il soit inclus dans le forfait)		<input type="checkbox"/>	
Mode de paiement			
<input type="radio"/> Carte Bleue VISA (ou comparable)			
<input checked="" type="radio"/> Chèque bancaire ou postal à l'ordre de l'AFCP			
<input type="radio"/> Bon de commande à l'ordre de l'AFCP			
<input type="button" value="Envoyer"/>		<input type="button" value="Annuler"/>	

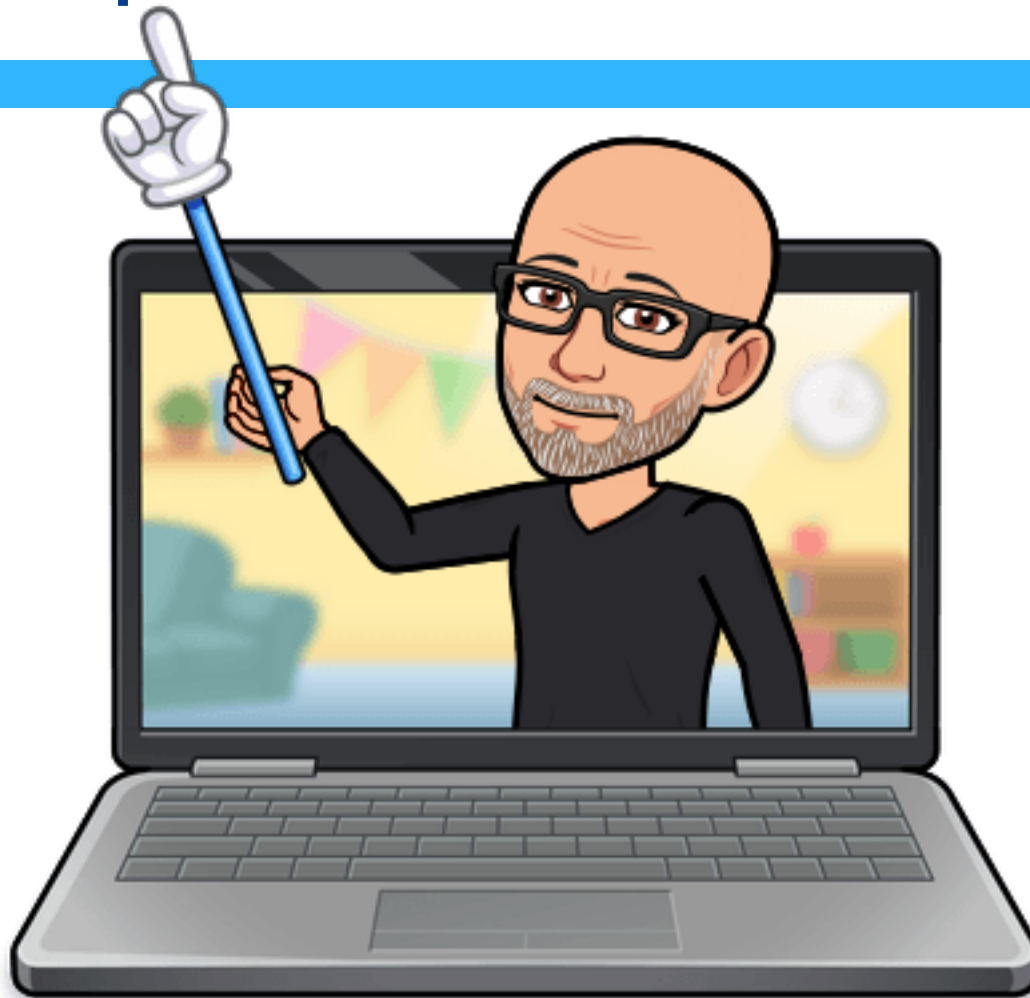
@form_workshop-inscription.html

```
<body style="font-family:Trebuchet MS" alink="#990000" bgcolor="#FFFFFF" link="#881111" text="#000000" vlink="#993300">
  <h2><a class="mozTocH2" name="mozTocId686986"></a>Inscriptions</h2></a>
  <form action="form_workshop-inscription.html" method="get">
    <table style="border: 0pt none ;width:400px;font-family:Trebuchet MS">
      <col span="4" width="1*">
      <tbody>
        <tr style="background-color:#cccccc">
          <td>Prénom</td>
          <td><input name="surname"></td>
          <td>Nom</td>
          <td><input name="name"></td>
        </tr>
        <tr>
          <td>Labo/Institution</td>
          <td colspan="3"><input name="institution" size="60"></td>
        </tr>
        <tr style="background-color:#cccccc">
          <td>Adresse</td>
          <td colspan="3"><textarea name="address" rows="3" cols="50"></textarea></td>
        </tr>
        <tr>
          <td>Code Postal</td>
          <td><input name="zipcode"></td>
          <td>Ville</td>
          <td>
            <select name="city">
              <option>Avignon</option>
              <option>Marseille</option>
              <option>Paris</option>
            </select>
          </td>
        </tr>
        <tr style="background-color:#cccccc">
          <td>Téléphone</td>
          <td><input name="phone"></td>
          <td>Fax</td>
          <td><input name="fax"></td>
        </tr>
        <tr>
          <td>Courriel</td>
          <td colspan="3"><input name="email" size="60"></td>
        </tr>
      </tbody>
    </table>
  </form>
</body>
```

Formulaires (1)

- Objectif : transmettre des informations fournies par l'utilisateur vers le serveur
 - ➔ INTERACTIVITE @form_workshop-inscription.html
- <FORM>... </FORM>
 - ▣ Attribut NAME = "Identifiant du formulaire"
 - ▣ Attribut ACTION = "Adresse cible de la soumission"
 - ▣ Attribut METHOD = "Type de transmission des infos"
- Action doit désigner un « document » sur un serveur capable de traiter les objets du formulaire et retourner une information à l'utilisateur
- Formulaires ont des objets de saisie ou de sélection associés

En pratique



Formulaires (2)

- Formulaire n'est qu'une interface de saisie : d'autres « éléments » sont nécessaires pour traiter l'information qui y est collectée
- Informations du formulaire sont
 - ▣ vérifiées au niveau du client (Javascript, par exemple) puis
 - ▣ traitées ensuite au niveau du serveur par des scripts (scripts CGI ou langage PHP, par exemple) → désignés par l'attribut action
- Objets d'un formulaire peuvent être de plusieurs types mais chaque objet est associé à un nom unique → comme un nom de variable, permet le traitement sur le serveur
- Pour permettre la saisie de TOUS les objets, il existe un objet particulier pour SOUMETTRE le formulaire (i.e. envoyer les informations au serveur désigné par Action)

Mode de transmission (1)

- METHOD="POST"
 - ▣ Les infos sont transmises directement dans un datagramme (paquet réseau) du protocole HTTP
 - ▣ Pas de restriction au niveau de la quantité d'information qui peut être transmise
 - ▣ Informations non visibles au travers du navigateur !
 - Attention : « lisibles » dans le paquet réseau
 - ▣ Recommandée dans la majorité des cas, notamment dans le transfert de fichiers

Mode de transmission (2)

□ METHOD="GET"

- Les infos sont transmises au travers de l'URL vers une variable d'environnement QUERY_STRING

■ Transmission

`http://mon_url?champ1=valeur1&champ2=valeur2`

- Restreint à 2ko caractères maximum

- Informations visibles directement dans l'URL du navigateur :

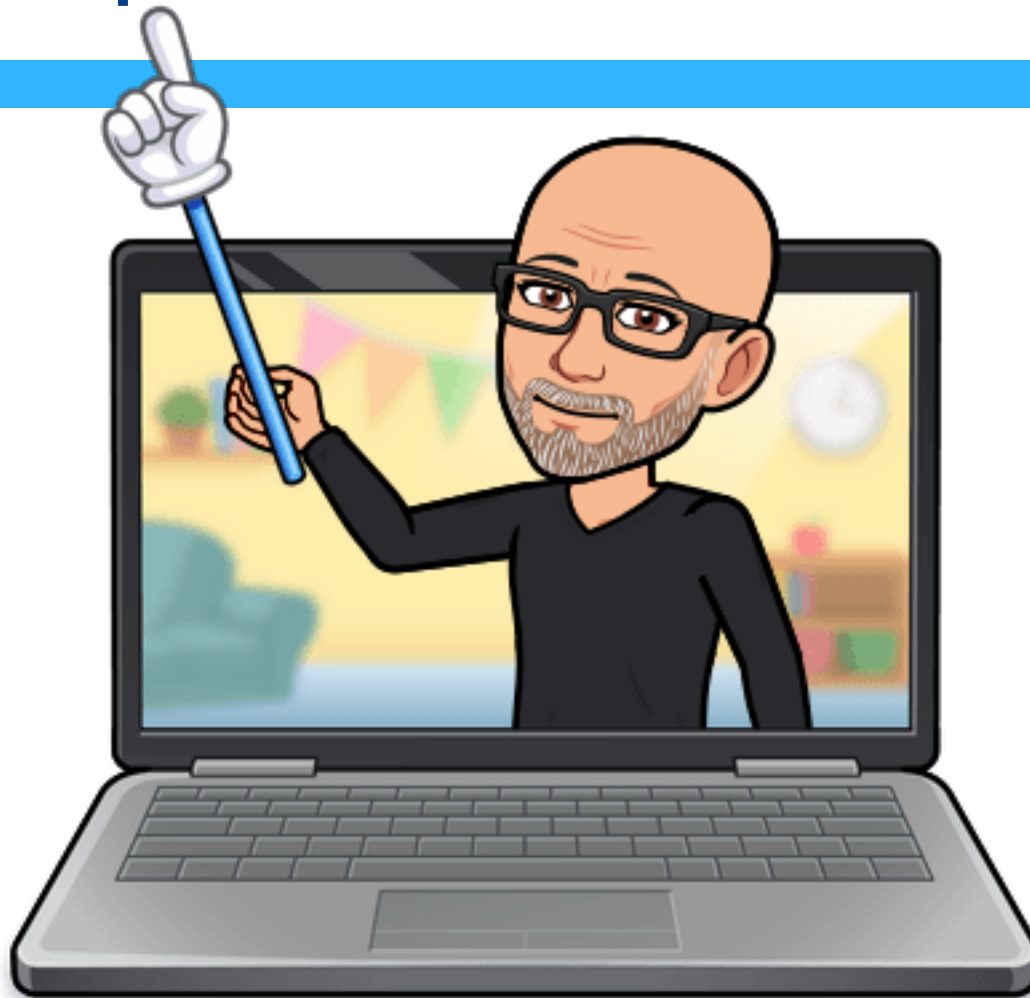
- Pratique pour le « bookmark » si on veut accéder directement à une info particulière sans la ressaisir
- Dangereux si les informations sont sensibles !

Objets de formulaires

Types (logiques) d'objets de formulaires principaux :

- champs de saisies
 - `<INPUT>` types text, password, hidden, file...
 - `<TEXTAREA>`
- sélection d'alternatives
 - `<INPUT>` types radio, checkbox
 - `<SELECT>` listes ou menus déroulants
- boutons d'actions
 - `<INPUT>` types button, submit, image, reset
 - `<BUTTON>`

En pratique



Mise en page des formulaires

- Utilisation classique des CSS @form_input_formater_css.html
- `<FIELDSET>...</FIELDSET>` @form_input_fieldset_legend_label.html
 - ▣ Regroupe des champs
 - ▣ Affichage avec une bordure
 - ▣ `<LEGEND>` permet de donner un titre au groupe de champs
- `<LABEL>...</LABEL>`
 - ▣ `for` = nom de l'input associée
 - ▣ permet de cliquer sur le nom directement
- Modifier l'ordre des tabulations
 - ▣ `tabindex`
- Association de raccourcis-clavier
 - ▣ `accesskey`
- Désactivation
 - ▣ `disabled`
- Mais surtout W3.CSS @form_sansw3css.html @form_w3css.html

Nouveautés HTML 5

- Pas grand chose en fait...
- Nouveaux champs pour FORM :
 - ▣ datalist pour INPUT type = list (simplifie et généralise SELECT)
 - ▣ keygen pour générer clés d'authentification
 - ▣ output pour affichage dynamiques (fonctionne avec Javascript)
- Nouveaux types pour INPUT :
 - ▣ color
 - ▣ date, datetime, datetime-local, month, time week
 - ▣ email
 - ▣ number
 - ▣ range
 - ▣ search
 - ▣ tel
- Nouveaux attributs pour FORM et INPUT : complétion, validation...



Futur des formulaires

□ XFORMS

- ▣ successeur des formulaires HTML
 - ▣ sépare les données de leur représentation/présentation
 - ▣ utilise XML pour définir les données
 - ▣ utilise XML pour stocker et transporter les données
 - ▣ est indépendant de la machine
 - ▣ est une recommandation du W3C
-
- ▣ XForms sera opérationnel avec XHTML 2.0
 - ➔ On a encore un peu de marge !

Formulaire Xforms, exemple (1)

```
<xforms>
<model>
  <instance>
    <person>
      <fname/>
      <lname/>
    </person>
  </instance>
  <submission id="form1" action="action.php" method="get"/>
</model>

<input ref="fname"><label>Prenom</label></input>
<input ref="lname"><label>Nom</label></input>
<submit submission="form1"><label>Soumettre</label></submit>
</xforms>
```

Formulaire Xforms, exemple (2)

Prenom

Nom

```
<person>  
  <fname>Fabrice</fname>  
  <lname>Lefevre</lname>  
</person>
```

DBWEB 2 :
Architecture Web Dynamique
Licence 1ère Année

Formulaires HTML En pratique

Fabrice Lefèvre

Fabrice.Lefevre@univ-avignon.fr

2021

@form_workshop-inscription.html

Inscriptions

Prénom	<input type="text"/>	Nom	<input type="text"/>
Labo/Institution	<input type="text"/>		
Adresse	<input type="text"/>		
Code Postal	<input type="text"/>	Ville	<input type="text" value="Avignon"/>
Téléphone	<input type="text"/>	Fax	<input type="text"/>
Courriel	<input type="text"/>		
Je ne souhaite pas bénéficier de la nuit d'hôtel incluse dans le forfait (et donc je me débrouillerai seul pour mon logement)		<input type="checkbox"/>	
Je ne souhaite pas partager ma chambre		<input type="checkbox"/>	
Dans le cas contraire, donnez le nom de votre colocataire souhaité		<input type="text"/>	
Je ne souhaite pas participer au repas du 30 au soir (bien qu'il soit inclus dans le forfait)		<input type="checkbox"/>	
Mode de paiement			
<input type="radio"/> Carte Bleue VISA (ou comparable)			
<input checked="" type="radio"/> Chèque bancaire ou postal à l'ordre de l'AFCP			
<input type="radio"/> Bon de commande à l'ordre de l'AFCP			
<input type="button" value="Envoyer"/>		<input type="button" value="Annuler"/>	

Objets de formulaires

Types (logiques) d'objets de formulaires principaux :

- **champs de saisies**
 - **<INPUT>** types text, password, hidden, file...
 - **<TEXTAREA>**
- **sélection d'alternatives**
 - **<INPUT>** types radio, checkbox
 - **<SELECT>** listes ou menus déroulants
- **boutons d'actions**
 - **<INPUT>** types button, submit, image, reset
 - **<BUTTON>**

Entrées de formulaire

- **<INPUT>**, une entrée de formulaire @form_inputs.html
 - ▣ Liés obligatoirement à un formulaire
 - ▣ Réalisation de tous les types logiques d'objets de formulaires
 - ▣ Attribut TYPE spécifie le type du champ d'entrée à utiliser dans la page :

Text	Champ de saisie
Password	Champ de saisie password
File	Bouton de sélection fichier
Hidden	Champ caché

Button	Bouton poussoir (pas action)
Submit	Bouton de soumission
Image	Bouton image
Reset	Bouton de réinitialisation

Checkbox	Cases à cocher
Radio	Cases radio

Champs de saisie

@form_saisies.html

- `<INPUT>`, un champ d'entrée parmi :

Text	Champ de saisie
Password	Champ de saisie password
File	Bouton de sélection fichier
Hidden	Champ caché

- `<TEXTAREA>`, un champ de saisie de texte « élargi »
 - ▣ attributs ROWS et COLS pour définir les dimensions de la zone de saisie

Objets de formulaires

Types (logiques) d'objets de formulaires principaux :

- champs de saisies
 - ▣ `<INPUT>` types text, password, hidden, file...
 - ▣ `<TEXTAREA>`
- sélection d'alternatives
 - ▣ `<INPUT>` types radio, checkbox
 - ▣ `<SELECT>` listes ou menus déroulants
- boutons d'actions
 - ▣ `<INPUT>` types button, submit, image, reset
 - ▣ `<BUTTON>`

Listes de choix (1)

- `<INPUT>`, un champ d'entrée parmi :

Checkbox	Cases à cocher
Radio	Cases radio

- Différence principale :
 - ▣ choix multiples → checkbox
 - ▣ choix exclusif → radio
 - entre les boutons portant le même nom

Listes de choix (2)

```
<HTML>                                     @form_choix.html
<HEAD>
  <TITLE>Exemple de liste de choix</TITLE>
</HEAD>
<BODY>
  <FORM action= "form_choix.html" method="get">
    <INPUT type="radio" name="it" value="item1">1er item<BR/>
    <INPUT type="radio" name="it" value="item2">2eme item<BR/>
    <INPUT type="radio" name="it" value="item3" checked>3eme item<BR/>
    <INPUT type="radio" name="it" value="item4">4eme item<BR/>
    <INPUT type="submit" value="Envoyer la sélection">
  </FORM>
</BODY>
</HTML>
```

Listes de choix (3)

- `<SELECT></SELECT>`, une liste simple ou un menu déroulant
 - ▣ Attribut NAME = "Nom du champ"
 - ▣ Attribut SIZE = "V"
 - si V=1 => menu déroulant,
 - si V>1 => liste simple avec V items visibles
 - ▣ Attribut MULTIPLE → liste à choix multiples
 - ▣ Attribut VALUE = "Nom d'1 item dans la liste" → item sélectionné par défaut
- `<OPTION>nom de l'item</OPTION>`, un nouvel item d'une liste
 - ▣ Attribut SELECTED → item sélectionné par défaut
 - ▣ Attribut VALUE= "Valeur renvoyée lors de la sélection"
- `<OPTGROUP><OPTION>...`, un sous-groupe d'options
- Attention : La sélection dans une liste n'implique en rien la soumission des informations → utilisation par exemple d'un bouton submit

Listes de choix (4)

```
<HTML>                                     @form_choix.html
<HEAD>
  <TITLE>Exemple de liste de choix</TITLE>
</HEAD>
<BODY>
  <FORM action="form_choix.html" method="get">
    <SELECT name="listederoul" value="item1" size="1">
      <OPTION value="item1">1er item</OPTION>
      <OPTION value="item2">2ième item</OPTION>
      <OPTION value="item3">3ième item</OPTION>
      <OPTION value="item4">4ième item</OPTION>
    </SELECT>
    <INPUT type="submit" value="Envoyer la sélection">
  </FORM>
</BODY>
</HTML>
```

Objets de formulaires

Types (logiques) d'objets de formulaires principaux :

- champs de saisies
 - ▣ <INPUT> types text, password, hidden, file...
 - ▣ <TEXTAREA>
- sélection d'alternatives
 - ▣ <INPUT> types radio, checkbox
 - ▣ <SELECT> listes ou menus déroulants
- boutons d'actions
 - ▣ <INPUT> types button, submit, image, reset
 - ▣ <BUTTON>

Boutons (1)

- Liés obligatoirement à un formulaire
- Réalisation d'actions :
 - ▣ pré-définies (cf boutons spéciaux)
 - ▣ définies par le concepteur (requiert Javascript)
- Deux possibilités :
 - ▣ `<INPUT type="button" >`
 - ▣ `<BUTTON></BUTTON>`
- `INPUT (type button)` vs `BUTTON` : enrichissement graphique (image cliquable par ex.). Mais possible avec `INPUT type image` et javascript...
- Boutons spéciaux :
 - ▣ `<INPUT type="submit | reset" >`
 - ▣ `<BUTTON type="submit | reset" ></BUTTON>`
 - `submit` = soumission du formulaire
 - `reset` = annulation (remise en état initial des champs)

Boutons (2)

```
<HTML>                                @form_input_submit_reset.html
<HEAD>
<TITLE>Exemple de boutons dans un formulaire</TITLE>
</HEAD>
<BODY>
  <FORM action="" method="get">
    <INPUT type=text name="nom" size="20"><BR />
    <INPUT type=button name="fabrice" value="fabrice_ok"><BR />
    <INPUT type=submit name="butt_submit" value="soumettre"><BR />
    <INPUT type=reset name="butt_reset" value="annuler"><BR />
    <INPUT type=image name="butt_img" src="mountain.jpg" alt="Click
on" height="100" weight="200">
    Cliquer sur l'image pour soumettre<BR />
  </FORM>
</BODY>
</HTML>
```

Boutons (3)

- `<BUTTON>...</BUTTON>`, un bouton
- Attribut TYPE spécifie le type de bouton à utiliser dans la page :

Button	Bouton de lancement d'un script spécifié dans l'attribut NAME
Submit	Bouton de soumission
Reset	Bouton de réinitialisation

- Contenu peut être du texte ou une image
- Liés obligatoirement à un formulaire

Mise en page des formulaires

- `<FIELDSET>...</FIELDSET>` @form_input_fieldset_legend_label.html
 - ▣ Regroupe des champs
 - ▣ Affichage avec une bordure
 - ▣ `<LEGEND>` permet de donner un titre au groupe de champs
- `<LABEL>...</LABEL>`
 - ▣ `for` = nom de l'input associée
 - ▣ permet de cliquer sur le nom directement
- Modifier l'ordre des tabulations
 - ▣ `tabindex`
- Association de raccourcis-clavier
 - ▣ `accesskey`
- Désactivation
 - ▣ `disabled`
- Utilisation classique des CSS @form_input_formatter_css.html
- Mais surtout W3.CSS @form_sansw3css.html @form_w3css.html