

## Architecture et dev Web

### TP 2. Formulaire

#### Rappels :

- Utilisation de la VM Linux
- utilisation de phpstorm
- php -S localhost:8080

#### I. Les formulaires

1. Récupérez les fichiers *index.html* et *traitement.php* sous moodle et testez leur fonctionnement.
2. Lorsque vous appuyez sur le bouton OK du formulaire, remarquez que les valeurs transmises apparaissent dans la barre d'adresse du navigateur :

```
http://127.0.0.1/formulaire/traitement.php?nom=toto&prenom=titi&age=12
```

C'est l'une des particularités de l'envoi de données par la méthode GET. Cette méthode n'est pas satisfaisante lorsqu'il s'agit de transmettre des données confidentielles.

3. Modifiez le fichier *index.html* de façon à ce qu'il utilise la méthode POST pour l'envoi des données ainsi que le fichier *traitement.php* ?

#### Remarque :

Le W3C envisage de supprimer la valeur GET pour l'attribut « method » des formulaires HTML

3. Dans le fichier « *traitement.php* » avant le doctype, faites afficher (fonction `print_r()`) le contenu du tableau `$_POST`. Prenez l'habitude de toujours afficher le contenu de vos tableaux.
4. Si les 3 champs du formulaire sont remplis (vérification avec la fonction `empty()`), créez pour chaque champ une variable (`$nom`, `$prenom` et `$age`) sinon utilisez le code suivant permettant d'ouvrir une boîte d'alerte javascript qui permet revenir à la page précédente :

```
echo "<script type=\"text/javascript\">";  
echo "alert('Complétez tous les champs');";  
echo "window.history.back();";  
echo "</script>";
```

#### Attention :

Ne faites jamais confiance aux données reçues (faille XSS), il est toujours préférable de récupérer les données en utilisant le code ci dessous

```
$prenom=htmlspecialchars($_POST['prenom']);
```

Regarder la documentation de cette fonction pour voir son intérêt. Nous pouvons également utiliser la fonction `strip_tags()`

5. Faites afficher le message suivant dans un paragraphe

```
Bonjour prenom nom. Vous avez n ans.
```

6. Si un internaute arrive sur la page *traitement.php* sans être passé par le formulaire de saisie, il faut le rediriger immédiatement vers ce formulaire. Utilisez la fonction *header* (cherchez sur le net son fonctionnement) pour mettre en œuvre cette modification. Pour détecter le non passage par la page *index.html* vous utiliserez la fonction `isset()` sur la variable « send ».

7. Rajoutez dans le formulaire du fichier "index.html" des cases à cocher de la forme :

## Mes sports préférés

Danse ☐ Handball ☐ Musculation ☐ Surf ☐

8. Ajoutez à chaque élément input la propriété `name="sport[]"` pour éviter une multitude de « name » différents. Pour récupérer les valeurs en PHP vous utiliserez `$_POST["sport"]`  
Dans le fichier traitement.php, faites afficher les compétences de la personne sous forme de liste en utilisant la boucle `foreach()`.
9. Modifiez le code HTML pour utiliser les attributs du HTML5
- les valeurs du nom, prénom et age sont exigées
10. Ajoutez à ce formulaire, les champs de saisie suivant :
- un champ de type email permettant de saisir un email pour la personne. Affichez celui-ci sur la page traitement.php
  - modifier le champ « age » par un champ de type « date » et sur la page traitement faites afficher l'age de la personne.

## II. Formulaire : Transfert de fichier

1. Ecrivez un fichier "envoieFichier.php" contenant le formulaire ci-dessous :

Transfert de fichier

Fichier  Aucun fichier choisi

Clic!

L'utilisateur peut envoyer uniquement un fichier de type jpg.

2. Le traitement de ce formulaire se réalisera **dans le même fichier** "envoieFichier.php" et permettra de faire afficher les valeurs renvoyées par le tableau `$_FILES` (comme `$_POST`). N'oubliez pas de faire afficher le contenu de ce tableau, il a une particularité par rapport à `$_POST`

Transfert de fichier

Fichier

Clic!

**Clés et valeurs du tableau `$_FILES`**

clé : name valeur : 1-3.jpg  
clé : type valeur : image/jpeg  
clé : tmp\_name valeur : C:/Program Files/EasyPHP 2.0b1/tmp/phpE0.tmp  
clé : error valeur : 0  
clé : size valeur : 7537

Le transfert est réalisé !

3. En utilisant la fonction `move_uploaded_file()`, enregistrez dans un dossier nommé « upload » le fichier transféré avec un nom identique à l'origine

### III. A vous

Vous disposez comme lors du TP1 du fichier « srilanka.php »

Nous souhaitons rajouter dans le tableau associatif des nouvelles données sur des villes du Sri Lanka.

1. Créez la page index.php contenant un formulaire permettant d'ajouter des données au tableau associatif. Le traitement de ce formulaire se fera sur la même page
2. Ajoutez le traitement du formulaire. Ce traitement se fera si le visiteur a cliqué sur le bouton « envoyer » et ajoutera dans le tableau associatif les villes suivantes :
  - kandy => lat : 7.2905715000000001                      long : 80.633726199999996    popu : 109343
  - Arugam Bay => lat : 6.8379771                      long : 81.825168699999995    popu : 33625.
3. Quel est le problème ? Parlez en avec votre enseignant.
4. Nous souhaitons ajouter maintenant des images pour chaque ville. Pour les 3 premières villes du tableau associatif, ajoutez directement dans le code du fichier srilanka.php, les chemins des images du dossier image. Sinon ajoutez un champ de formulaire permettant d'ajouter une image à une ville et faites le traitement du formulaire.
5. Créez un fichier « fonctions.php » (ou complétez celui de la semaine précédente) et créez une fonction distance() permettant de calculer la distance totale entre les différentes villes. Utilisez cette fonction en calculant la distance totale. Nous faisons le chemin : Negombo -> Anuradhapura -> kandy -> Arugam Bay

**Rappel du calcul :**

$$\text{Distance} = R \text{ ArcCos} ( \text{Sin}( a ) \text{ Sin}( b ) + \text{Cos}( a ) \text{ Cos}( b ) \text{ Cos}( c-d ) )$$

- R est le rayon de la terre soit 6347 km
- a est la latitude du point A en radian
- b est la latitude du point B en radian
- c la longitude du point A en radian
- d la longitude du point B en radian

Attention : la latitude et la longitude fournie sont en degrés