

- TP n°1 -

Formulaire et intégration Web/PHP

Votre compte-rendu de TP doit être rendu sous la forme d'un **seul fichier dans un format éditable** (en particulier pas d'extension `.pdf`) et comprendre :

- les listings des différents fichiers sources
- des commentaires concis sur les parties importantes du programme
- des exemples d'utilisation du programme, ainsi que des captures d'écran illustrant les résultats
- un rapport de synthèse rappelant l'objectif du TP.

Avertissement : Il est rappelé que le compte-rendu est strictement personnel, les devoirs comportant des ressemblances seront sanctionnés dans la note.
Tout travail rendu après la date limite **ne sera pas corrigé**.

Objectif : Créer un formulaire HTML, et traiter les données entrées par l'utilisateur à l'aide d'un script PHP.

Outils : le cours, la spécification HTML et le validateur du W3C, le manuel PHP en français (disponible en ligne sur <http://www.php.net/manual/fr/>), les feuilles de style CSS...

Exercice 1. Créez une page HTML de votre choix comprenant un formulaire avec les champs suivants :

- un champ texte (`type="text"`)
- un champ mot de passe (`type="password"`)
- une liste déroulante comprenant au moins 5 entrées (balise `<select> ... </select>`)
- un groupe de boutons radio (`type="radio"`)
- un bouton de réinitialisation du formulaire (`type="reset"`)
- un bouton de soumission du formulaire (`type="submit"`)

Vous utiliserez la méthode `post` pour l'envoi du formulaire, en positionnant la valeur de l'attribut `method` de la balise `form` à `"post"`. L'attribut `action` du formulaire devra pointer vers un script, nommé `affiche_parametres.php`, prenant en charge le traitement des données saisies par l'utilisateur dans les champs du formulaire (ce script fait l'objet de l'exercice suivant). Validez la page HTML obtenue.

Exercice 2. Créez le script `affiche_parametres.php` permettant d'afficher les valeurs saisies par l'utilisateur dans les différents champs du formulaire.

Pour ne pas perdre trop de temps avec la mise en page, vous pouvez :

- inclure au début du script le fichier `"entete.inc"` donné avec le sujet (utilisez la fonction `include` vue en cours)
- afficher ensuite le contenu du tableau associatif `$_POST` sous la forme d'un tableau HTML constitué de deux colonnes : la première colonne contenant les noms des différents paramètres du formulaire et la deuxième leurs valeurs (l'utilisation de `foreach` pour le parcours d'un tableau est fortement recommandée)

- inclure en fin de script le fichier "pied.inc" donné avec le sujet (toujours avec `include`).
Votre page pourra avoir l'allure suivante :

Nom du paramètre	Valeur paramètre
nom	merlin
password	abracadabra
couleur	Array
reponse	oui

[Retour au formulaire](#)

Liens utiles : [manuel PHP](#) [validateur W3C](#)

Exercice 3.

1. Ajoutez l'attribut `multiple` à la liste d'options (champ `select`) du formulaire. Sélectionnez plusieurs éléments de la liste (touche `Ctrl`) et regardez le résultat affiché par `affiche_parametres.php`. Que se passe-t-il ?
2. Pour corriger ce problème, appliquez à la liste d'options la méthode vue en cours, consistant à donner un nom de tableau aux champs ayant des valeurs multiples (par exemple : `<select name="liste[]">`). Adaptez en conséquence le script PHP pour tenir compte des paramètres à valeurs multiples ; vous utiliserez la fonction PHP `is_array` pour savoir si le paramètre est une valeur simple ou un tableau.
3. Ajoutez au formulaire HTML un groupe de cases à cocher (`type="checkbox"`), que vous traiterez dans le script PHP avec la même méthode que pour la liste d'options.

Pour ceux à qui il reste du temps :

Exercice 4. On souhaite rendre les champs de type `text` et `password` du formulaire obligatoires : dans le cas où l'utilisateur soumettrait un formulaire mal rempli, il serait automatiquement redirigé sur la page initiale du formulaire.

Pour réaliser ce travail, vous pouvez :

1. modifier la page HTML, afin de stocker tous les champs obligatoires dans un tableau selon la méthode appliquée dans l'exercice précédant
(par ex : Nom : `<input type="text" name="obligatoires[]">`
Mot de passe : `<input type="password" name="obligatoires[]">`)
2. rajouter en début de script PHP un test déterminant si les variables correspondant au champ Nom (`$_POST["obligatoires"][0]`) et au champ Mot de passe (`$_POST["obligatoires"][1]`) ont été correctement initialisées. La fonction `empty` pourra être utilisée pour ce test.
 - Si les champs ont été correctement remplis, le contenu du tableau `$_POST` sera affiché comme précédemment
 - Sinon, un appel à la fonction `header` permettant de rediriger l'utilisateur sur une autre page sera fait.
Vous pourrez par exemple lui passer en argument
`Location: URL_absolue_du_formulaire`
pour une redirection sur la page initiale du formulaire.