

Objectif : Manipuler la syntaxe du langage Php. Prendre en main des librairies de fonctions. Utiliser des redirections et des variables de session.

Plan de la séance :

1. Installation/Prise en main d'une distribution « WAMP ».
2. Découverte de HTTP/php : chaînes de requêtes, entêtes HTTP, Redirections, Cookies, Sessions

Exercice 1 : Chaînes de requête, champs multiples, GET & POST

Objectif : Appréhender les mécanismes permettant d'envoyer des données au serveur, et la manière de les récupérer en php. Utiliser l'onglet "Réseau" de la console de développement.

1. Placer le fichier "ex1.php" fourni sur moodle dans un répertoire de votre choix à l'intérieur du répertoire www de votre serveur Web. Ce fichier devra être utilisé comme cible des formulaires que vous rédigerez dans les questions suivantes.
2. Effectuer une requête HTTP depuis votre navigateur vers le fichier ex1.php. Activer l'onglet Réseau de votre console de développement. Entrer une chaîne de caractères dans le champ texte du formulaire qui s'affiche et soumettre le formulaire. Constater ce qui s'affiche dans la barre d'adresse et dans l'onglet Réseau de la console de développement.
 - Quelle méthode a été utilisée ? Pourquoi ?
 - Comment envoyer d'autres données en éditant la barre d'adresse ?
3. Ajouter un champ sans attribut name, un champ à l'extérieur des balises, et soumettre les données. Que se passe-t-il ?
4. A quoi sert l'attribut value des champs de formulaire ?
5. Essayer d'envoyer plusieurs valeurs associées à la même clé en éditant manuellement la chaîne de requête dans la barre d'adresse.
 - A quel type de champ de formulaire cela correspond-il ?
 - Le serveur est-il capable de récupérer toutes les valeurs ?
 - Comment nommer la clé pour pouvoir récupérer toutes les valeurs côté serveur ?
6. Changer la méthode du formulaire et soumettre des données. Constater que l'onglet Réseau de la console de développement permet toujours de visualiser les données envoyées alors qu'elles ne sont plus visibles dans la barre d'adresse, en cliquant sur la requête affichée et en consultant l'onglet "Paramètres".
7. Est-il possible d'utiliser plusieurs boutons de soumission dans un même formulaire, associés au même attribut "name" ? Mettre en place un exemple mettant en évidence l'intérêt de cette pratique.

Exercice 2 : Sécurisation, rappels de HTTP, Redirections, Cookies & Sessions

Objectif : Comprendre les entêtes HTTP, utiliser la fonction `header()` en php, comprendre les mécanismes mis en oeuvre par les variables de session. Découvrir les bonnes pratiques de développement par l'utilisation des fonctions `isset` et `die`.

1. Créer une page « `accueil.html` » contenant un lien vers une page « `redirection.php` » permettant au serveur d'envoyer une entête pour rediriger vers la page « `accueil.php` ».
2. Ecrire du code HTML après l'appel à la commande de redirection. Afficher les échanges d'entêtes dans l'onglet Réseau correspondant à la requête vers la page « `redirection.php` ». Quelle est la valeur de l'entête `Content-length` ?
 - Utiliser la fonction « `die` » juste après la redirection. Quelle est maintenant la valeur de l'entête `Content-length` ? Expliquer ce phénomène.
3. Une page « `ouvrir_session.php` » appelle `session_start` et crée une variable de session « pseudo » valant « tom ». Une seconde page « `profil.php` » affiche cette variable.
 - Que se passe-t-il lorsque l'on supprime les cookies ?
 - Modifier le code de la page `profil.php` pour détecter l'absence de la variable de session et éviter le message d'alerte qui se produit.
 - Modifier la page `ouvrir_session.php` pour qu'elle redirige l'utilisateur directement vers la page `profil.php`.
4. Une page « `connexion.php` » contient un formulaire qui permet à l'utilisateur d'entrer lui-même un pseudo et un mot de passe et de l'envoyer à la page « `ouvrir_session.php` ».
 - En cas d'absence du pseudo ou du mot de passe, on évitera les alertes en déclenchant une redirection vers la page « `connexion.php` » tout en envoyant un message « Pseudo & Passe ne doivent pas être vides => Vérifiez votre saisie ! ».
 - Comment encoder correctement ce message ?
 - Ce message doit s'afficher dans la page `connexion.php`. Le code affichant ce message ne doit pas provoquer d'erreur si aucun message n'est fourni.
 - Si pseudo et mot de passe sont fournis, ils doivent vérifier l'une des conditions suivantes pour correspondre à des identifiants corrects (cf. <http://php.net/manual/fr> pour vous documenter sur ces fonctions) :
 - Leur distance de levenshtein est inférieure à 2;
 - Leurs clés soundex sont identiques
 - Quand c'est le cas, la page `ouvrir_session.php` renvoie l'utilisateur vers la page `profil.php`;
 - Quand ce n'est le cas, la page `ouvrir_session.php` renvoie l'utilisateur vers la page `connexion.php` en renvoyant le message « Mot de passe incorrect »;
 - Quelle URL mettre en favoris pour se logger automatiquement avec le pseudo « `slideman` » ?
5. On souhaite permettre à l'utilisateur de se déconnecter. Proposer une solution dans laquelle la déconnexion se fait en cliquant sur un lien, de sorte que l'utilisateur une fois déconnecté revienne à la page de connexion qui affiche alors le message « A Bientôt ».