

TP PHP : Sessions, Fichiers, XML

L'objectif est de d velopper pas   pas diff rents scripts permettant d'inscrire des "employ s" et d'en afficher la liste. Cette liste sera dans un premier temps cr  e et stock  e sous forme de fichier au format CSV. Vous d velopperez ensuite un fichier au format XML. L'archive   t l charger de l'espace Moodle comprend diff rents fichiers que vous aurez   compl ter ou utiliser :

- `accueil.php` : la page d'accueil.
- `employee_form.php` : la page d'inscription.
- `employees.dtd` : une DTD pour fichier XML d'employ s.

Exercice 1 - Compl tez le script `accueil.php` de sorte que l'hyperlien redirige vers le script `employee_form.php`.



Exercice 2 - Le script `employee_form.php` est un formulaire permettant d'inscrire un nouvel employ  (Figure 1) ou de revenir   la page d'accueil. Chaque inscription met imm diatement la page   jour en affichant la liste des employ s sous forme de tableau HTML (Figure 2).

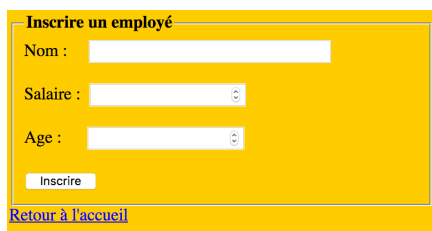
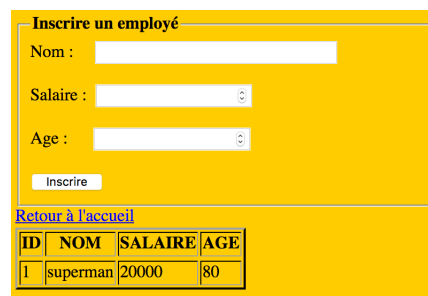


FIGURE 1 –



ID	NOM	SALAIRE	AGE
1	superman	20000	80

FIGURE 2 –

Toute inscription autoris  e est ajout  e dans le fichier `employees.csv` dont le format est le suivant (les sauts de ligne correspondent au caract  re `\n`) :

```
1;superman;20000;80
2;batman;10000;70
```

Afin de cr  er et mettre   jour ce fichier, utilisez les fonctions :

- `fopen` (en mode lecture ou ajout), `fclose` et `flock` pour verrouiller/déverrouiller l'accès au fichier.
- `fgetcsv` pour lire et décomposer chaque ligne du fichier.

Vous prendrez garde à l'inscription des employés dénommés :

```
<h1>My HTML</h1>
<script>alert('Gotcha!');window.history.back();</script>
```

Exercice 3 - On souhaite rejeter les doubles inscriptions en cours de “session” où deux individus seront considérés identiques s'ils ont le même nom et le même âge. Exemple d'avertissement :

Modifiez vos scripts en utilisant le mécanisme des sessions PHP afin de mémoriser chaque inscription effectuée et de rejeter toute nouvelle tentative (un véritable test consisterait à analyser le fichier à chaque tentative). Notez que Firefox permet de supprimer le cookie de session PHPSESSID créé par PHP (Outil de développement->Inspecteur de stockage->Cookies).

Exercice 4 - Développez un script `employees2xml.php` qui génère un fichier XML d'employés à partir du fichier CSV `employees.csv`. Le fichier XML devra être conforme à la DTD `employee.dtd`. Par exemple :

```
<?xml version='1.0' encoding='UTF-8'?>
<!DOCTYPE employees SYSTEM 'employees.dtd'>
<employees>
  <employee>
    <id>1</id>
    <name>superman</name>
    <salary>20000</salary>
    <age>80</age>
  </employee>
  <employee>
    <id>2</id>
    <name>batman</name>
    <salary>10000</salary>
    <age>70</age>
  </employee>
</employees>
```

Exercice 5 - Modifiez ensuite la page d'accueil de sorte que :

- le tableau des noms d'employés s'affiche s'il en existe (Figure 3),
- chaque nom soit un hyperlien qui, par invocation du script `get_employee.php`, permet de récupérer toutes les informations sur cet employé (Figure 4).



employee: id=2 name=batman salary=10000 age=70

FIGURE 4 –

FIGURE 3 –

Le script `get_employee.php` doit lire et extraire du fichier XML les informations relatives à l'employé demandé. Utilisez l'API `SimpleXML` ou l'API `DOM` avec requête `XPath` pour rechercher les bons éléments.

Exercice 6 - Créez un formulaire de saisie des deux codes couleur préférés du visiteur du site pour la couleur de fond et le texte de la page. Les `codes CSS` pourront être donnés sous forme symbolique, hexadécimale ou RGB. Enregistrez les dans deux cookies valables 20 secondes et appliquez le style demandé à l'ensemble de la page. Testez les cookies par ré-envoi avant et après délai d'expiration.

