

Programmation Web

CSV ⇔ MySQL

Enoncé

Vous devez compléter les fonctions manquantes d'un outil d'import/export de fichiers CSV. Cet outil permet d'une part d'importer des données en provenance d'un fichier structuré de type CSV et de les rajouter dans une DB MySQL, et d'autre part d'exporter certaines données de la DB vers un fichier structuré de type CSV.

Interfaces graphiques

1) Interface vierge de données

Contrôle CSV<=>MySQL

localhost:8888/ig1/exercices/controle/controle.php

Importation : CSV 2 MySQL

Fichier : Choisir le fichier aucun fichier sél. Envoyer le fichier

Exportation : MySQL 2 CSV

Prenom	Nom	Adresse	Telephone	Exporter
--------	-----	---------	-----------	----------

Exporter au format CSV

2) Upload d'un fichier CSV

Contrôle CSV<=>MySQL

localhost:8888/ig1/exercices/controle/controle.php

Importation : CSV 2 MySQL

Fichier : Choisir le fichier aucun fichier sél. Envoyer le fichier

1 2

Exportation : MySQL 2 CSV

Prenom	Nom	Adresse	Telephone	Exporter
--------	-----	---------	-----------	----------

Exporter au format CSV

3

Selectionnez les lignes à ajouter dans la DB

Prénom	Nom	Adresse	Téléphone	Importer
Jean	Nemarre	12 rue des bidules	02344444	<input type="checkbox"/>
Nicolas	Sticot	1 rue du spa	54545	<input type="checkbox"/>
Albert	Deux	1 rue royale	222	<input type="checkbox"/>
Anna	Conda	3 rue la	535	<input type="checkbox"/>

Insérer dans la DB

- 1 : on sélectionne le fichier CSV à uploader
- 2 : on envoie le fichier CSV
- 3 : si tout s'est bien passé alors un formulaire d'importation s'affiche dans le bas de la page

3) Ajout de personne(s) dans la DB

Contrôle CSV<=>MySQL

localhost:8888/ig1/exercices/controle/controle.php

Importation : CSV 2 MySQL

Fichier : aucun fichier sél.

Exportation : MySQL 2 CSV

Prenom	Nom	Adresse	Telephone	Exporter
Jean	Nemarre	12 rue des bidules	02344444	<input type="button" value="Exporter au format CSV"/>

Selectionnez les lignes à ajouter dans la DB

Prénom	Nom	Adresse	Téléphone	Importer
Jean	Nemarre	12 rue des bidules	02344444	<input checked="" type="checkbox"/>
Nicolas	Sticot	1 rue du spa	54545	<input type="checkbox"/>
Albert	Deux	1 rue royale	222	<input checked="" type="checkbox"/>
Anna	Conda	3 rue la	535	<input type="checkbox"/>

- 1 : on coche les personnes que l'on souhaite ajouter à la DB
- 2 : on valide l'insertion

4) Affichage après l'ajout de personne(s)

Contrôle CSV<=>MySQL

localhost:8888/ig1/exercices/controle/controle.php

Importation : CSV 2 MySQL

Fichier : aucun fichier sél.

Exportation : MySQL 2 CSV

Prenom	Nom	Adresse	Telephone	Exporter
Jean	Nemarre	12 rue des bidules	02344444	<input type="button" value="Exporter au format CSV"/>
Albert	Deux	1 rue royale	222	<input type="button" value="Exporter au format CSV"/>

Exportation réussie

- 1 : un message sur l'état de l'importation s'affiche
- 2 : les données importées sont affichées et prêtes à être éventuellement exportées

5) Exportation de personne(s) vers un fichier CSV

Contrôle CSV<=>MySQL

localhost:8888/ig1/exercices/controle/controle.php

Lecteur

Importation : CSV 2 MySQL

Fichier : Choisir le fichier aucun fichier sél. Envoyer le fichier

Exportation : MySQL 2 CSV

Prenom	Nom	Adresse	Telephone	Exporter
Jean	Nemarre	12 rue des bidules	02344444	<input checked="" type="checkbox"/>
Albert	Deux	1 rue royale	222	<input checked="" type="checkbox"/>

Exporter au format CSV

1 : on sélectionne les personnes à exporter dans un fichier CSV

2 : on valide l'exportation

6) Affichage après l'exportation de personne(s)

Contrôle CSV<=>MySQL

localhost:8888/ig1/exercices/controle/controle.php

Lecteur

Importation : CSV 2 MySQL

Fichier : Choisir le fichier aucun fichier sél. Envoyer le fichier

Exportation : MySQL 2 CSV

Prenom	Nom	Adresse	Telephone	Exporter
Jean	Nemarre	12 rue des bidules	02344444	<input type="checkbox"/>
Albert	Deux	1 rue royale	222	<input type="checkbox"/>

Exporter au format CSV

exportation réussie !

1 : un message sur l'état de l'exportation s'affiche et si tout s'est bien passé un fichier CSV a été créé.

Description des éléments du site

Les fichiers :

- **controle.php** : la page principale. Celle-ci vous est fournie et est incomplète. Vous devez compléter la partie « actions ».
- **fonctions_controle.php** : cette page est incomplète et doit contenir toutes les fonctions d'affichage et de modification de données.
- **connexion_pdo.php** : script de connexion à la DB. A modifier selon votre configuration
- **annuaire.txt** : le fichier texte contenant les informations que l'on souhaite importer
- **annuaire.sql** : script SQL permettant de créer la table nécessaire pour le contrôle.

1) *formulaireImportation(\$_nom_fichier,\$_pageEnCours)*

Cette fonction renvoie un formulaire permettant de sélectionner les personnes que l'on souhaite importer dans la DB. Elle sera appelée juste après l'upload du fichier CSV.

Les paramètres d'entrée sont :

- `$_nom_fichier` (type string) : nom du fichier CSV à manipuler
- `$_pageEnCours` (type string) : la page PHP en cours d'exécution

Le paramètre de sortie est de type string et contient le formulaire d'importation.

Ce formulaire contient une table HTML construite à partir des données en provenance du fichier CSV uploadé précédemment. Il faudra donc ouvrir le fichier spécifié en paramètre, extraire les données CSV à l'aide de la méthode « *fgetcsv()* » et les intégrer dans la table HTML.

Chaque ligne de la table devra contenir une checkbox permettant de sélectionner la personne à insérer dans la DB.

Afin d'éviter de relire le fichier CSV lors de l'importation, je vous impose une astuce qui a ses avantages et ses inconvénients mais qui simplifiera le code. Cette astuce consiste à faire en sorte que l'attribut « *value* » de chaque checkbox contienne toutes les données de la personne correspondante (nom, prénom, adresse et téléphone). Pour ce faire, nous allons utiliser un nouvel outil : la sérialisation. Cela consiste à regrouper plusieurs variables, un tableau ou autre sous la forme d'une chaîne de caractère. Ex : si *\$data* contient le résultat d'une lecture CSV et est de type array :

```
$data[0] = 'fab'
$data[1] = 'med'
$data[2] = '12 rue en haut'
$data[3] = '025543423'
```

on pourra écrire : *\$serial_personne* = *serialize(\$data)*.

\$serial_personne sera donc de type string et contiendra le nom, prénom, adresse et téléphone de manière structurée. Pour récupérer les éléments comme ils l'étaient avant la sérialisation il suffira d'écrire : *\$data* = *unserialize(\$serial_personne)* (à utiliser dans la méthode « *importer()* »). *\$data* sera un tableau (array) avec la même structure et le même contenu que précédemment.

Il faudra également penser à transformer les éventuels caractères spéciaux présents dans *\$data* en vue de les envoyer via un formulaire web. Pour cela nous allons travailler avec la fonction *urlencode* pour l'envoi et *urldecode* pour la réception.

Donc pour vous aider dans la construction du formulaire, vous aurez quelque chose dans le style :

```
$serial_personne = serialize($data) ;
$checkbox = '<input type="checkbox" name="???" value="'.urlencode($serial_personne).'" />';
```

Il manque volontairement des éléments dans *\$checkbox*, à vous de les rajouter en fonction des besoins de la page « *controle.php* ».

Lorsque le formulaire sera validé, la page recevra ainsi directement les données à insérer et il ne sera plus nécessaire de parcourir à nouveau le fichier CSV.

Lors du traitement, il faudra effectuer le décodage de cette manière-là, par exemple :

```
$personne = unserialize(urldecode($serial_personne));
```

N'oubliez pas les champs cachés du formulaire (voir les besoins de la page « *controle.php* »).

2) *importer(\$_tableau_personnes,\$_dbh)*

Cette fonction permet d'insérer de nouvelles données de personnes en provenance du formulaire d'importation vers la base de données.

Elle sera appelée juste après la validation du formulaire d'importation.

Les paramètres d'entrée sont :

- `$_tableau_personnes` (type array) : tableau contenant des chaînes de caractères de personnes sérialisées (voir « [formulaireImportation\(\)](#) »)
- `$_dbh` (type handle) : objet de connexion à la base de donnée

Le paramètre de sortie est de type string et contient le message d'erreur ou de réussite de l'importation.

A travers cette fonction vous devez insérer dans la table « *annuaire* » de votre DB, les personnes cochées dans le formulaire d'importation. N'oubliez pas que les personnes ont été sérialisées dans la fonction précédente donc il faudra désérialiser de cette manière là :

```
$personne = unserialize(urldecode($serial_personne)) ;
```

Vous ne pouvez utiliser que PDO (pas MySQL_) et vous devez travailler avec les requêtes paramétrées. Il ne peut pas y avoir de doublons dans la table (nom et prénom) !

3) `formulaireExportation($_dbh,$_pageEnCours)`

Cette fonction renvoie un formulaire permettant de sélectionner les personnes que l'on souhaite exporter dans un fichier CSV.

Les paramètres d'entrée sont :

- `$_dbh` (type handle) : objet de connexion à la DB
- `$_pageEnCours` (type string) : la page PHP en cours d'exécution

Le paramètre de sortie est de type string et retourne le formulaire d'exportation.

Ce formulaire affiche les données des personnes présentes dans la base de données.

Comme vous pouvez le voir sur l'interface graphique 5 la sélection de personnes à exporter se fait encore à l'aide de checkbox. Ici libre à vous de déterminer comment vous allez faire pour transmettre les informations des personnes sélectionnées vers la fonction « [exporter\(\)](#) » .

Tuyau :

- soit transmettre les « *id* » des personnes à exporter et dans la fonction « [exporter\(\)](#) » il faudra alors faire des requêtes SQL pour récupérer les données des personnes à exporter vers CSV
- soit transmettre toutes les informations des personnes de la même manière que cela a été fait pour l'importation (sérialisation)

N'oubliez pas les champs cachés du formulaire en fonction des besoins de la page « [contrôle.php](#) ».

4) `exporter($personnes,$_dbh,$_nom_fichier_csv)`

Cette fonction permet d'exporter les données envoyées par le formulaire d'exportation vers un fichier CSV. Elle sera appelée juste après la validation du formulaire d'exportation.

Les paramètres d'entrée sont :

- `$personnes` (type array) : tableau contenant les informations des personnes à exporter
- `$_dbh` (type handle) : objet de connexion à la DB
- `$_nom_fichier_csv` (type string) : le nom du fichier qu'il faudra créer pour y sauver les enregistrements.

Le paramètre de sortie est de type string et contient le message d'erreur ou de réussite de l'exportation.

A travers cette fonction, vous devez créer ou écraser le fichier dont le nom est défini en paramètre d'entrée et y sauvegarder, sous le format CSV, les données spécifiées dans le paramètre d'entrée

\$_personnes.

Pour construire votre fichier CSV et afin de vous faciliter le travail, vous devrez utiliser la fonction « *fputcsv()* ».

Ce que vous devez compléter dans le fichier *fonctions_controle.php* :

1) Partie traitement des actions

Vous devez mettre en place le code nécessaire à la gestion des différentes actions possibles et aux affichages correspondant.

Contraintes

Veillez à produire une bonne structure, utilisez des commentaires, aérez votre code, utilisez l'indentation et choisissez des noms de variables appropriés ! Il y a moins de 100 lignes de code à écrire !

Commencez par la partie importation (le formulaire et la sauvegarde sur DB) et ensuite l'exportation.

Vous ne pouvez utiliser que les ressources du cours disponibles sur ecampus ainsi que la documentation en ligne à cette adresse : <https://www.php.net/manual/fr/index.php>.

Bon Travail !