

Feuille de travaux pratiques

Série 4 : Les CRUD avec des fichiers JSON

Exercice 1 :

L'objectif de cette série est d'implémenter les opérations **CRUD** (*Create/Read/Update/Delete*) pour la gestion des modules en utilisant un fichier **JSON** *data/modules.json*. Ce dernier aura le format suivant :

```
[
  {
    "code": "XLG5IE030",
    "nom": "Comptabilité financière",
    "responsable": "C. Suaudeau",
    "email": "christophe.suaudeau@univ-nantes.fr",
    "CM": "0h",
    "TD": "44h",
    "TP": "0h"
  },
  {
    "code": "XLG5IE020",
    "nom": "Ingénierie des exigences",
    "responsable": "G. Jean",
    "email": "geraldine.jean@univ-nantes.fr",
    "CM": "13h20",
    "TD": "22h40",
    "TP": "8h"
  },
  {
    "code": "XLG5IE060",
    "nom": "BD et Web",
    "responsable": "L. Monceaux",
    "email": "laura.monceaux@univ-nantes.fr",
    "CM": "0h",
    "TD": "24h",
    "TP": "16h"
  }
]
```

- 1) Modifier la fonction **getModules** (définie dans la série N° 2) pour pouvoir importer les modules à partir du fichier JSON. On supposera que le chemin du fichier JSON est une **constante** définie dans le fichier *inc/config.php*.

Le script utilisant la méthode **getModules** sera appelé *index.php* pour la suite de la série.

Indications :

- Pour lire un fichier, on peut utiliser la fonction *file_get_contents*.
- Pour décoder une chaîne de caractères en format JSON, on peut utiliser la fonction *json_decode*.
- Étant donné que chaque module est défini dans le fichier JSON comme un objet, la valeur retournée par la fonction est un tableau d'objets. Il faut adapter la vue *index.view.php* pour pouvoir afficher ces objets.

2) Écrire un script ***show.php*** qui affiche les détails d'un module dont le code est transmis dans le tableau ***\$_GET***. Ce script fera appel à une fonction ***getModule*** permettant de parcourir les modules stockés dans le fichier JSON à la recherche du module dont le code correspond à son paramètre.

3) Écrire un script ***search.php*** permettant de chercher un module en utilisant un mot clé.

Nous souhaitons réserver les autres opérations CRUD à l'administrateur de l'application. Pour faciliter l'organisation des dossiers/fichiers de l'application, nous mettrons les scripts associés à l'administration dans un dossier spécial appelé ***admin***.

4) recopier le script ***index.php*** dans le dossier ***admin***, et la vue ***views/index.view.php*** dans le dossier ***views/admin/***. Modifier le script ***admin/index.php*** pour utiliser la vue ***views/admin/index.php***.

le script ***admin/index.php*** fonctionne il ?

5) Le problème rencontré dans la question 4) est dû à l'utilisation du **chemin relatif** du dossier ***views*** dans l'affichage des vues. Remplacer ce chemin relatif par un **chemin absolu**.

Indication :

Les chemins absolus doivent être calculés par l'application et non pas écrits en dur. Pour ce faire, vous pouvez utiliser ***dirname(__FILE__)*** pour récupérer le chemin absolu du dossier racine de l'application. Ce chemin peut être calculé et stocké dans une constante dans le fichier ***inc/config.php***.

6) Écrire un script ***admin/create.php*** permettant de (i) saisir les informations d'un nouveau module et (ii) stocker le nouveau module dans le fichier JSON ***data/modules.json***.

7) Écrire un script ***admin/edit.php*** permettant de modifier les informations d'un module.

Indication :

Un module est identifié par son code. Ce dernier peut aussi être modifié par l'utilisateur lors de l'édition du module. Il faut alors garder en mémoire l'ancien code du module. Pour ce faire, on peut utiliser un composant de type ***hidden*** dans le formulaire de l'édition.

8) Écrire un script ***admin/delete.php*** permettant de supprimer un module.

Indication :

Pour supprimer un élément d'un tableau ***\$tab***, on peut utiliser la fonction ***unset*** en lui passant en paramètre ***\$tab[\$i]*** avec ***\$i*** étant l'indice de l'élément en question.

9) Mettre en place un **système d'authentification** pour sécuriser les opérations d'administration (création, édition et suppression) de l'application. Vous pouvez vous inspirer du dernier exercice de la série N°3.