

Architecture et dev web

TP 9 - PHP Objet

I. Finir le TP8 si ce n'est pas fait

Vous disposez sous moodle de la correction qui permet de faire afficher l'ensemble des villes sur la page d'accueil.

Finissez le code qui permet de faire afficher les caractéristiques d'une seule ville lorsque l'url est la suivante : `index.php?id=2`

II. Sauvegarde de données

II.1. Version 1

Nous souhaitons sauvegarder des données si l'url demandée est la suivante :

`index.php?action=save&nom=chiang%20rai&pop=69888&lat=19.90859&long=99.8325&paysId=2`

Mettez en place l'architecture MVC permettant de répondre à ce problème

II.2. Version 2

Nous souhaitons avoir un formulaire permettant d'ajouter des villes si l'url demandée est la suivante : `index.php?action=save`

Mettez en place l'architecture MVC permettant de répondre à ce problème.

III. Création d'une API

Pour l'instant vous avez affiché le contenu de la base de données sur la page d'index selon l'url souhaitée. Nous souhaiterons avoir une API pour que vos données soient utilisables par d'autres personnes comme une api de type opendata (revoir l'opendata de paris par exemple)

Si l'url est la suivante : `index.php?controller=api`

alors la réponse de votre page sera uniquement des données au format json.

Cette url, vous amènera sur un nouveau fichier `indexApi.php` (à la racine du site) en utilisant la fonction `header()`. Dans ce fichier vous créerez un controller, et appellerez la vue renvoyant le JSON.

Comme ceci :

```
{
  "nhits": 4,
  "format": "json",
  "records": [
    {
      "datasetid": "villeVoyage",
      "fields": {
        "nom": "negombo",
        "pop": 121701,
        "pays": "Sri lanka",
        "geo": [
          7.2083,
          79.8358
        ]
      }
    },
    {
      "datasetid": "villeVoyage",
      "fields": {
        "nom": "Anuradhapura",
        "pop": 50595,
        "pays": "Sri lanka",
        "geo": [
          8.31223,
          80.41306
        ]
      }
    },
    ...
  ]
}
```

Créez le fichier indexApi.php (attention cette index n'aura pas de doctype)

Créez un sous-dossier « api » dans le dossier controller

Créez une classe ControllerVilleApi dans le dossier api

créez comme le ControllerVille une méthode getAllApi()

Codez la méthode getAllApi() qui devra appeler le model pour récupérer les données ainsi que le nombre de hit (nombre de données dans la base) puis appeler la vue renvoyant les données JSON.

Maintenant il est nécessaire de créer la vue. Cette vue aura les lignes suivantes en début de fichier :

```
header("Access-Control-Allow-Origin: *");  
header("Content-Type: application/json; charset=UTF-8");
```

Elle devra renvoyer une réponse 200 pour signifier que les données sont correctement renvoyées

```
http_response_code(200),
```

Et finir par l'affichage des données en json

```
echo json_encode($data);
```

Le tableau \$data aura été créé selon le modèle fourni page précédente

Créez cette vue dans un fichier viewApi.php que vous enregistrerez dans un sous-dossier de view

Vous allez à présent tester votre API. Le dossier fourni « testApi » permet de créer une map leaflet et d'ajouter sur cette map les données issues d'un fichier JSON. Il s'agit bien sûr d'un fichier de test. Vous utiliserez dans un second temps votre API

Récupérez sous moodle le dossier testApi et testez son fonctionnement.

Remplacer l'appel au fichier test.json par l'url de votre Api (Attention vous devez mettre votre projet sur un serveur) si vous n'en avez pas vous pouvez utiliser l'url ci dessous :

```
https://abourmau.lpmiaw.univ-lr.fr/archiWeb2/ville/indexApi.php
```

Comment pouvez vous mieux centrer votre carte ? Proposez une solution. Pensez à tester le niveau de zoom <https://leafletjs.com/>.