

Travaux pratiques LbD4

Pr. Ahmed Bentajer

TP 7 : JS XML et AJAX

Exercice 1 :

Soit le fichier **déjeuner.xml**. Écrire le code XSL permettant d'afficher d'une manière organisée les différents menus.

Exercice 2 :

Soit le fichier **books.xml** :

1. Écrire le code XSL permettant d'afficher les livres d'une manière lisible pour l'utilisateur final
2. Proposer différentes Template pour afficher les livres par catégorie
3. Améliorer la Template de la question précédente pour afficher les livre avec prix croissant

Exercice 3 :

Vous devez créer une page web qui affiche les informations météorologiques pour une ville donnée. Pour cela, vous utiliserez l'API de **OpenWeatherMap** pour récupérer les données météorologiques pour une ville donnée en utilisant **XMLHttpRequest**.

Voici les étapes à suivre :

- Créez une page web avec un formulaire permettant à l'utilisateur de saisir le nom d'une ville.
- Lorsque l'utilisateur soumet le formulaire, utilisez XMLHttpRequest pour récupérer les données météorologiques pour la ville saisie.
- Affichez les données météorologiques dans la page web. Vous pouvez afficher les informations telles que la température, la description du temps, la vitesse du vent, etc.

N.B. : Pour que les requêtes puissent fonctionner, vous aurez besoin d'une clef API que vous pouvez avoir en vous inscrivant sur : https://home.openweathermap.org/users/sign_up

Exercice 4 :

Supposant qu'on a une base de données avec la table Utilisateurs qui contient un champs pseudo.

Écrire le code adéquat permettant trouver si un pseudo est libre ou pas lors de l'inscription sur le site.

N.B. : Le nombre de caractère minimum requis pour le pseudo est 5

Exercice 5 :

En utilisant la base de données World. Écrire le code permettant d'afficher la liste des pays et une fois que l'utilisateur aie choisi un pays le programme doit lui afficher les informations relatives du pays.

Exercice 6

En utilisant une de ces API :

- lightStats API (<https://developer.flightstats.com/>) : cette API fournit des informations en temps réel sur les vols, les horaires, les aéroports et les compagnies aériennes.
- OpenSky Network (<https://opensky-network.org/>) : cette API fournit des informations en temps réel sur les vols, les positions des avions, les aéroports et les compagnies aériennes.
- AviationStack (<https://aviationstack.com/>) : cette API fournit des informations sur les vols, les horaires, les aéroports et les compagnies aériennes.
- Skyscanner API (<https://rapidapi.com/skyscanner/api/skyscanner-flight-search/>) : cette API fournit des informations sur les vols, les horaires, les aéroports et les compagnies aériennes.

Proposer un projet permettant à un visiteur de votre application de chercher des vols selon les critères proposées par l'API.

Exercice 7 : Pour aller plus loin

Réaliser une page Web permettant d'afficher des images en fonction des mots-clés saisis par l'utilisateur. Pour récupérer l'URL des images, utiliser une requête AJAX vers l'API de Flickr ou de Giphy.

Les images doivent être affichées directement dans la page à l'aide d'éléments ``, après la saisie de l'utilisateur.

Exemples d'API interrogeables via AJAX

APIs simples:

- Adresse IP: <httpbin.org/ip>
- Heure: time.jsontest.com (*HTTP Only*)
- Utilisateur: jsonplaceholder.typicode.com/users/1

APIs plus complexes:

- Animations: [Giphy](https://giphy.com)
- Photos: [Flickr](https://www.flickr.com/photos/)
- Météo: [openweathermap \(doc\)](https://openweathermap.org/doc) (*HTTP Only*)