



Introduction AJAX avec PHP

Le PoleS
2024

1. Qu'est-ce qu'AJAX ?

Définition

- AJAX (Asynchronous JavaScript and XML) est une méthode permettant de mettre à jour des parties d'une page web sans la recharger complètement. Il repose sur des technologies telles que HTML, CSS, JavaScript, Fetch API, et un serveur (PHP) pour traiter les données.

Objectifs principaux :

- Améliorer l'expérience utilisateur avec des pages interactives.
- Réduire les temps de chargement et l'utilisation du réseau.
- Créer des applications modernes comme les questionnaires de tâches ou les sites e-commerce.

2. Fonctionnement d'AJAX avec Fetch API

1. L'utilisateur interagit avec la page (exemple : clique sur un bouton).
2. Fetch API envoie une requête HTTP au serveur (PHP).
3. Le serveur traite la requête et renvoie une réponse (JSON, texte, ou HTML).
4. JavaScript met à jour dynamiquement la page avec la réponse.

3. Pourquoi Fetch API ?

- Fetch API est la méthode moderne pour effectuer des requêtes HTTP en JavaScript.
- Plus simple et lisible grâce à l'utilisation des Promises et d'async/await.
- Meilleure gestion des erreurs comparée à XMLHttpRequest.

3. JSON : le format le plus courant

Pourquoi JSON est préféré à XML ?

- JSON (JavaScript Object Notation) est léger et facile à lire/écrire.
- Utilisé nativement en JavaScript, donc facilement manipulable.
- XML est plus verbeux (balises d'ouverture/fermeture) et nécessite des bibliothèques supplémentaires pour le traitement.

Exemple json:

```
{  
  "nom": "Alisson",  
  "age": 25,  
  "profession": "Développeuse"  
}
```

Exemple XML:

```
<personne>
  <nom>Alisson</nom>
  <age>25</age>
  <profession>Développeuse</profession>
</personne>
```

Pourquoi JSON est plus avantageux que XML ?

1. Taille réduite :

- JSON est plus compact, car il n'a pas de balises d'ouverture et de fermeture. Dans XML, chaque donnée nécessite une balise supplémentaire, ce qui augmente la taille des fichiers.

2. Lisibilité :

- JSON est plus facile à lire pour les humains grâce à sa structure simple et familière (des objets et des tableaux JavaScript).

3. Performance :

- JSON est nativement pris en charge dans JavaScript avec `JSON.parse()` et `JSON.stringify()`, ce qui le rend plus rapide à manipuler dans un environnement web. Avec XML, il faut souvent utiliser un parseur comme `DOMParser` ou des bibliothèques externes.

4. Interopérabilité :

- JSON est le format standard pour les API modernes et les échanges de données entre clients et serveurs.

5. Simplicité :

- JSON est facilement intégrable dans les objets JavaScript, contrairement à XML qui demande une étape de conversion en un format manipulable.

Quand utiliser XML ?

Malgré ces inconvénients, XML peut être utile dans certains cas :

- Lorsqu'on a besoin d'une structure strictement définie avec validation (ex. : XML Schema).
- Pour représenter des documents complexes (comme des documents Word ou des données géospatiales).
- Lorsque des systèmes anciens ou des standards spécifiques nécessitent XML.