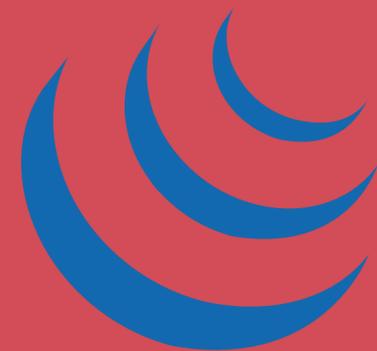
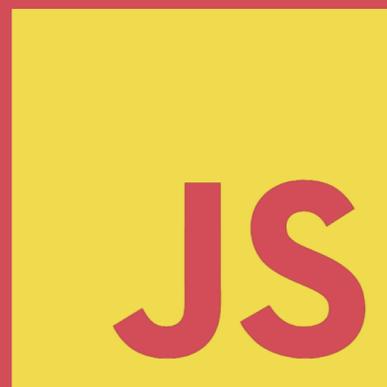


`simplonBoulogne.Javascript();`

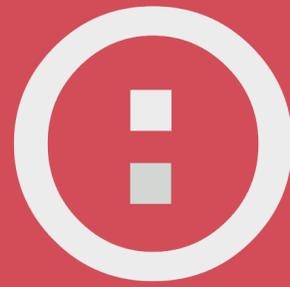
Introduction à Ajax





Ce que l'on va voir

- Concept (Exposé Jérémie, David, Raph)
- Utilisation en Javascript (Moi)
- Utilisation avec jQuery (Manu)



Présentation

AJAX = Asynchronous JavaScript and XML.

Ajax permet de d'afficher de nouvelles informations sur votre page SANS recharger celle ci.

Vous pouvez également envoyer / recevoir des données du serveur une fois que la page s'est chargé, et ce, en arrière plan.



Étapes avec Javascript

- Créer un objet XML HTTP Request
- Créer la fonction à exécuter lors d'une réponse
- Ouvrir la requête
- Envoyer la requête



Fonctionnement

NAVIGATEUR

Créer et envoi un objet
XMLHttpRequest

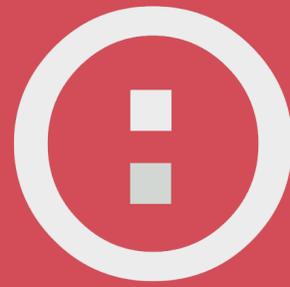
Traitement des données et affichage
grâce à Javascript/DOM



SERVEUR

Traitement

Envoi d'une réponse



Créer un objet XMLHttpRequest

XMLHttpRequest est un standard web supporté par tous les navigateurs modernes (Chrome, Firefox, Safari, Opéra etc...)

```
xhr = new XMLHttpRequest();
```



Utilisation avec - IE 7 (...)

```
var xmlhttp;  
if (window.XMLHttpRequest) {  
    xmlhttp = new XMLHttpRequest();  
} else {  
    // code for IE6, IE5  
    xmlhttp = new ActiveXObject("Microsoft.XMLHTTP");  
}
```



Création d'une fonction

```
xhr = new XMLHttpRequest();  
xhr.onreadystatechange = function() {  
    //Fonction à executer lors d'un changement  
    //de statut pendant la procédure AJAX  
};
```

ouverture d'une requête, envoi, réception d'une donnée..

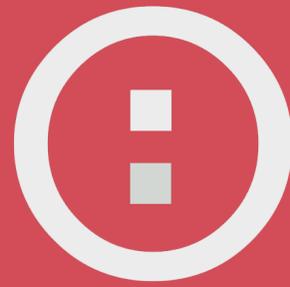


Vérifier l'état

Mais comment savoir quel changement de état/statut il y a eu précisément ?

```
if(xhr.readyState === 4) { }
```

0	1	2	3	4
Requête non initialisée	Connexion au serveur établie	Requête reçue	Traitement de la requête	Requête terminée et réponse prête



Vérifier le statut

Mais comment savoir quel changement de statut il y a eu précisément ?

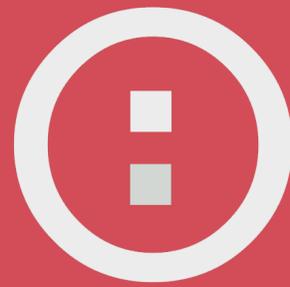
```
if(xhr.status == 200) { }
```

200

404

OK

Page not found



Premier aperçu

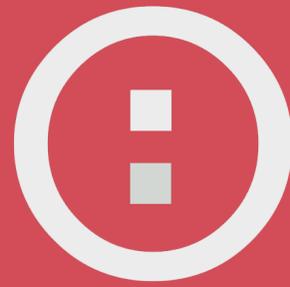
```
xhr = new XMLHttpRequest();  
xhr.onreadystatechange = function() {  
    if(xhr.readyState == 4 && xhr.status == 200) {  
        //Réponse serveur  
    }  
};
```



Récupérer la réponse serveur

Mais comment savoir quel changement de état/statut il y a eu précisément ?

```
document.getElementById('reponse').innerHTML =  
xhr.responseText;
```



Ouvrir la requête en POST

Cette fonction prépare le navigateur pour envoyer la requête

```
xhr.open('POST', 'traitement.php', true)
```

Méthode HTTP
utilisée
POST, GET...

Fichier sur le serveur

Asynchrone (true)
Synchrone (false)



Ouvrir la requête en GET

Cette fonction prépare le navigateur pour envoyer la requête

```
xhr.open('GET', 'traitement.php?var1='param, true)
```

Méthode HTTP
utilisée
POST, GET...

Fichier sur le serveur
+ envoi des param dans l'URL

Asynchrone (true)
Synchrone (false)



Envoyer la requête en POST

```
params="var1=".id_client;  
xhr.setRequestHeader("Content-Type", « application/x-www-form-  
urlencoded »);  
xhr.send(params);
```

Vous devez préparer vos paramètres dans une variable et l'envoyer avec `xhr.send()`



Envoyer la requête en GET

C'est tout, car les paramètres sont déjà placés dans l'url serveur

```
xhr.send();
```



Challenge number one

Créer une base de donnée contenant des informations sur un client (nom, prénom, âge, profession, email, téléphone etc...)

Créer une liste déroulante avec les noms des clients de la base et lorsque que vous sélectionnez un client, ses informations apparaissent plus bas avec AJAX.

Github plz.