
TP1 : AJAX

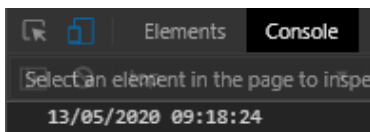
Ayoub KARINE (ayoub.karine@isen-ouest.yncrea.fr)

1. Dans la WSL, démarrez le serveur apache
2. Dans la WSL, déplacez-vous dans /var/www/html, et puis tapez :

`code` .
3. Créez un dossier ComWeb/TP1 dans le serveur apache

Exercice1 : Récupération d'une réponse sous forme de texte

1. Téléchargez le dossier [tp1ex1Input](#)
2. Appelez le fichier "ajax.js" dans "index.html"
3. Écrivez une fonction en JS : `function ajaxRequest(type, url)` qui crée une requête AJAX de type "type" sur l'url "url". Une fois la réponse du serveur obtenue, la réponse de la requête doit être affichée dans la console du navigateur (inspirez-vous de l'exemple du cours)
4. Découvrez le code du fichier php/timestamp.php puis affichez son contenu dans le navigateur
5. Dans le fichier « ajax.js », testez la fonction "ajaxRequest(type,url)" en envoyant une requête de type GET au fichier "timestamp.php" et vérifiez si l'heure s'affiche dans la console du navigateur. L'affichage dans la console doit être comme suit :



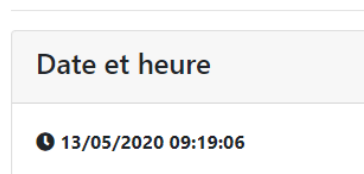
Requête AJAX et affichage statique dans la page HTML :

6. Ajoutez un troisième argument à la fonction ajaxRequest :
`ajaxRequest(type, url, callback)`. Ceci doit permettre de créer une requête AJAX de type "type" sur l'url "url". Une fois la réponse du serveur est

obtenue, la réponse de la requête doit appeler une fonction **callback** ([Rappel sur les callback](#)).

7. Créez une fonction **displayTimestamp(response)** qui injecte l'argument "response" dans l'élément HTML <section> d'id "timestamp" du fichier « index.html » (icône de l'horloge : <https://fontawesome.com/icons/clock>, CDN à récupérer depuis : <https://cdnjs.com/libraries/font-awesome>)
8. Testez la fonction "ajaxRequest" en envoyant une requête de type GET au fichier timestamp.php tout en utilisant la fonction "displayTimestamp" comme callback et vérifiez si l'heure s'affiche dans la page index.html. L'affichage dans la page index.html doit être comme suit :

AJAX



Requête AJAX et affichage dynamique dans la page HTML :

9. Appelez la fonction **ajaxRequest(type, url, callback)** chaque seconde et vérifiez si l'heure se modifie dynamiquement (se met à jour après chaque seconde) dans la page index.html. Pour ce faire, utilisez la fonction [setInterval](#).

Traitement des erreurs:

10. Lorsqu'une erreur survient lors de la requête AJAX, l'utilisateur doit en être informé. Pour ce faire, créez une fonction **httpErrors(errorCode)** qui va être appelée en cas d'erreur. Cette fonction va afficher le message d'erreur dans l'élément <section> du fichier index.html qui a l'ID "errors". Pour ce faire :
 - a. Changez le "display" de "none" à "block"
 - b. Affichez le message d'erreur. Voici quelques exemples
400: 'Requête incorrecte', 401: 'Authentifiez vous', 403: 'Accès refusé', 404: 'Page non trouvée', 500: 'Erreur interne du serveur', 503: 'Service indisponible'
11. Testez la fonction en envoyant une requête de type GET au fichier timestamp.php et vérifiez si le message d'erreur va s'afficher dans la page index.html. L'affichage dans la page index.html doit être comme suit (le

point d'exclamation entouré :

<https://fontawesome.com/icons/exclamation-circle>) :

AJAX

Page non trouvée

Date et heure

Exercice2 : Récupération d'une réponse JSON

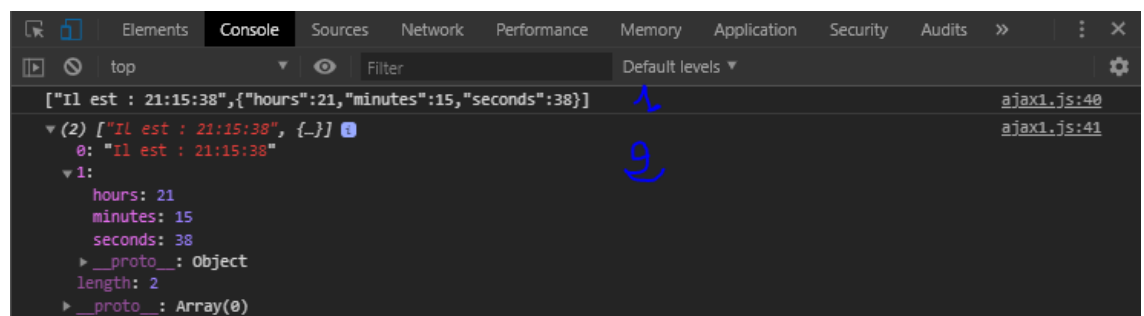
1. Téléchargez le dossier [tp1ex2Input](#)
2. Installez l'extension JSON viewer sur Chrome ou Firefox

Encodage, par le serveur, de la donnée à envoyer au client en utilisant PHP :

3. Testez le fichier "time.php"

Décodage, côté client, de la donnée JSON en utilisant JavaScript :

5. Créez un fichier "ajax.js" dans le dossier js dans lequel il est demandé de :
 - a. Ecrire une fonction en JS : `function ajaxRequest(type, url)` qui crée une requête AJAX de type "type" sur l'url "url". Une fois la réponse du serveur obtenue, la réponse de la requête doit être affichée dans la console du navigateur de deux manières : texte brute et JSON décodé = tableau
 - b. Testez la fonction en envoyant une requête de type GET au fichier "time.php" et vérifiez si l'heure s'affiche dans la console du navigateur (texte brute (1) et JSON décodé = tableau (2)). L'affichage dans la console doit être comme suit :



- c. Ajoutez un troisième argument à la fonction `ajaxRequest` : `function ajaxRequest(type, url, callback)` qui crée une requête AJAX de type "type" sur l'url "url". Une fois la réponse du serveur obtenue, la réponse de la requête doit appeler une fonction `callback` ([Rappel sur les callback](#)). Cette fonction aura pour rôle de modifier le contenu du fichier "index.html", plus précisément de l'élément h4 d'id "title" et de l'élément p d'id "detail"
- d. Testez la fonction en envoyant une requête de type GET au fichier "time.php" et vérifiez si l'heure s'affiche dans la page index.html. L'affichage dans la page index.html doit être comme suit :

AJAX-JSON

Il est : 22:43:47

*** **Détail** ***

hours : 22

minutes : 43

seconds : 47

- e. Appelez la fonction `ajaxRequest(type, url, callback)` chaque seconde et vérifiez si l'heure se modifie dynamiquement dans la page index.html
 - f. Traitez les erreurs en cas d'échec de la requête GET
6. Dans le fichier "clock.js" du dossier tp1ex2Input/js, une fonction appelée "displayclock" permettant d'afficher une horloge en JS est déjà implémentée :
- g. Testez cette fonction, dans "clock.js", en envoyant une requête de type GET au fichier "time.php" et vérifiez si l'horloge s'affiche dans la page index.html. L'affichage dans la page index.html doit être comme

suit :
AJAX-JSON

Il est : 23:23:16

*** Détail ***

hours : 23
minutes : 23
seconds : 16



- h. appelez la fonction `ajaxRequest(type, url, callback)` chaque seconde et vérifiez si l'horloge se modifie dynamiquement dans la page `index.html`