

**Intervenants :** Jonathan Alcuta ([jonathan.alcuta@loria.fr](mailto:jonathan.alcuta@loria.fr)), Mira Bou Saleh ([mira.bou-saleh@univ-lorraine.fr](mailto:mira.bou-saleh@univ-lorraine.fr)), Gaël Guibon ([gael.guibon@univ-lorraine.fr](mailto:gael.guibon@univ-lorraine.fr))

L'objectif de ce TD est de vous permettre de prendre en main les requêtes asynchrones de javascript, le plus souvent nommées AJAX. Pour cela vous allez créer un tableau sommaire dont les éléments seront mis à jour par déclenchement d'événement, et non par rechargement de la page entière.

Il est important de noter que vous êtes libres de faire de l'AJAX comme vous le souhaitez :

- Soit par [XMLHttpRequest](#) (pour les dinosaures mais très pédagogique)
- Soit par [fonctions asynchrones](#) (pour du javascript moderne, une simplification de la syntaxe)

## Exercice 1 – Tableau et design

Dans cet exercice, vous allez préparer une page HTML pour afficher une liste de films. La liste sera affichée dans un tableau HTML (balises `<table>`). La page sera remplie dans l'exercice suivant en utilisant AJAX.

Le tableau HTML attendu ressemble à ceci :

Image	Nom	Note	Nombre de lignes
			Toutes ▼

Pour atteindre ce résultat vous devez vous concentrer dans un premier temps uniquement sur la partie HTML et CSS en suivant les étapes que voici :

(💡 Cela va de soi, mais vous devez avoir créé un dossier dans lequel vous avez mis vos fichiers HTML et CSS. Vous devez également ne pas avoir oublié de lier votre feuille de style CSS avec votre fichier HTML. 😊)

1. Créez une page « movies.html » contenant un tableau vide ayant les champs « Image », « Nom » et « Note ». Pour cela aidez-vous des balises `<th>`, qui servent à montrer les entêtes de tableau (*Table Header*, soit `th`).
2. Améliorez votre page en utilisant une feuille de style. Votre page doit respecter les critères suivants :
  - a. La police d'écriture doit être soit être Arial, soit Helvetica.
  - b. Les cellules `<th>` des tableaux doivent avoir un *padding* (traduisons le par rembourrage) de 10px, un texte aligné à gauche et de couleur blanche et un fond de couleur grise. Leur fond doit devenir couleur sardes (*teal*) quand on passe la souris dessus (à l'aide de [hover](#)) et vert clair (*lightgreen*) quand on clique dessus ([active](#)).
  - c. Les cellules `<td>` des tableaux doivent avoir une couleur de texte noire et un fond de couleur blanche. Elles doivent également avoir un *padding* gauche et droite de 10px.
  - d. Les bordures du tableau doivent être gris clair, avoir la propriété « collapse » et le tableau doit avoir une largeur de 800px.
  - e. La largeur de la colonne des images doit être 50px et le *padding* de ses cellules doit être nul.

## Exercice 2– Manipulation du tableau

La page doit interagir avec un script PHP situé à l'adresse <http://gguibon.fr/movies.php>. Le script renvoie un contenu JSON correspondant à une liste de films à intégrer à la page. Il est possible de transmettre des valeurs au script de sorte que le contenu JSON renvoyé soit différent :

- <http://gguibon.fr/movies.php?sort=name> : trie les films de la liste par ordre alphabétique.
- <http://gguibon.fr/movies.php?sort=rating> : trie les films par note croissante.
- <http://gguibon.fr/movies.php?sort=rating&inverted=true> : trie les films par note décroissante.
- <http://gguibon.fr/movies.php?sort=name&limit=5> : renvoie les 5 premiers films de la liste triée par ordre alphabétique.

D'autres combinaisons sont évidemment possibles.

1. Créez un script JavaScript permettant de récupérer le contenu JSON renvoyé par le script PHP et de remplir le tableau. Le résultat doit ressembler à l'image sur la droite.
2. Modifiez votre code de manière à permettre un tri par nom ou par note quand on clique sur le champ correspondant. Plus précisément, en cliquant sur un des champs, la page PHP « movies.php » doit être appelée avec les paramètres appropriés en utilisant AJAX.
3. Modifiez votre code de manière à ce que deux clics successifs sur le même champ permettent de trier le tableau par ordre décroissant du champ correspondant. Le premier clic doit engendrer un tri croissant, le deuxième un tri décroissant, le troisième un tri croissant, le quatrième un tri décroissant, etc.
4. Ajoutez une [liste déroulante](#) (balises `<select>` et `<option>`) à droite du tableau pour permettre de limiter (ou de ne pas limiter) le nombre de ses lignes à 5, 10, 15 ou 20. La sélection d'une limite ne doit pas déclencher d'action et ne doit être prise en compte que lors des tris suivants.

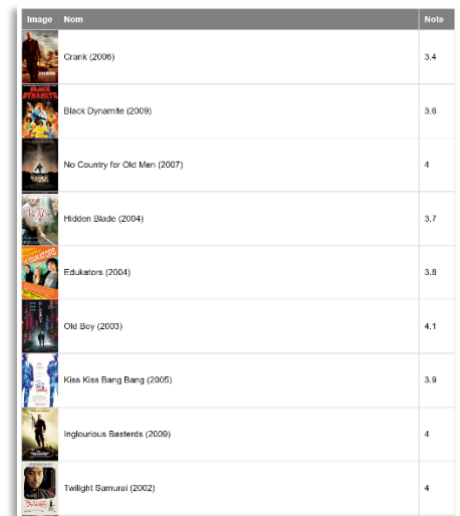








Image	Nom	Note
	Crank (2006)	3.4
	Black Dynamite (2009)	3.6
	No Country for Old Men (2007)	4
	Hidden Blade (2004)	3.7
	Educators (2004)	3.8
	Old Boy (2003)	4.1
	Kiss Kiss Bang Bang (2005)	3.9
	Inglourious Basterds (2009)	4
	Twilight Samurai (2002)	4

## Exercice 3 – Cache et bouton précédent

1. Modifiez votre code JavaScript de manière à sauvegarder le résultat de tous les appels AJAX engendrés par les clics de l'utilisateur et de les réutiliser directement si l'utilisateur fait à nouveau les mêmes actions.
2. Modifiez votre code de manière à permettre à l'utilisateur de revenir aux tris précédents en cliquant sur le bouton « Précédent » du navigateur. Pour cela, aidez-vous de l'objet **window** et de [onpopstate](#).