



JavaScript & Node.js

Exercices

1 Structurer son code en orienté-objet & en modules

Veillez créer une application web permettant d'enregistrer temporairement et localement des films et d'afficher les films enregistrés en dessous du formulaire.

Votre formulaire doit contenir les champs :

- **title** : titre du film
- **duration** : durée du film en minutes
- **budget** : pour informer du coût qu'a coûté la production du film, en millions
- **link** : pour donner un lien vers la description du film (lien vers imdb, rottentomatoes ou autre)

A chaque submit du formulaire, vous allez afficher la liste des films « enregistrés ». Voici un exemple d'affichage :

Title	Duration (min)	Budget (million)
Harry Potter and the Philosopher's Stone	152	125
Avengers: Endgame	181	181
...		

Veillez noter que le champs **link** est utilisé pour créer un lien dans la colonne **Title**.

Veillez valider votre formulaire.

Afin de réaliser cet exercice, voici les contraintes d'implémentation :

- Créez une classe **FilmLibrary** au sein du module **FilmLibrary.js**
- Créez une classe **Film** au sein du module **Film.js**
- Le module **FilmLibrary.js** et le module **Film.js** seront appelés par votre script d'entrée **index.js**
- La classe **FilmLibrary** doit fournir comme méthode :
 - o **addFilm(film)** : ajoute un film dans une structure de données, un **Array** d'instances de **Film** ;
c'est cette méthode qui doit être appelée par votre script d'entrée (**index.js**) pour « enregistrer » un film au sein d'une structure de données après chaque submit valide de formulaire ;
 - o **getHtmlTable()** : renvoie une string contenant la représentation HTML de la structure de données (un **Array** d'instances de **Film**) ;



JavaScript & Node.js

Exercices

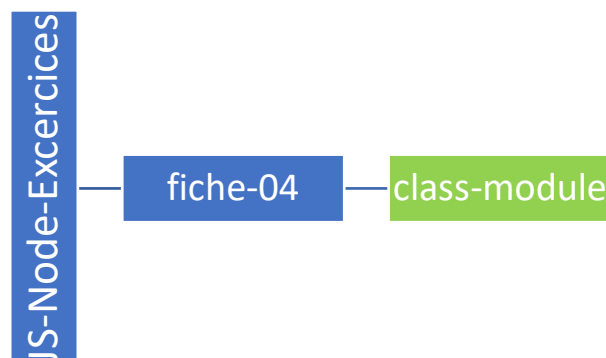
c'est cette méthode qui doit être appelée par votre script d'entrée (**index.js**) pour afficher la liste des films enregistrés, après chaque submit valide de formulaire ;

- Vous pouvez bien sûr repartir de la démonstration </demo/modules> pour vous inspirer de l'utilisation de modules.



- Comme vous l'auriez fait dans d'autres exercices, vous pouvez utiliser la propriété **.innerHTML** d'une div pour afficher dynamiquement la table.
- Dans le cadre de l'exercice « 2. Formulaire & HTML dynamique » de la fiche 02, vous avez créé une fonction qui renvoie sous forme de string une table HTML basée sur un **Array**. N'hésitez pas à vous en inspirer pour la méthode **getHtmlTable()**.

Le code de votre application web doit se retrouver dans ce dossier (en vert) de votre repository local et de votre web repository (**JS-Node-Exercices**).



2 Exercice optionnel :

JQuery

Si vous souhaitez vraiment découvrir **JQuery**, vous pourriez adapter l'exercice fait au §1 en remplaçant toutes les manipulations faites avec le DOM par des appels à **JQuery**.



JavaScript & Node.js

Exercices

Le code de votre application web devrait alors se retrouver dans ce dossier (en vert) de votre repository local et de votre web repository (**JS-Node-Exercices**).

