

Travail pratique #2 : JavaScript

Pondération : 15 %

Travail obligatoirement en équipe de 2 à moins d'une entente préalable avec le professeur

Remise : Au plus tard jeudi le 2 juin avant minuit

1 Objectif

Développer 2 fonctionnalités asynchrones en utilisant JavaScript.

2 Visionneuse

Vous devez réaliser une visionneuse dont les images seront celles des planètes visitées dans un film Star Wars. Des images sont fournies mais vous n'êtes pas obligé de les utiliser.

Vous devrez commencer par présenter une liste de quelques films de Star Wars (3 minimum) puis, lorsque l'utilisateur choisi un film, démarrer une visionneuse qui alterne les images des différentes planètes du film choisi.

Voici l'API qui vous permettra d'obtenir ses informations : <https://swapi.dev/api>

La visionneuse doit avoir les fonctionnalités suivantes :

- Cette page doit présenter les images en rotation. Il doit y avoir à la fois le nom de la planète affichée et son image.
 - Le nom de la planète doit être affiché par-dessus l'image (dans le bas)
- La visionneuse doit également afficher 3 boutons : **Précédent**, **Suivant** et **Arrêter**.
 - Bouton *Précédent* :
 - Permet à l'utilisateur d'afficher l'image précédente. Si l'utilisateur clique sur ce bouton alors que la visionneuse affiche la première image, c'est la dernière image qui doit s'afficher.
 - Bouton *Suivant* :
 - Permet à l'utilisateur d'afficher l'image suivante. Si l'utilisateur clique sur ce bouton alors que la visionneuse affiche la dernière image, c'est la première image qui doit s'afficher.
 - Bouton *Arrêter* :

- Permet à l'utilisateur de désactiver la rotation automatique de la visionneuse. Lorsque l'utilisateur clique sur ce bouton, la rotation doit se désactiver et le bouton doit afficher « Activer ».
- Lorsque la visionneuse est démarrée (bouton *Activer*), elle doit changer d'image automatiquement à une vitesse déterminée par une liste déroulante (ou des boutons radios) :
 - Lent : 1500 millisecondes
 - Moyen : 1000 millisecondes
 - Rapide : 500 millisecondes

Attention ! Pour changer de vitesse, vous devrez stopper la minuterie courante et la redémarrer avec la nouvelle vitesse précisée.

- Lorsque le bouton « Arrêter » affiche « Activer », il doit permettre à l'utilisateur de réactiver la rotation de la visionneuse. Lorsque la rotation automatique de la visionneuse est activée, le bouton doit afficher « Arrêter ».
- Si l'utilisateur sélectionne le bouton « Précédent » ou « Suivant » alors que la rotation automatique est activée, le délai pour afficher la prochaine image doit être réinitialisé (la prochaine image doit s'afficher x secondes après que l'utilisateur ait cliqué sur le bouton – selon la vitesse déterminée plus haut).

3 Évolution des prix de l'essence

Vous devrez faire afficher un graphique de l'évolution du prix du pétrole sur différentes périodes de temps selon les critères entrés par l'utilisateur.

L'utilisateur devra pouvoir choisir 3 options avant que le graphique s'affiche :

- Le type de monnaie
 - Canada, France, Angleterre, Danemark, Egypte, Hongrie, Etc.
 - Créer une liste à partir de l'API
- L'état américain pour lequel il veut voir les prix (avec leur code pour l'API) :
 - New York : PET.EMM_EPMO_PTE_SNY_DPG.M
 - Californie : PET.EMM_EPMO_PTE_SCA_DPG.M
 - Floride : PET.EMM_EPMO_PTE_SFL_DPG.M
 - Colorado : PET.EMM_EPMO_PTE_SCO_DPG.M
 - Massachusetts : PET.EMM_EPMO_PTE_SMA_DPG.M
 - Texas : PET.EMM_EPMO_PTE_STX_DPG.M
- La période de temps
 - Début/fin : mois et année

Voici deux APIs qui pourront vous être utiles pour répondre à cette demande :

- Taux de change : https://apilayer.com/marketplace/description/exchangerates_data-api#documentation-tab
- Prix du pétrole par état : <https://api.eia.gov/series/>
 - Commencer par obtenir votre API key

Pour ce qui est du graphique, vous devrez trouver une librairie JavaScript qui vous permettra de le faire. Essayez-en quelques-unes avant d'arrêter votre choix 😊

4 Collaboration et plagiat

Un travail pratique ne peut **en aucun temps** être le produit d'une collaboration ou de partage de code entre les équipes. Toute collaboration, partage de code¹, ou autre interaction sera sanctionnée selon la clause 6.1.12 du plan de cours :

Tout acte de plagiat, de tricherie et de fraude sera sanctionné. Constitue notamment un plagiat, une tricherie ou une fraude tout acte de copier, de fournir ou de recevoir volontairement de l'information lors d'un examen, de reproduire en tout ou en partie le travail d'une autre personne, qu'il s'agisse d'un document imprimé, multimédia ou électronique, sans y faire expressément référence, de remplacer un étudiant ou de se faire remplacer lors d'un examen, de remettre un travail réalisé par une autre personne, d'obtenir, posséder ou utiliser frauduleusement des questions ou réponses d'examen, d'utiliser du matériel, des applications, des sites Web ou des ressources non autorisés, de se faire aider par une autre personne lorsqu'il est demandé de réaliser l'évaluation seul, de falsifier les résultats de travaux ou d'examens.

Source : Politique d'évaluation des apprentissages du collège

Si de tels comportements sont observés en classe, le professeur se réserve le droit de mettre fin à l'évaluation des étudiants fautifs en cours de réalisation et d'attribuer la note 0.

5 Critères d'évaluation

Éléments	Pondération
• Visionneuse	30%
• Évolution des prix du pétrole	45%
• Qualité du code	20%

¹ Du code identique avec des variables renommées et/ou des espaces introduits est considéré comme du plagiat.

Éléments	Pondération
<ul style="list-style-type: none"> ○ Bonne utilisation des patrons de conception <ul style="list-style-type: none"> ▪ MVC, DTO, Entité, Modèles, Relations, Request, etc. ○ Bien découpé et indenté ○ Code clair et simple ○ Principes SOLID ○ Respecte les normes et la structure de Laravel 	
<ul style="list-style-type: none"> • Utilisation adéquate de Git <ul style="list-style-type: none"> ○ Branches, participation équilibrée et régulière, etc. 	5%

Pour chaque faute de français, 0,5% sera soustrait de la note finale (pour un maximum de 20%).

6 Documents à réaliser et à remettre

Veillez remettre sur « LÉA » une archive « zip » contenant tous les fichiers du projet :

- Lien Git avec accès pour l'utilisateur **filiat00**
- Code (sans les dépendances – **sans le dossier *vendor***)

Bon succès!

Annexe – Base de données

