**Ce que vous visez**

Dans ce point de contrôle, nous verrons les sujets de l'itération avec async/await, de l'attente d'un appel, de la gestion des erreurs avec async/await, du chaînage async/await, de l'attente de requêtes simultanées et de l'attente d'appels parallèles :

Choisissez au moins 3 tâches à résoudre.

**Instructions**

**Tâche 01 :**

**Itération avec Async/Await :** écrivez une fonction asynchrone **iterateWithAsyncAwait** qui prend un tableau de valeurs et enregistre chaque valeur avec un délai de 1 seconde entre les journaux.

**Tâche 02 :**

**En attente d'un appel :** créez une fonction asynchrone **waitCall** qui simule la récupération de données à partir d'une API. Utilisez **wait** pour attendre la réponse de l'API, puis enregistrez les données.

**Tâche 03 :**

**Gestion des erreurs avec Async/Await :** modifiez la fonction **waitCall** pour gérer les erreurs avec élégance. Si l'appel d'API échoue, détectez l'erreur et enregistrez un message d'erreur convivial.

**Chaînage Async/Await :** écrivez une fonction **chainedAsyncFunctions** qui appelle séquentiellement trois fonctions asynchrones distinctes. Chaque fonction doit enregistrer un message après un délai de 1 seconde. Chaînez ces fonctions en utilisant **wait** .

**Tâche 04 :**

**En attente de demandes simultanées :** créez une fonction asynchrone **concurrentRequests** qui effectue deux appels d'API simultanément à l'aide **de Promise.all()** . Enregistrez les résultats combinés une fois les deux demandes résolues.

**Tâche 05 :**

**En attente d'appels parallèles :** écrivez une fonction **parallelCalls** qui prend un tableau d'URL et récupère les données de chaque URL simultanément à l'aide **de Promise.all()** . Enregistrez les réponses une fois que toutes les demandes sont terminées.

**Comment nous évaluerons**

Votre point de contrôle sera évalué en fonction de ces critères, chacun noté de 0 à 5.

Évaluation du point de contrôle

* Vos compétences techniques
* Qualité de travail
* Des talents pour la résolution des problèmes
* Alerte date limite : (0 score si manqué)