
Création d'un Quiz

PARTIE 1

Objectif

Développer un script PHP en ligne de commande qui permet à l'utilisateur de répondre à un quiz. Le script devra poser une série de questions, recueillir les réponses de l'utilisateur, et enfin afficher le score ainsi que les corrections à la fin du quiz.

Instructions

Création du Quiz

Le quiz doit contenir au moins 5 questions.

Chaque question doit proposer 4 options de réponse, numérotées de 1 à 4.

Il doit y avoir une et une seule réponse correcte par question.

Stockage des questions

Les questions, les réponses et la réponse correcte pour chaque question doivent être stockées dans un tableau multidimensionnel en PHP.

Voici une structure du tableau à utiliser :

```
$questions = [
    [
        'question' => "Quelle est la capitale de la France ?",
        'reponses' => ["1. Paris", "2. Berlin", "3. Rome", "4. Madrid"],
        'correct' => 1
    ],
    [
        'question' => "Quel est le résultat de 3 + 4 ?",
        'reponses' => ["1. 5", "2. 7", "3. 9", "4. 12"],
        'correct' => 2
    ],
    // Ajoutez d'autres questions ici
];
```

Déroulement du Quiz

Le script doit afficher les questions une par une, en indiquant les réponses possibles.

L'utilisateur doit pouvoir saisir le numéro de la réponse qu'il choisit.

Le script doit valider que la réponse saisie est un nombre valide entre 1 et 4. Si la réponse n'est pas valide, un message d'erreur doit être affiché et l'utilisateur doit être invité à saisir à nouveau sa réponse.

Calcul du score

Le script doit calculer le score de l'utilisateur en fonction de ses réponses.

Chaque réponse correcte vaut un point. Il n'y a pas de points négatifs pour les réponses incorrectes.

Affichage des résultats

À la fin du quiz, le script doit afficher le score total de l'utilisateur ainsi que le nombre total de questions.

Après l'affichage du score, le script doit afficher les corrections pour chaque question, en indiquant la question, la réponse de l'utilisateur et la bonne réponse.

Consignes supplémentaires

Commentez votre code pour expliquer le fonctionnement de votre script.

Assurez-vous que votre code est bien organisé et lisible pour faciliter sa compréhension.

Testez votre script pour vous assurer qu'il fonctionne correctement dans différents scénarios (réponses correctes, réponses incorrectes, saisies invalides).

PARTIE 2

Objectif

Le script PHP devra charger les questions depuis un **fichier CSV**, puis à la fin du quiz, afficher le score de l'utilisateur et les corrections avec un affichage en couleur pour différencier les réponses correctes des incorrectes.

Instructions

Préparation du fichier CSV :

Créez un fichier CSV nommé **questions.csv**.

Chaque ligne du fichier représentera une question avec ses réponses et la bonne réponse.

Format de chaque ligne :

Question,Réponse1,Réponse2,Réponse3,Réponse4,NuméroRéponseCorrecte

Exemple de ligne :

Quelle est la capitale de la France ?,Paris,Berlin,Rome,Madrid,1

Lecture des questions depuis le fichier CSV

Au début du script PHP, les questions devront être chargées depuis le fichier CSV et stockées dans un tableau multidimensionnel (avec la même structure que dans la partie 1).

Affichage de la correction en couleur

Après l'affichage du score, le script doit afficher les corrections pour chaque question.

Si la réponse de l'utilisateur est correcte, affichez la correction en vert.

Si la réponse est incorrecte, affichez la correction en rouge.

```
Quiz terminé ! Votre score est : 1/2

Corrections :
Quelle est la capitale de la France ?
Votre réponse : 1 - Correct!
Quel est le résultat de 3 + 4 ?
Votre réponse : 4 - Incorrect! La bonne réponse était : 2
```

Fichiers CSV et utilisation en PHP

Qu'est-ce qu'un fichier CSV ?

CSV signifie "Comma-Separated Values" (Valeurs Séparées par des Virgules).

C'est un format de fichier simple, très utilisé, pour stocker des données tabulaires, comme un tableau ou une base de données, sous forme de texte.

Chaque ligne du fichier CSV correspond à une ligne du tableau, et chaque champ (ou colonne) de cette ligne est séparé par une virgule.

Pourquoi utiliser des fichiers CSV ?

- **Simplicité** : Les fichiers CSV sont faciles à lire et à écrire, aussi bien pour les humains que pour les machines.
- **Compatibilité** : Ils sont compatibles avec de nombreux programmes, y compris les tableurs comme Excel et les bases de données.
- **Légèreté** : Ils sont plus légers que d'autres formats de données, ce qui les rend idéaux pour le stockage et le partage de grandes quantités de données.

Utilisation des fichiers CSV en PHP

PHP offre plusieurs fonctions pour travailler avec des fichiers CSV, ce qui facilite la lecture et l'écriture de ces fichiers.

Lire un fichier CSV :

- Utilisez la fonction **fopen()** pour ouvrir le fichier en mode lecture ;
- Utilisez la fonction **fgetcsv()** pour lire les données ligne par ligne ;
- Utilisez la fonction **fclose()** pour fermer le fichier.

fgetcsv() analyse la ligne et retourne un tableau contenant les champs séparés.

VOIR DOCUMENTATION PHP !