📌 Projet : Jeu du Pendu en PHP 8 (POO)

🎯 Objectif

Vous allez développer un jeu du pendu en PHP 8 en utilisant la programmation orientée objet (POO). Ce projet vous permettra de renforcer vos compétences en POO, en gestion des chaînes de caractères et en interaction avec l'utilisateur via la console. Vous devrez mettre en pratique la création et l'interaction entre plusieurs classes, ainsi que les principes de base du jeu du pendu.

📝 Description du jeu

Le jeu du pendu est un jeu classique dans lequel le joueur doit deviner un mot en proposant des lettres. Le joueur a un nombre limité de tentatives pour découvrir le mot avant de perdre la partie. Chaque lettre correctement devinée est révélée, tandis qu'une lettre incorrecte réduit le nombre de tentatives restantes. Le joueur gagne s'il devine toutes les lettres du mot avant d'épuiser toutes ses tentatives.

📦 Structure du programme

Le programme est divisé en plusieurs classes :

1. Classe `MotADeviner`

Prototypes des méthodes de la classe `MotADeviner` :

- `__construct()`: Initialise la liste des mots possibles et sélectionne aléatoirement un mot à deviner. Initialise également l'état du mot caché avec des underscores.
- `getMot(): string`: Retourne le mot à deviner.
- `getMotCache(): array`: Retourne l'état actuel du mot caché, incluant les lettres trouvées et les underscores.
- `updateMotCache(string $lettre): void`: Met à jour le mot caché avec une lettre correctement devinée par le joueur.
- `isComplete(): bool`: Vérifie si le mot a été complètement deviné par le joueur.

Cette classe gère le mot à deviner.
- Elle contient une liste de mots parmi lesquels un mot est sélectionné aléatoirement.
- Elle gère également l'état du mot caché (avec des underscores "_") et permet de mettre à jour le mot caché lorsque le joueur propose une lettre correcte.

2. Classe `JeuPendu`

Prototypes des méthodes de la classe `JeuPendu` :

- `__construct()`: Initialise une nouvelle instance du jeu du pendu, crée un objet `MotADeviner`, définit le nombre de tentatives restantes et initialise la liste des lettres proposées.
- `jouer(): void`: Méthode principale qui lance la boucle de jeu. Elle affiche l'état actuel du mot, gère les tentatives et vérifie si le joueur a gagné ou perdu.
- `afficherMotCache(): void`: Affiche l'état actuel du mot à deviner avec les lettres découvertes et les underscores pour celles non trouvées.
- `demanderLettre(): string`: Demande à l'utilisateur d'entrer une lettre, valide l'entrée et renvoie la lettre.
- `traiterLettre(string $lettre): void`: Traite la lettre proposée par le joueur, met à jour l'état du mot si la lettre est correcte et réduit le nombre de tentatives restantes si elle est incorrecte.

Cette classe gère la logique principale du jeu.
- Elle permet au joueur de proposer des lettres, affiche l'état actuel du mot à deviner, et maintient le compte des tentatives restantes.
- Elle est responsable de vérifier si le joueur a gagné ou perdu.

⚙ Fonctionnalités attendues

- Le joueur doit pouvoir entrer des lettres via la console.
- Le programme doit valider l'entrée de l'utilisateur (uniquement une lettre, pas de caractères spéciaux ou de chiffres).
- Le programme doit afficher le mot à deviner avec les lettres trouvées et les underscores pour celles qui restent à deviner.
- Le programme doit tenir compte des lettres déjà proposées par le joueur.
- Le joueur dispose de 7 tentatives pour deviner le mot.
- Le programme doit indiquer clairement si le joueur a gagné ou perdu.

💡 Indications supplémentaires

- Le mot est sélectionné aléatoirement parmi une liste définie de mots (par exemple : "developpement", "informatique", "programmation", etc.). Vous pouvez améliorer le jeu en chargeant la liste des mots à deviner depuis un fichier externe (par exemple, un fichier `.txt` avec une liste de mots). Cela permettrait de séparer la logique de jeu de la configuration, rendant l'application plus flexible et facile à mettre à jour.

- Chaque fois qu'une lettre est proposée, l'état du mot caché est mis à jour si la lettre est correcte.

- Le joueur perd une tentative s'il propose une lettre incorrecte.

💻 Exercice à réaliser

- Implémentez ce jeu en respectant la structure fournie.
- Assurez-vous que le jeu fonctionne correctement, avec toutes les fonctionnalités décrites.