

OOP : Création d'une application de gestion d'employés

Description :

Vous allez créer une application de gestion d'employés pour une entreprise. Chaque employé aura un nom, un identifiant unique, un poste et un salaire. L'entreprise aura plusieurs types d'employés, tels que les employés à temps plein, les employés à temps partiel et les cadres. Chaque type d'employé aura ses propres caractéristiques spécifiques.

Tâches à accomplir :

1. Créez une classe abstraite `Employe` avec les propriétés suivantes :
 - `\$nom` : le nom de l'employé.
 - `\$id` : un identifiant unique pour l'employé.
 - `\$poste` : le poste occupé par l'employé.
 - `\$salaire` : le salaire de l'employé.
 - Définissez des méthodes abstraites pour calculer le salaire de l'employé en fonction de son type.
2. Créez des classes spécifiques pour chaque type d'employé :
 - `EmployeTempsPlein` : Un employé à temps plein aura un salaire fixe mensuel.
 - `EmployeTempsPartiel` : Un employé à temps partiel sera payé à l'heure.
 - `Cadre` : Un cadre recevra un salaire de base plus des bonus en fonction de la performance de l'entreprise.
3. Implémentez des méthodes spécifiques pour chaque type d'employé :
 - `calculerSalaire()` : Calcule le salaire de l'employé en fonction de son type.
4. Créez plusieurs instances de chaque type d'employé pour simuler une équipe diversifiée.
5. Testez les fonctionnalités en calculant le salaire de chaque employé et en affichant les détails de chaque employé.

Critères de performance :

- Utilisez l'héritage pour éviter la duplication de code et améliorer la réutilisation du code.
- Utilisez l'encapsulation pour protéger les propriétés de la classe et garantir l'intégrité des données.
- Implémentez des méthodes pour définir et récupérer les valeurs des propriétés de manière sécurisée (getters et setters).
- Expérimentez avec le polymorphisme en créant des méthodes spécifiques pour chaque type d'employé.