

## TP : Gestion des Utilisateurs

### Contexte :

Vous travaillez pour une entreprise de développement de logiciels qui a besoin d'une fonctionnalité de gestion des utilisateurs pour leur nouveau système. Vous êtes chargé de créer une classe PHP pour gérer les utilisateurs, en utilisant les concepts que vous avez appris sur les classes, les interfaces, les énumérations, et autres.

### Tâches :

1- Créez une classe abstraite Utilisateur avec les méthodes suivantes :

getId() : Cette méthode doit renvoyer l'identifiant de l'utilisateur.

getNom() : Cette méthode doit renvoyer le nom de l'utilisateur.

getType() : Cette méthode doit renvoyer le type de l'utilisateur (par exemple, "administrateur", "modérateur", "utilisateur").

abstract public function getDescription(): Cette méthode doit être abstraite et doit renvoyer une description de l'utilisateur.

2-Implémentez trois classes concrètes qui étendent la classe abstraite Utilisateur :

Administrateur : Un utilisateur avec des privilèges administratifs.

Modérateur : Un utilisateur avec des privilèges de modération.

UtilisateurNormal : Un utilisateur standard sans privilèges spéciaux.

3-Utilisez une énumération pour représenter les différents types d'utilisateurs.

4-Créez une classe GestionUtilisateurs avec les méthodes suivantes :

ajouterUtilisateur(Utilisateur \$utilisateur) : Cette méthode doit ajouter un utilisateur à la liste des utilisateurs.

afficherUtilisateurs() : Cette méthode doit afficher la liste des utilisateurs avec leurs détails (identifiant, nom, type et description).

### Instructions :

Créez un nouveau fichier PHP et implémentez les classes et les méthodes décrites ci-dessus.

Utilisez des instances de ces classes pour ajouter des utilisateurs à la liste et afficher la liste des utilisateurs.

Testez votre implémentation en ajoutant différents types d'utilisateurs et en affichant la liste.

Assurez-vous de respecter les principes de l'orienté objet et d'utiliser les concepts appris en classe, tels que les classes abstraites, les interfaces, les énumérations, etc.

### Critères de réussite :

Les classes Utilisateur, Administrateur, Modérateur et UtilisateurNormal sont correctement implémentées avec les méthodes demandées.

La classe GestionUtilisateurs est correctement implémentée avec les méthodes demandées.

L'ajout d'utilisateurs et l'affichage de la liste des utilisateurs fonctionnent comme prévu.

Le code est bien organisé, propre et respecte les principes de l'orienté objet.