

# Exercices centrés sur les interfaces en PHP :

---

## Exercice 1 : Interface avec Paramètres

### Énoncé :

Créez une interface `Calculable` avec une méthode `calculer($a, $b)` qui retourne un nombre. Ensuite, créez deux classes `Addition` et `Multiplication` qui implémentent l'interface `Calculable`.

### Instructions :

1. Définissez l'interface `Calculable` avec la méthode `calculer($a, $b)`.
  2. Implémentez la classe `Addition` qui retourne la somme de `$a` et `$b`.
  3. Implémentez la classe `Multiplication` qui retourne le produit de `$a` et `$b`.
  4. Créez des instances de `Addition` et `Multiplication`, puis appelez `calculer()` sur chaque instance avec des valeurs d'exemple.
- 

## Exercice 2 : Interfaces et Polymorphisme

### Énoncé :

Créez une interface `Connectable` avec une méthode `connecter()`. Ensuite, créez deux classes `BaseDeDonnees` et `Api` qui implémentent l'interface `Connectable`.

### Instructions :

1. Définissez l'interface `Connectable` avec la méthode `connecter()`.
  2. Implémentez la classe `BaseDeDonnees` qui redéfinit `connecter()` pour afficher "Connexion à la base de données".
  3. Implémentez la classe `Api` qui redéfinit `connecter()` pour afficher "Connexion à l'API".
  4. Créez un tableau d'instances de `Connectable` contenant des objets de type `BaseDeDonnees` et `Api`, puis parcourez ce tableau pour appeler `connecter()` sur chaque objet.
- 

## Exercice 3 : Interface avec Méthodes Magiques

### Énoncé :

Créez une interface `Renderable` avec une méthode `render()`. Créez deux classes `Image` et `Texte` qui implémentent l'interface `Renderable` et utilisent une méthode magique `__toString()` pour retourner une représentation de l'objet.

### Instructions :

1. Définissez l'interface `Renderable` avec la méthode `render()`.
2. Implémentez la classe `Image` qui redéfinit `render()` pour retourner une chaîne indiquant "Rendu d'une image".
3. Implémentez la classe `Texte` qui redéfinit `render()` pour retourner une chaîne indiquant "Rendu d'un texte".
4. Ajoutez la méthode magique `__toString()` dans chaque classe pour retourner une description de l'objet.
5. Créez des instances de `Image` et `Texte`, et affichez les objets en utilisant `echo` pour vérifier la méthode magique.