# Cours 3: Interfaces, Traits et Gestion des Erreurs avec PHP 8 (2 heures)

## Objectifs du Cours :

- Comprendre le rôle des interfaces et des traits en PHP 8.

- Apprendre à utiliser les interfaces et les traits pour organiser le code de manière efficace.

- Explorer les mécanismes de gestion des erreurs dans PHP 8.

---

## Partie 1: Interfaces (1 heure)

### 1.1 Introduction aux Interfaces

#### Qu'est-ce qu'une interface?

Une interface définit un ensemble de méthodes que des classes doivent implémenter. Elle permet de définir un contrat que les classes doivent suivre, favorisant ainsi une structure plus souple et modulaire du code.

#### Utilisation des Interfaces en PHP 8

```php

<?php

interface Forme {

public function calculerSurface();

public function calculerPerimetre();

}

class Rectangle implements Forme {

private $longueur;

private $largeur;

public function \_\_construct($longueur, $largeur) {

$this->longueur = $longueur;

$this->largeur = $largeur;

}

public function calculerSurface() {

return $this->longueur \* $this->largeur;

}

public function calculerPerimetre() {

return 2 \* ($this->longueur + $this->largeur);

}

}

?>

```

##### Explications :

- L'interface `Forme` définit les méthodes `calculerSurface` et `calculerPerimetre`.

- La classe `Rectangle` implémente cette interface en fournissant les implémentations nécessaires.

### 1.2 Traits

#### Qu'est-ce qu'un trait?

Un trait est un mécanisme de réutilisation de code dans une seule classe. Il permet d'inclure des méthodes dans une classe de manière cohérente, même si PHP ne supporte pas l'héritage multiple.

#### Utilisation des Traits en PHP 8

```php

<?php

trait Enregistrement {

public function enregistrer() {

echo "L'objet est enregistré.";

}

}

class Utilisateur {

use Enregistrement;

}

$utilisateur = new Utilisateur();

$utilisateur->enregistrer(); // Appel de la méthode du trait

?>

```

##### Explications :

- Le trait `Enregistrement` contient la méthode `enregistrer`.

- La classe `Utilisateur` utilise ce trait pour inclure la méthode `enregistrer`.

---

## Partie 2: Gestion des Erreurs (30 minutes)

### 2.1 Exceptions

#### Qu'est-ce qu'une exception?

Les exceptions sont des erreurs qui se produisent pendant l'exécution d'un script. En PHP, les exceptions permettent de gérer des situations exceptionnelles de manière propre et structurée.

#### Utilisation des Exceptions en PHP 8

```php

<?php

class DivisionParZeroException extends Exception {}

function diviser($num, $denom) {

if ($denom === 0) {

throw new DivisionParZeroException("Division par zéro.");

}

return $num / $denom;

}

try {

$resultat = diviser(10, 0);

echo "Résultat : $resultat";

} catch (DivisionParZeroException $e) {

echo "Erreur : " . $e->getMessage();

}

?>

```

##### Explications :

- La fonction `diviser` lance une exception si le dénominateur est zéro.

- La structure `try-catch` permet de gérer l'exception et d'afficher un message approprié.

### 2.2 Gestion des Erreurs personnalisée

#### Utilisation du Gestionnaire d'Erreurs personnalisé en PHP 8

```php

<?php

set\_error\_handler(function ($errno, $errstr) {

echo "Erreur [$errno] : $errstr";

});

echo $variableNonDeclaree; // Déclenche une erreur

?>

```

##### Explications :

- La fonction `set\_error\_handler` permet de définir une fonction de gestion des erreurs personnalisée.

- L'erreur est interceptée et traitée selon les besoins du développeur.

---

## Partie 3: Exercices Pratiques (30 minutes)

1. \*\*Interfaces et Traits :\*\*

- Créez une interface `Transport` avec les méthodes `demarrer` et `arreter`. Implémentez-la dans deux classes distinctes représentant un vélo et une voiture.

2. \*\*Gestion des Erreurs :\*\*

- Écrivez une fonction de division qui génère une exception si le dénominateur est nul. Utilisez ensuite un bloc `try-catch` pour gérer cette exception.

---

## Ressources Complémentaires

- [Documentation PHP sur les interfaces](https://www.php.net/manual/en/language.oop5.interfaces.php)

- [Documentation PHP sur les traits](https://www.php.net/manual/en/language.oop5.traits.php)

- [Documentation PHP sur les exceptions](https://www.php.net/manual/en/language.exceptions.php)