



Workshop N°5:

PHP

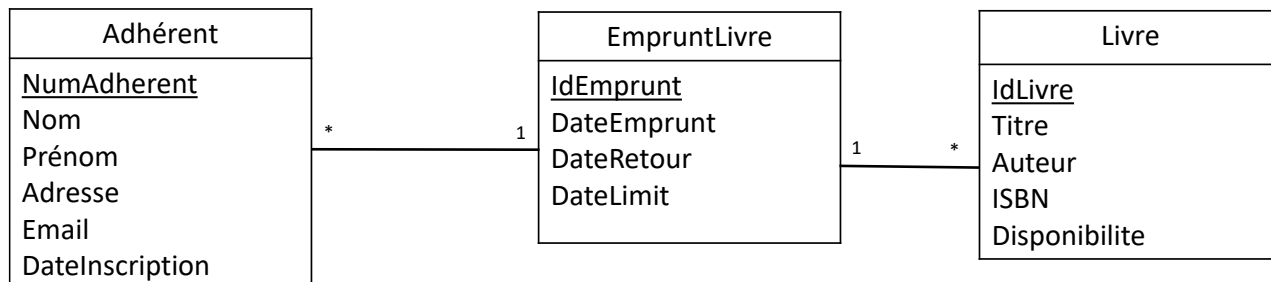
Objectifs :

- ✓ Maîtriser les notions de l'OO
- ✓ Création d'une base de données MySQL
- ✓ Introduction à l'interface PDO.

Remarque : Respecter la structure MVC.

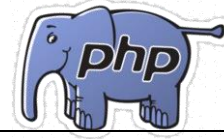
Exercice

Une société, éditrice de progiciels, souhaite lancer prochainement sur le marché un logiciel de gestion de bibliothèque. Dans cet atelier, on s'intéresse en particulier à la gestion des adhérents ainsi que les emprunts des ouvrages.



Partie 1 : Configuration

1. A travers l'application **phpMyAdmin**, créer une base de données intitulée « **Bibliothèque** ».
2. Créer la table « **Adhérent** » sachant que :
 - a. Les attributs de cette table sont les propriétés de la classe Adhérent
 - b. Ajouter un attribut « **NumAbon** » qui sera la clé primaire de la table.
3. Ajouter des enregistrements à la table « **Adhérent** » via l'application **phpMyAdmin**.
4. Créer la classe « **config** » dans le fichier « **config.php** ». Au niveau de cette classe, créer la fonction « **getConnexion()** » qui permet d'établir la connexion avec la base de données comme présentée si dessous :



Workshop N°5:

PHP

```
<?php
class config {
    private static $pdo = NULL;

    public static function getConnexion() {
        if (!isset(self::$pdo)) {
            try{
                self::$pdo = new PDO('mysql:host=localhost;dbname=atelierPHP', 'root', '',
                    [
                        PDO::ATTR_ERRMODE => PDO::ERRMODE_EXCEPTION,
                        PDO::ATTR_DEFAULT_FETCH_MODE => PDO::FETCH_ASSOC
                    ]
                );
            }catch(Exception $e){
                die('Erreur: '.$e->getMessage());
            }
        }
        return self::$pdo;
    }
}
```

Partie 2 : Afficher la liste des adhérents

5. Créer un fichier « **AdhérentC.php** » au niveau du dossier « **Controller** ».
6. Créer la méthode « **afficherAdhérent ()** » permettant d’afficher la liste des adhérents dans la base de données.
7. Créer un fichier « **afficherListeAdhérents.php** » au niveau du dossier « **Views** ».
8. Dans le fichier « **afficherListeAdhérents.php** », faites appel à cette méthode afin d’afficher la liste des adhérents enregistrés dans la table « **Adhérent** » dans un tableau.

Partie 3 : Ajouter un Adhérent

9. Créer la méthode « **ajouterAdhérent ()** » permettant d’enregistrer un utilisateur passé en paramètre dans la base de données.



Workshop N°5:

PHP

10. Au niveau du fichier « **ajouterAdhérent.php** », ajouter une redirection vers le fichier « **afficherListeAdhérents.php** » après l'appel de la fonction d'ajout.

Partie 4 : Supprimer un Adhérent

11. Ajouter un bouton « **supprimer** » permettant de supprimer un Adhérent selon son numéro.
Après la suppression, ajouter une redirection vers le fichier « **afficherListeAdhérents.php** ».

Partie 5 : Modifier un Adhérent

12. Ajouter un lien « **modifier** » qui redirige vers le fichier « **modifierAdhérent.php** » en envoyant le numéro d'adhérent à modifier.
13. Au niveau du fichier « **modifierAdhérent.php** », les informations de l'adhérent à modifier doivent être affichées dans un formulaire. Après modification, les nouvelles informations sont enregistrées dans la base.

Partie 6 : Jointure

14. Ajouter la table « **Livre** » à la base de données « **Bibliothèque** » sachant que :
- a. Les attributs de cette table sont les propriétés de la classe Livre
 - b. L'attribut « **IdLivre** » est la clé primaire de la table.
15. Ajouter la table « **EmpruntLivre** » à la base de données « **Bibliothèque** » sachant que :
- a. Les attributs de cette table sont les propriétés de la classe **EmpruntLivre**.
 - b. L'attribut « **IdEmprunt** » est la clé primaire de la table.
 - c. Ajouter les deux liaisons de jointure suivantes :
 - ✓ Entre la table « **EmpruntLivre** » et la table « **Adhérent** ». L'attribut



Workshop N°5:

PHP

«**NumAbon**» de la table « **EmpruntLivre** » va référencer l'attribut
«**NumAbon**» de la table « **Adhérent** ».

- ✓ Entre la table « **EmpruntLivre** » et la table « **Livre** ». L'attribut « **IdLivre** »
de la table « **EmpruntLivre** » va référencer l'attribut « **IdLivre** » de la table «
Livre »

Partie 7

16. Créer la méthode « **rechercheLivre()** » qui permet de rechercher un livre selon son titre.
17. Créer la méthode « **dateLimiteDepassee ()** » permettant de lister les différents empreints
dont la date de retour est dépassée (Date retour < date limite).
18. Créer la méthode « **adherentsTardifs()** » permettant d'afficher la liste des adhérents qui
ont dépassés la date limite de retour.