esprit Se former autrement HONORIS UNITED UNIVERSITIES	**Année Universitaire 2023-2024**

Workshop N°6:

PHP

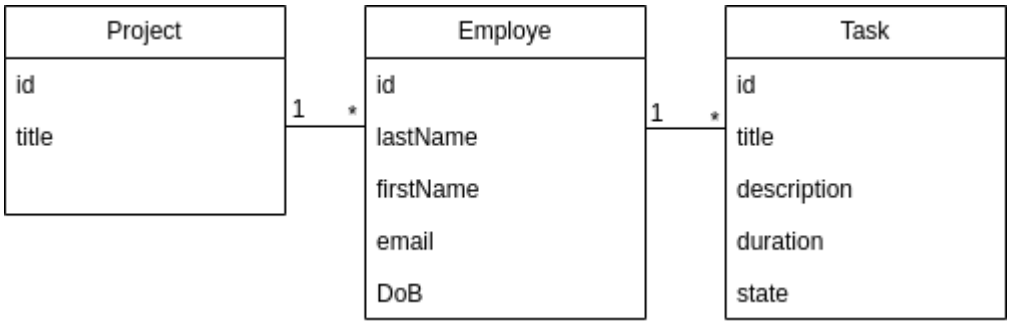
Objectifs :

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

<u>Remarque :</u> Respecter la structure MVC.

Énoncé :

Une société de développement souhaite créer un site web pour une entreprise qui offre la possibilité à ses employés de s'inscrire, de consulter leurs projets et leurs tâches.



Partie 1 : Configuration

1. A travers l'application **phpMyAdmin**, créer une base de données intitulée « **atelierPHP** ».
2. Créer la table « **Employe** » sachant que :
   a. Les attributs de cette table sont les propriétés de la classe Employe
   b. Ajouter un attribut « **id** » qui sera la clé primaire de la table.
3. Ajouter des enregistrements à la table « **Employe** » 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 ci-dessous :

esprit Se former autrement HONORIS UNITED UNIVERSITIES	**Année Universitaire 2023-2024**

Workshop N°6:

PHP

```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 employés

5. Créer un fichier « **EmployeC.php** » au niveau du dossier « **Controller** ».
6. Créer la méthode « **listEmployes ()** » permettant d'afficher la liste des employés dans la base de données.
7. Créer un fichier « **ListEmployes.php** » au niveau du dossier « **Views** ».
8. Dans le fichier « **ListEmployes.php** », faites appel à la méthode « **listEmployes ()** » afin d'afficher la liste des employés enregistrés dans la table « **Employe** » dans un tableau.

Partie 3 : Supprimer un Employé

9. Créer la méthode « **deleteEmploye ($id)** » de supprimer un employé selon son identifiant passé en paramètre.
10. Ajouter un lien « **delete** » permettant de faire appel à la fonction « **deleteEmploye($id)** ». Après la suppression, ajouter une redirection vers le fichier «**ListEmployes.php**».

esprit Se former autrement HONORIS UNITED UNIVERSITIES	**Année Universitaire 2023-2024**

Workshop N°6:

PHP

Partie 4 : Ajouter un Employé

11. Créer la méthode « **addEmploye ()** » permettant d'enregistrer un employé passé en paramètre dans la base de données.
12. Créer un fichier « **addEmploye.php** » au niveau du dossier « **Views** ».
13. Au niveau du fichier « **addEmploye.php** », ajouter une redirection vers le fichier «**ListEmployes.php** » après l'appel de la fonction d'ajout.

Partie 5 : Modifier un Employé

14. Ajouter un bouton « **update** » qui redirige vers le fichier « **updateEmploye.php** » en envoyant l'identifiant de l'employé à modifier.
15. Au niveau du fichier « **updateEmploye.php** », les informations de l'employé à modifier doivent être affichées dans un formulaire. Après modification, les nouvelles informations sont enregistrées dans la base.

Partie 6 : Jointure

16. Ajouter la table **« Task »** à la base de données « **atelierPHP** » sachant que :
    a. Les attributs de cette table sont les propriétés de la classe Task.
    b. L'attribut « **Id** » est la clé primaire de la table.

17. Ajouter la table **« Project »** à la base de données « **atelierPHP** » sachant que :
    a. Les attributs de cette table sont les propriétés de la classe **Project**.
    b. L'attribut « **Id** » est la clé primaire de la table.

c. Ajouter les deux liaisons de jointure suivantes :

- ✓ Entre la table « **Project**» et la table «**Employe**». L'attribut « **IdProject** » de la table «**Employe**» va référencer l'attribut «**Id**» de la table «**Project** ».
- ✓ Entre la table «**Employe**» et la table «**Task**». L'attribut «**IdEmploye** » de la table «**Task** » va référencer l'attribut «**Id** » de la table « **Employe** ».

Partie 7 : Recherche

18. Créer la méthode « **searchTaskByTitle ($title)** » qui permet de lister les tâches selon l'attribut **« title »**.
19. Créer la méthode « **getTasksByDuration ($duration)** » permettant de lister les tâches qui dépasse la durée mise en paramètre (**$task->duration** > **$duration**).
20. Créer la méthode « **deleteTasksDone()** » qui permet de supprimer les tâches dont l'état (**« state »**) est défini comme 'terminé'.