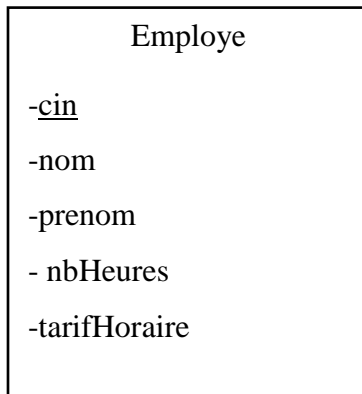


Soit le diagramme de classe suivant :



### Partie 1 : Manipulation de classes

1- Définir la classe "**Employe**" sachant que :

- Tous les attributs sont privés.
- Possède un constructeur qui permet d'initialiser tous ses attributs

2- Créer les méthodes suivantes :

- `afficherEmploye` : permet d'afficher les informations relatives à un employé.
- `calculerSalaire` : permet de calculer le salaire d'un employé

(nombreHeures\* tarifHoraire)

3- Créer un fichier appelé "**start.php**" permettant de:

a. Créer un employé:

- cin: **75757575**
- nom: **BEN Ahmed**
- prenom: **Salah**
- nombresHeures: **50**
- tarifHoraire: **20**

b. Afficher les informations de l'employé.

c. Afficher le salaire de l'employé.

4-

- Mettre en commentaire le constructeur paramétré de la classe employé.
- Refaire les questions 3.a et 3.b

### Partie 2 : Création d'un objet employé à travers un formulaire

- 5- Dans un fichier appelé **ajoutEmploye.html** créer un formulaire qui nous permette de saisir les informations d'un employé.

Ajouter Employe

CIN	<input type="text"/>
Nom	<input type="text"/>
Prenom	<input type="text"/>
nb heures	<input type="text"/>
tarif horaire	<input type="text"/>
<input type="button" value="ajouter"/>	

- 6- Dans le fichier **ajoutEmploye.php**, créer un objet « **employe1** » à partir des informations récupérées à partir du formulaire. La méthode d'envoi des données est POST.
- 7- Afficher les informations de l'employé avec la méthode `var_dump()`.

### Partie 3 : Ajouter, afficher, supprimer, modifier un employé

- 8- A travers l'application `phpmyadmin`, créer une base de données intitulée « **atelierphp** ».
- 9- Créer ensuite la table « **employe** ».
- Les colonnes de cette table sont les attributs de la classe `employe`.
  - `Cin` est la clé primaire de la table `employe`.
- 10- 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 des données.

```
<?php

class config {

private static $instance = NULL;

public static function getConnexion() {

    if (!isset(self::$instance)) {

        self::$instance = new PDO('mysql:host=localhost;dbname=formationphp',
        'root', '');

    }

}
```

```

        return self::$instance;

    }

}

?>

```

- 11- Créer la méthode **ajouterEmployer()** permettant d'enregistrer dans la base de données un employé passé en paramètre.
- 12- Créer la méthode **afficherEmployes()** permettant d'afficher la liste des employés dans la base des données.
- 13- Dans le fichier `afficherEmploye.php`, appeler cette méthode afin d'afficher la liste des employés enregistrés dans la table `employe` dans un tableau.

Cin	Nom	Prenom	nb heures	tarif horaire
123456	rommene	ameni	30	20
123457	amine	amine	40	30
123458	ayari	asma	40	30

- 14- Au niveau du fichier `ajoutEmploye.php`, ajouter une redirection vers le fichier `afficherEmploye.php` après l'appel de la fonction d'ajout.
- 15- Ajouter un bouton « supprimer » permettant de supprimer un employe selon son cin. A la fin ajouter une redirection vers le fichier `afficherEmploye.php`.
- 16- Ajouter un lien « **modifier** » qui redirige vers le fichier **`modifierEmployer.php`** en envoyant le cin de l'employé courant.

**Modifier Employe**

CIN	<input type="text" value="123457"/>
Nom	<input type="text" value="amine"/>
Prenom	<input type="text" value="amine"/>
nb heures	<input type="text" value="40"/>
tarif horaire	<input type="text" value="30"/>
<input type="button" value="modifier"/>	

- 17- En cliquant sur modifier les informations sont modifiées dans la base et la liste des employés et affichés de nouveau