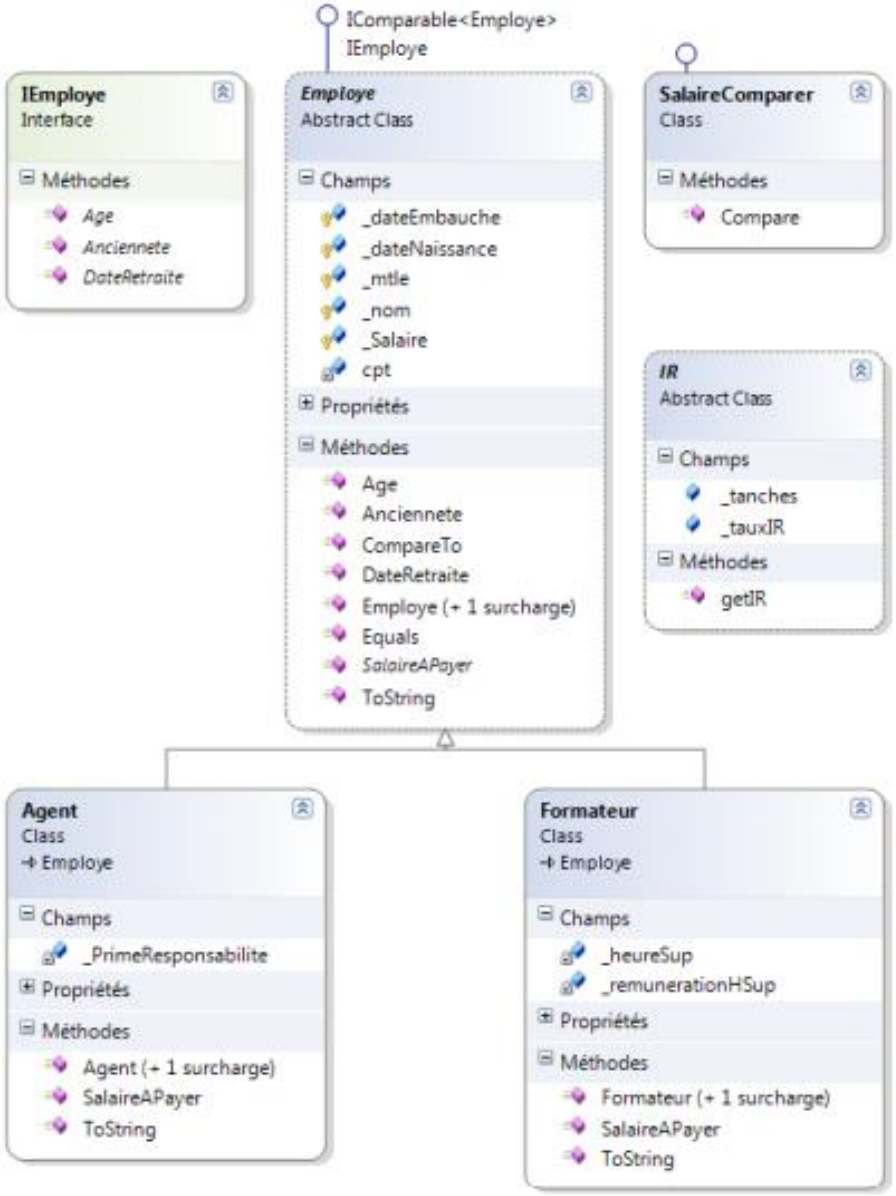
Université Internationale de Casablanca LAUREATE INTERNATIONAL UNIVERSITIES Nous innovons pour votre réussite !	**Ecole d'Ingénieurs** **Filières :** MIAGE **Classe :** 2ème année - S3	**Cours : POO, C#** **Professeur** : MOUJAHID Abdallah

Travaux Pratiques N° 5

POO avec C#

On souhaite créer une application pour la gestion des employés. Le diagramme de classes de cette application est affiché ci-dessous.



Travail demandé

L'IR ou l'impôt sur le revenu est un impôt global établi sur la totalité des revenus (le salaire) dont dispose une personne au cours d'une année déterminée. Pour déterminer le montant de l'IR on utilise le tableau de barème de l'IR suivant

Tranches de salaire annuel (en DH)	Taux IR en %
0 à 28 000	0%
28 001 à 40 000	12%
40 001 à 50 000	24%
50 001 à 60 000	34%
60 001 à 150 000	38%
Au-delà de 150 000	40%

Par exemple si le salaire d'un employé est 9 500 dh alors : Son salaire annuel est = salaire x 12 = 11500 x 12 =114 000 dh ➔ le taux IR est = 38%

1. Créer la classe abstraite **IR** avec :
    a. L'attribut _tranches : Un tableau statique initialisé par les valeurs des tranches ci-dessus.
    b. L'attribut _tauxIR : Un tableau statique de Taux IR initialisé par les valeurs de l'IR correspondantes
    c. Une fonction statique getIR(salaire) qui permet de retourner le taux d'IR correspondant au salaire passé en paramètre.

2. Créer l'interface **IEmploye** avec les méthodes suivantes :
    a. Une méthode Age() qui retournera l'âge d'un employé (Entier).
    b. Une méthode Ancienneté() qui retournera l'ancienneté d'un employé (le nombre d'années qu'il a travaillé).
    c. Une Méthode DateRetraite(ageRetraite) pour renvoyer la date de retraite de l'employé.

3. Créer la classe **Employe** avec :
    a. Les attributs (ils doivent être visibles dans les classes filles de la classe Employe) :
        - Matricule (incrémenté automatiquement) : _mtle
        - Nom : _nom
        - Date de naissance : _dateNaissance
        - Date d'embauche (la date de recrutement) : _dateEmbauche
        - Salaire de base : _salaireBase

    b. Les deux accesseurs DateEmbauche et DateNaissance sachant que l'âge de l'employé à la date de recrutement ne doit pas être inférieur à 16 ans, lever une exception sinon.
    c. Un constructeur par défaut et un autre d'initialisation.
    d. Une méthode abstraite SalaireAPayer() pour retourner le salaire net d'un employé

e. L'implémentation de l'interface IComparable pour comparer deux employés par nom.
f. L'implémentation de l'interface IEmploye avec ces trois méthodes.
g. La redéfinition de la méthode toString() qui renvoie toutes les propriétés séparées par '-'
h. La redéfinition de la méthode Equals() : 2 employés sont égaux s'ils ont le même matricule.

Un formateur est un employé avec en plus :

- Le nombre des heures supplémentaires par mois : _heureSup. La rémunération par heur supplémentaire, dont la valeur est partagée par tous les formateurs et par défaut égale à 70,00 dh : _remunerationHSup

4. Créer la classe **Formateur** avec :
   a. Les attributs.
   b. L'accesseur RemunerationHSup.
   c. Les deux constructeurs par défaut et d'initialisation.
   d. La méthode SalaireAPayer() pour calculer le salaire net d'un formateur sachant que :

Salaire net = (**SalaireBase** + NbrHS **x** **RemunerationHSup**) **x** (1 − taux IR)

Où $NbrHS = \begin{cases} \textbf{heureSup}, & si\ \textbf{heureSup} \leq 30 \\ \textbf{30}, & si\ non \end{cases}$

N.B : Utiliser la méthode getIR de la classe IR pour calculer le taux IR.

   e. La méthode ToString().

Un Agent est un employé avec en plus le montant de la prime de responsabilité : _primeResponsabilite.

5. Créer la classe **Agent** avec La méthode SalaireAPayer() pour calculer le salaire net d'un agent sachant que : Salaire net = (SalaireBase + PrimeResponsabilité) x (1 – taux IR)