Conception & Développement Informatique

NOM DU MODULE

TYPE: APPRENTISSAGE / TRAVAUX PRATIQUES



Exercice base de l'objet en java

sous-titre

CO	NIT	ENI	П
CU	IN I	EIN	U

EXERCICE: Classe Employé	
--------------------------	--

MEISTERTZHEIM Adeline 23/01/2024

EXERCICE: CLASSE EMPLOYE

Vous devez définir une classe Employe caractérisée par les attributs : matricule, nom, prenom, anneeNaissance, anneeEmbauche, salaire.

- Ajouter à la classe la méthode age() qui retourne l'âge de l'employé.
- Ajouter à la classe la méthode anciennete () qui retourne le nombre d'années d'ancienneté de l'employé.
- Ajouter à la classe la méthode augmentationDuSalaire() qui augmente le salaire de l'employé en prenant en considération l'ancienneté en utilisant le pseudo-code suivant :

```
si ancienneté < 5 ans, alors
on ajoute 2%
sinon si Ancienneté < 10 ans, alors
on ajoute 5%
sinon
on ajoute 10%
fin si
fin si
```

• Ajouter la méthode afficherEmploye() qui affiche les informations de l'employé : matricule, nom, prenom, age, ancienneté et salaire en €.

Diagramme UML:

```
Employe

- matricule: string

- nom: string

- prenom: string

- anneeNaissance: int

- anneeEmbauche: int

- salaire: double

+ Employe(int_matricule, string_nom, string_prenom, int_anneeNaissance, int anneEmbauche, double salaire)

+ age (): int

+ anciennete(): int

+ augmentationDuSalaire(): void

+ afficherEmploye(): void
```

Vérification:

```
Employe agent = new Employe("007", "Bond", "James", 1970, 1995, 7500);
agent.augmentationDuSalaire();
agent.afficherEmploye();
```



Affichage dans la console :

Matricule : 007

Nom : Bond

Prenom : James

Age : 54

Anciennete : 29 Salaire : 8250.0

Remarque : Instructions permettant de récupérer l'année courante de l'horloge du PC :

```
import java.util.Calendar;
private Calendar cal = Calendar.getInstance();
private int anneeCourante = cal.get(Calendar.YEAR);
```

--- FIN DU DOCUMENT ---

