

TP08 Correction

Matière : ATELIER DE PROGRAMMATION OBJET

Classes : SEM2

```
package societe;
public interface Salarie {
    public abstract float getSalaireBrut();
    public abstract float getSalaireNet();
    public abstract float getImpot();
}
package societe;
public abstract class Employe implements Salarie {
    protected String nom;
    protected String prenom;
    protected String matricule;
    protected float tauxImpot;
    public Employe(String matricule, String nom, String prenom, float tauxImpot) {
        super();
        this.matricule = matricule;
        this.nom = nom;
        this.prenom = prenom;
        this.tauxImpot = tauxImpot;
    }
    public void afficher() {
        System.out.println("Matricule :" + matricule);
        System.out.println("Nom :" + nom);
        System.out.println("Prenom :" + prenom);
        System.out.println("Taux Impot :" + tauxImpot);
        System.out.println("Salaire Brut :" + getSalaireBrut());
        System.out.println("Impot :" + getImpot());
        System.out.println("Salaire Net :" + getSalaireNet());
    }
    public float getImpot() {
        return getSalaireBrut() * tauxImpot;
    }
    public float getSalaireNet() {
        return getSalaireBrut() - getImpot();
    }
    public String getMatricule() {
        return matricule;
    }
}
package societe;
public class Ouvrier extends Employe {
    protected int nbHeure;
    protected float prixHeure;
    public Ouvrier(String matricule, String nom, String prenom,
        float tauxImpot, int nbHeure, float prixHeure) {
        super(matricule, nom, prenom, tauxImpot);
        this.nbHeure = nbHeure;
        this.prixHeure = prixHeure;
    }
    public float getSalaireBrut() {
        return nbHeure * prixHeure;
    }
    public void afficher() {
        super.afficher();
        System.out.println("Nb Heure :" + nbHeure);
        System.out.println("Prix Heure :" + prixHeure);
    }
}
```

```

    }
}
package societe;
public class Technicien extends Employe {
    protected float salaireMensuel;
    protected float prixHeureSupp;
    protected int nbHeureSupp;
    protected String specialite;

    public Technicien(String matricule, String nom, String prenom,
        float tauxImpot, float salaireMensuel, float prixHeureSupp,
        int nbHeureSupp, String specialite) {
        super(matricule, nom, prenom, tauxImpot);
        this.salaireMensuel = salaireMensuel;
        this.prixHeureSupp = prixHeureSupp;
        this.nbHeureSupp = nbHeureSupp;
        this.specialite = specialite;
    }
    public float getSalaireBrut() {
        return salaireMensuel + nbHeureSupp * prixHeureSupp;
    }
    public void afficher() {
        super.afficher();
        System.out.println("Salaire Mensuel :" + salaireMensuel);
        System.out.println("Nb Heure Supp :" + nbHeureSupp);
        System.out.println("Prix Heure Supp :" + prixHeureSupp);
        System.out.println("Specialite :" + specialite);
    }
}
package societe;
public class Ingenieur extends Employe {
    protected float salaireMensuel;
    protected float prime;
    protected String specialite;
    public Ingenieur(String matricule, String nom, String prenom,
        float tauxImpot, float salaireMensuel, float prime,
        String specialite) {
        super(matricule, nom, prenom, tauxImpot);
        this.salaireMensuel = salaireMensuel;
        this.prime = prime;
        this.specialite = specialite;
    }
    public float getSalaireBrut() {
        return salaireMensuel + prime;
    }
    public void afficher() {
        super.afficher();
        System.out.println("SalaireMensuel :" + salaireMensuel);
        System.out.println("Prime :" + prime);
        System.out.println("Specialite :" + specialite);
    }
}
package societe;
public class Societe {
    public static final int MAX_EMPLOYE = 1000;
    private String nom;
    private Employe[] tEmploye;
    private int nbEmploye;

    public Societe(String nom) {
        this.nom = nom;
        tEmploye = new Employe[MAX_EMPLOYE];
        nbEmploye = 0;
    }
    public int getNbEmploye() {
        return nbEmploye;
    }
}

```



```

public boolean ajouter(Employe employe) {
    for (int i = 0; i < tEmploye.length; i++)
        if (tEmploye[i] == null) {
            tEmploye[i] = employe;
            nbEmploye++;
            return true;
        }
    return false;
}

public boolean supprimer(int indice) {
    if (indice >= 0 && indice < tEmploye.length) {
        if (tEmploye[indice] != null) {
            tEmploye[indice] = null;
            nbEmploye--;
            System.out.println("Employé supprimé...");
            return true;
        }
    }
    return false;
}

public void supprimer(String matricule) {
    System.out.println("Recherche de la matricule : " + matricule);
    for (int i = 0; i < tEmploye.length; i++)
        if (tEmploye[i] != null)
            if (tEmploye[i].getMatricule().equals(matricule)) {
                supprimer(i);
                return;
            }
    System.out.println("Aucun employé pour cette matricule...");
}

public void rechercher(String matricule) {
    System.out.println("Recherche de la matricule : " + matricule);
    for (int i = 0; i < tEmploye.length; i++)
        if (tEmploye[i] != null)
            if (tEmploye[i].getMatricule().equals(matricule)) {
                System.out.println("Employé trouvé...");
                tEmploye[i].afficher();
                return;
            }
    System.out.println("Aucun employé pour cette matricule...");
}

public void supprimerTout() {
    for (int i = 0; i < tEmploye.length; i++)
        supprimer(i);
}

public float getSommeSalaireBrut() {
    int somme = 0;
    for (int i = 0; i < tEmploye.length; i++)
        if (tEmploye[i] != null)
            somme += tEmploye[i].getSalaireBrut();
    return somme;
}

public float getSommeImpot() {
    int somme = 0;
    for (int i = 0; i < tEmploye.length; i++)
        if (tEmploye[i] != null)
            somme += tEmploye[i].getImpot();
    return somme;
}

public float getSommeSalaireNet() {
    int somme = 0;
    for (int i = 0; i < tEmploye.length; i++)
        if (tEmploye[i] != null)
            somme += tEmploye[i].getSalaireNet();
    return somme;
}

```



```

    public float getSommeSalaireNetIngenieur() {
        int somme = 0;
        for (int i = 0; i < tEmploye.length; i++)
            if (tEmploye[i] instanceof Ingenieur)
                somme += tEmploye[i].getSalaireBrut();
        return somme;
    }

    public void afficher() {
        System.out.println("-----");
        System.out.println("                " + nom);
        System.out.println("-----");
        System.out.println();
        System.out.println("Nb Employe                :" + getNbEmploye());
        System.out.println("Somme Salaire Brut        :"
            + getSommeSalaireBrut());
        System.out.println("Somme Impot                :" + getSommeImpot());
        System.out.println("Somme Salaire Net         :"
            + getSommeSalaireNet());
        System.out.println("Somme Salaire Net Ingénieur :"
            + getSommeSalaireNetIngenieur());
        if (getNbEmploye() > 0) {
            System.out.println("La liste des employés est : ");
            System.out.println("-----");
            for (int i = 0; i < tEmploye.length; i++)
                if (tEmploye[i] != null)
                    tEmploye[i].afficher();
            System.out.println("-----");
        }
        System.out.println("-----");
    }
}

package societe;
public class TestSociete {
    public static void main(String[] args) {
        Societe societe;
        societe = new Societe("HorizonPOO");
        Ingenieur ingenieur;
        ingenieur = new Ingenieur("ING001", "Mohamed", "Ali", 0.06f, 850, 250,"Informatique");
        ingenieur.afficher();
        Technicien technicien;
        technicien = new Technicien("TECH001", "Omar", "Saleh", 0.5f, 400f, 5,8,
        "Informatique");
        technicien.afficher();
        Ouvrier ouvrier;
        ouvrier = new Ouvrier("OUV001", "Ahmed", "Sabeur", 0.05f, 200, 1.5f);
        ouvrier.afficher();
        societe.ajouter(ingenieur);
        societe.ajouter(technicien);
        societe.ajouter(ouvrier);
        societe.ajouter(new Ingenieur("ING002", "Ali", "Omar", 0.06f, 650, 150,"Gestion"));
        societe.ajouter(new Technicien("TECH002", "Saleh", "Sabeur", 0.5f,
        400f, 5, 8, "Gestion"));
        societe.ajouter(new Ouvrier("OUV002", "Ahmed", "Omar", 0.05f, 220,1.5f));
        societe.supprimer(1);    societe.supprimer("ING001");
        societe.supprimerTout();    societe.afficher();
    }
}

```

