

## Solutions

### Solution 1.

- La classe Livre :

```
public class Livre {

    private String titre , auteur;
    private int nbPages;
    private double prix;
    private boolean prixFixe;
    private final static int PRIX_NON_FIXE = -1;

    public Livre(String auteur, String titre , int nbPages, double prix) {
        this.auteur = auteur;
        this.titre = titre ;
        // l'appel au modificateur est plus sur
        setNbPages(nbPages);
        setPrix(prix);
    }

    public Livre(String auteur, String titre , int nbPages) {
        this.auteur = auteur;
        this.titre = titre ;
        // l'appel au modificateur est plus sur
        setNbPages(nbPages);
        prix = PRIX_NON_FIXE;
        prixFixe = false;
    }

    public String getAuteur() {
        return auteur;
    }

    public String getTitre() {
        return titre;
    }

    public int getNbPages() {
        return nbPages;
    }

    public double getPrix() {
        return prix;
    }
}
```

```

public void setNbPages(int n) {
    if (n < 0)
        System.err.println("Erreur.: nombre de pages negatif.");
    else
        nbPages = n;
}

public void setPrix(double prix) {
    if(prixFixe)
        System.err.println("Erreur.: prix deja fixe.");
    else {
        if (prix < 0)
            System.err.println("Erreur.: prix negatif.");
        else {
            this.prix = prix;
            prixFixe = true;
        }
    }
}

public void afficheToi() {
    System.out.print(this);
}

public String toString() {
    String resultat ;
    if (!prixFixe)
        resultat = "Prix pas encore fixe";
    else
        resultat = prix + " euros";
    return "Livre\\"" + titre + "\" de\\"" + auteur + "\" -\"" + resultat;
}

public int compare(Livre l) {
    if (getNbPages() == l.getNbPages())
        return 0;
    if (getNbPages() > l.getNbPages())
        return 1;
    return -1;
}
}

```

– La classe TestLivre :

```

public class TestLivre {

```

```

public static void main(String[]args) {
    Livre l1 = new Livre("Victor_Hugo",
                        "Les_miserables",
                        652,
                        8.20);
    Livre l2 = new Livre("Boris_Vian",
                        "L'ecume_des_jours",
                        220);

    System.out.println(l1);
    System.out.println(l2);
    l2.setPrix (4.50);
    System.out.println(l2);
    l2.setPrix (7.50);
    System.out.println(l2);

}
}

```

### Solution 3.

- La classe Etagere :

```

public class Etagere {

    private Livre[] etagere;
    private int nbLivres;

    private final static int TAILLE = 15;

    public Etagere() {
        etagere = new Livre[TAILLE];
        nbLivres = 0;
    }

    public static int taille () {
        return TAILLE;
    }

    public int nbLivres() {
        return nbLivres;
    }

    public void ajouter(Livre l) {
        if(nbLivres == TAILLE)
            System.err.println("Pas_de_place_sur_l'etagere");
        else {
            etagere[nbLivres] = l;
            nbLivres++;
        }
    }
}

```

```
}  
}
```

```
public Livre getLivre(int i) {  
    if(i > nbLivres)  
        return null;  
    else  
        return etagere[i - 1];  
}
```

```
public int chercher(String auteur, String titre) {  
    int i;  
    for(i = 0; i < nbLivres; i++)  
        if((etagere[i].getAuteur()).equals(auteur) &&  
            (etagere[i].getTitre()).equals(titre))  
            return i + 1;  
    return 0;  
}
```

```
public Livre[] chercherTous(String auteur, String titre) {  
    int i, j, cpt;  
    Livre[] resultat;  
  
    for(i = 0, cpt = 0; i < nbLivres; i++)  
        if((etagere[i].getAuteur()).equals(auteur) &&  
            (etagere[i].getTitre()).equals(titre))  
            cpt++;  
  
    if (cpt == 0)  
        return null;  
  
    resultat = new Livre[cpt];  
  
    for(i = 0, cpt = 0, j = 0; i < nbLivres && j < cpt; i++)  
        if((etagere[i].getAuteur()).equals(auteur) &&  
            (etagere[i].getTitre()).equals(titre)) {  
            resultat[j] = etagere[i];  
        }  
  
    return resultat;  
}
```

```
public Livre supprimerLivre(int position) {  
    if (position > nbLivres) {  
        System.err.println("Pas de livre a cet endroit.");  
    }  
}
```

```

        return null;
    }
    else {
        int i;
        Livre resultat = etagere[position - 1];
        for(i = position; i < nbLivres; i++)
            etagere[i - 1] = etagere[i];
        etagere[nbLivres - 1] = null;
        nbLivres--;
        return resultat;
    }
}

```

```

public Livre supprimerLivre(String auteur, String titre ) {
    int position = chercher(auteur, titre );
    return (position == 0 ? null : supprimerLivre(position));
}

```

```

public String toString() {
    int i;
    String resultat = new String();
    for(i = 0; i < nbLivres; i++) {
        resultat += etagere[i].toString() + "\n";
        resultat += "-----\n";
    }
    return resultat;
}

```

}

– La classe TestEtagere :

```

public class Etagere {

    private Livre[] etagere;
    private int nbLivres;

    private final static int TAILLE = 15;

    public Etagere() {
        etagere = new Livre[TAILLE];
        nbLivres = 0;
    }

    public static int taille () {
        return TAILLE;
    }
}

```

```

public int nbLivres() {
    return nbLivres;
}

public void ajouter(Livre l) {
    if(nbLivres == TAILLE)
        System.err.println("Pas de place sur l'etagere");
    else {
        etagere[nbLivres] = l;
        nbLivres++;
    }
}

public Livre getLivre(int i) {
    if(i > nbLivres)
        return null;
    else
        return etagere[i - 1];
}

public int chercher(String auteur, String titre) {
    int i;
    for(i = 0; i < nbLivres; i++)
        if((etagere[i].getAuteur().equals(auteur) &&
            (etagere[i].getTitre().equals(titre)))
            return i + 1;
    return 0;
}

public Livre[] chercherTous(String auteur, String titre) {
    int i, j, cpt;
    Livre[] resultat;

    for(i = 0, cpt = 0; i < nbLivres; i++)
        if((etagere[i].getAuteur().equals(auteur) &&
            (etagere[i].getTitre().equals(titre)))
            cpt++;

    if (cpt == 0)
        return null;

    resultat = new Livre[cpt];

    for(i = 0, cpt = 0, j = 0; i < nbLivres && j < cpt; i++)

```

```

        if((etagere[i].getAuteur()).equals(auteur) &&
            (etagere[i].getTitre()).equals(titre)) {
            resultat[j] = etagere[i];
        }

    }

    return resultat;
}

public Livre supprimerLivre(int position) {
    if (position > nbLivres) {
        System.err.println("Pas de livre a cet endroit.");
        return null;
    }
    else {
        int i;
        Livre resultat = etagere[position - 1];
        for(i = position; i < nbLivres; i++)
            etagere[i - 1] = etagere[i];
        etagere[nbLivres - 1] = null;
        nbLivres--;
        return resultat;
    }
}

public Livre supprimerLivre(String auteur, String titre) {
    int position = chercher(auteur, titre);
    return (position == 0 ? null : supprimerLivre(position));
}

public String toString() {
    int i;
    String resultat = new String();
    for(i = 0; i < nbLivres; i++) {
        resultat += etagere[i].toString() + "\n";
        resultat += "-----\n";
    }
    return resultat;
}
}

```