```java
public class Book
{
     private int id;
     private String title;
     private String author;
     private double price;

     public Book(int pID, String pTitle, String pAuthor, double pPrice)
     {
          this.id = pID;
          this.title = pTitle;
          this.author = pAuthor;
          this.price = pPrice;
     }

     public String getTitle()
     {
          return title;
     }

     public String getAuthor()
     {
          return author;
     }

     public double getPrice()
     {
          return price;
     }

     public int getID()
     {
          return id;
     }

     public String toString()
     {
          return "[" + id + "] -> " + author + " - " + title + " (" + price + "€)";
     }
}
```

```java
import java.util.ArrayList;
public class Library
{
     private ArrayList<Book> alBooks = new ArrayList<>();

     public void addBook(Book pBook)
     {
          alBooks.add(pBook);
     }

     public int countBooks()
     {
          return alBooks.size();
     }

     public Object[] toArray()
     {
          return alBooks.toArray();
     }

     public Double searchMostExpensivePrice()
     {
          Book b;

          if (alBooks.size() == 0)
               return null;

          double max = alBooks.get(0).getPrice();
          for (int i = 1; i < alBooks.size(); i++)
          {
               b = alBooks.get(i);
               if (b.getPrice() > max)
                    max = b.getPrice();
          }

          return max;
     }

     public String searchAuthorOfSmallestID()
     {
          Book b;

          if (alBooks.size() == 0)
               return null;

          Book min = alBooks.get(0);
          for (int i = 1; i < alBooks.size(); i++)
          {
               b = alBooks.get(i);
               if (b.getID() < min.getID())
                    min = b;
          }

          return min.getAuthor();
     }

     public Book searchFirstBookOfAuthor(String pAuthor)
     {
          int i = 0;
          boolean found = false;
          while (!found && (i < alBooks.size()))
          {
               if (alBooks.get(i).getAuthor().equals(pAuthor))
                    found = true;
               else
                    i++;
          }

          if (found)
               return alBooks.get(i);
          else
               return null;
     }

     public int countBooksOfAuthor(String pAuthor)
     {
          int count = 0;
          for (int i = 0; i < alBooks.size(); i++)
          {
               if (alBooks.get(i).getAuthor().equals(pAuthor))
                    count++;
          }
          return count;
     }
```

// ... suite page suivante ...

```
    public void sortByTitle()
    {
        for (int i=0; i<alBooks.size()-1; i++)
        {
            String min = alBooks.get(i).getTitle();
            int posMin = i;
            for (int j=i+1; j<alBooks.size(); j++)
            {
                if (alBooks.get(j).getTitle().compareTo(min) < 0)
                {
                    min = alBooks.get(j).getTitle();
                    posMin = j;
                }
            }
            if (i != posMin)
            {
                Book temp = alBooks.get(i);
                alBooks.set(i, alBooks.get(posMin));
                alBooks.set(posMin, temp);
            }
        }
    }

    public void deleteBooks(double price)
    {
        for (int i=alBooks.size()-1; i>=0; i--)
        {
            if (alBooks.get(i).getPrice() >= price)
                alBooks.remove(i);
        }
    }
}
```