//DO NOT EDIT OR ADD ANY CODE HERE

public class Book {

private String bookId;

private String bookName;

private String bookType;

private String bookCategory;

private String bookEdition;

public Book() {

super();

}

public Book(String bookId, String bookName, String bookType, String bookCategory, String bookEdition) {

super();

this.bookId = bookId;

this.bookName = bookName;

this.bookType = bookType;

this.bookCategory = bookCategory;

this.bookEdition = bookEdition;

}

public String getBookId() {

return bookId;

}

public void setBookId(String bookId) {

this.bookId = bookId;

}

public String getBookName() {

return bookName;

}

public void setBookName(String bookName) {

this.bookName = bookName;

}

public String getBookType() {

return bookType;

}

public void setBookType(String bookType) {

this.bookType = bookType;

}

public String getBookEdition() {

return bookEdition;

}

public void setBookEdition(String bookEdition) {

this.bookEdition = bookEdition;

}

public String getBookCategory() {

return bookCategory;

}

public void setBookCategory(String bookCategory) {

this.bookCategory = bookCategory;

}

}

import java.util.ArrayList;

import java.util.List;public class BookUtility implements Runnable{

private List<Book> bookList = new ArrayList<Book>();

private String searchbookName;

private int counter;

//FILL THE CODE HERE

public List<Book> getBookList(){

return bookList;

}

public String getSearchbookName(){

return searchbookName;

}

public void setBookList(List<Book> a){

this.bookList=a;

}

public void setSearchbookName(String x){

this.searchbookName=x;

}

public int getCounter(){

return counter;

}

public void setCounter(int counter){

this.counter=counter;

}

public void toValidateBookType(Book obj) throws InvalidBookException

{

//FILL THE CODE

if (obj.getBookType().compareToIgnoreCase("Engineering")==0){

bookList.add(obj);

}else{

throw new InvalidBookException("Book type Invalid");

}

}

public void run()

{

//FILL THE CODE HERE

int count=0;

for(int j=0;j<bookList.size();j++){

if (bookList.get(j).getBookName().compareToIgnoreCase(searchbookName)==0){

count=count+1;

}

}

if (count==0) {

System.out.println("No Books found");

}else {

System.out.println("Count of books in the library with the book name "+searchbookName.toUpperCase()+" is "+count);

}}}

public class InvalidBookException extends Exception{

//FILL THE CODE

String s;

public InvalidBookException (String s){

super(s);

}}

import java.util.ArrayList;

import java.util.List;

import java.util.Scanner;public class UserInterface {

public static void main(String [] args)

{ Scanner sc=new Scanner(System.in);

BookUtility ans = new BookUtility();

// FILL THE CODE HERE

System.out.println("Enter the number of entries");

int a = Integer.parseInt(sc.nextLine());

System.out.println("Enter the bookId/bookName/bookType/bookCategory/bookEdition");

String [] x = new String[a];

List<Book> clm=new ArrayList<Book>();

boolean flag = false;

for (int j =0;j<a;j++){

x[j] =sc.nextLine();

String[] p = x[j].split("/");

Book cd= new Book();

cd.setBookId(p[0]);

cd.setBookName(p[1]);

cd.setBookType(p[2]);

cd.setBookCategory(p[3]);

cd.setBookEdition(p[4]);

try{

ans.toValidateBookType(cd);

clm.add(cd);

}

catch(InvalidBookException e){

flag=true;

System.out.println(e.getMessage());

}

}

if (flag==false){

ans.setBookList(clm);

System.out.println("Print the book details");

for(int y=0;y<clm.size();y++){

System.out.println("bookId="+clm.get(y).getBookId()+", bookName="+clm.get(y).getBookName()+", bookType="+clm.get(y).getBookType());

}

System.out.println("Enter book name to search");

String hh = sc.nextLine();

ans.setSearchbookName(hh);

Thread myThread = new Thread(ans);

myThread.run();

}

}

}