

# Bookshelf - Interfacce ed eccezioni

## 1. Archivio di Libri

Scrivere un'applicazione Java per la gestione di un archivio di libri, in particolare libri cartacei ed e-book.

Un libro (Book) possiede i seguenti attributi

- titolo
- autore
- codice alfanumerico

L'archivio è di fatto una collezione che deve poter contenere allo stesso tempo libri cartacei (PaperBook) ed e-book (Ebook) e deve possedere le seguenti funzionalità:

- aggiunta di un nuovo libro in archivio
- rimozione di un libro in archivio
- ricerca di libri all'interno della collezione sia per titolo che per autore
  - restituendo oggetti trovati (findAllBy)
- ricerca di un libro all'interno della collezione sia per titolo che per autore
  - restituendo il primo indice di posizione in archivio (searchOneBy)
  - restituendo il primo libro in archivio (findOneBy)

**Importante constraint strutturale:** stabilita una regola di uguaglianza (override equals() e hashCode()) per i libri, la collezione **non** deve contenere libri uguali

Nel caso in cui si rimuove un libro non esistente o si aggiunga un libro in collezione già esistente sollevare le opportune eccezioni

## 2. Applicazione

Considerare nell'applicazione, la gestione di utenti, ciascuno con la propria collezione di libri (tipo archivio al punto precedente) che possono essere cartacei (dunque prestabili) o e-book (dunque condivisibili).

L'utente deve avere la possibilità di prestare i propri libri cartacei ad altri utenti.

Per ogni libro cartaceo può esistere uno ed un solo prestito per volta.

Un libro cartaceo può essere prestato (**Loanable**) e deve essere possibile ottenere informazione sul prestito (**Loan**), con informazioni su chi attualmente lo detiene e sulle date di inizio e fine prestito.

L'utente deve avere la possibilità di impostare i propri e-book come condivisibili (**Sharable**).

Un e-book può essere condiviso, prevedendo che non più di MAXSHARING utenti possano dividerlo

Naturalmente sia per i libri Loanable che quelli Sharable si devono prevedere operazioni di restituzione

### **Precisazioni**

Nell'applicazione, ogni utente e' caratterizzato da attributi:

- Nome
- ID (può essere un INT o un String Code)
- collezione di libri (ogni utente ha la sua collezione di libri)

il prestito, come anche la condivisione, ovviamente **non** implicano la rimozione dall'archivio del possessore

**Non** si richiede implementazione delle operazioni il codice utente che si occupa di:

- richiesta di condivisione libro
- ricezione di un libro in prestito/o condivisione

### **3. MAIN**

- istanziare due utenti (A e B), ognuno di essi con una collezione di libri a dimensione arbitraria
- Collezione dell'utente A contiene almeno 3 libri, di cui almeno uno prestabile e uno condivisibile
- Utente A deve cercare dentro la propria collezione un libro per titolo e prestarlo all'utente B. Il successo dell'operazione produce un aggiornamento nello stato del libro prestato. Questo stato deve essere stampato a video
- Nel caso in cui il libro non e' stato trovato, sollevare un eccezione che deve essere catturata
- Utente A vuole leggere un libro. Ci chiediamo se B possiede il libro e può prestarlo o dividerlo con A
- Utente B deve cercare dentro la propria collezione un libro per titolo e renderlo condiviso
  - se non trovato sollevare un eccezione