

# Biblioteca E-book

Progetto d'esame per il corso di Programmazione a Oggetti

Prof. Giacomo Cabri

## Traccia di progetto

### REGOLE PER LO SVOLGIMENTO

Il progetto deve essere svolto usando il linguaggio Java e possedere i seguenti requisiti implementativi:

- Essere dotato di **interfaccia grafica** tramite cui interagire con il programma stesso.
- Sfruttare i meccanismi di **incapsulamento, ereditarietà e polimorfismo**.
- Per l'ereditarietà è possibile sfruttare **classi astratte e interfacce**; si considerano **escluse** le relazioni di ereditarietà diretta da classi di biblioteca Java.
- Sfruttare le classi di sistema Java per la gestione dell'**input/output**.
- Utilizzare i **generics**.
- Il programma deve essere eseguibile da **linea di comando**.

Il software deve essere accompagnato da pagine di **documentazione HTML** (ivi incluse le pagine generate tramite Javadoc) che descrivano le scelte di progetto effettuate e la struttura del sistema software.

Nel seguito del testo, i paragrafi evidenziati in **azzurro** sono **facoltativi**, e servono per differenziare il voto.

Lo svolgimento della parte **obbligatoria** contribuisce al voto per **22 punti**. Il contributo delle parti facoltative è riportato nelle rispettive descrizioni. Si noti che il **punteggio massimo** rimane comunque **30/30**.

### DESCRIZIONE DEL PROGETTO

La disponibilità di libri in formato elettronico è sempre maggiore, e dà la possibilità di leggere i propri libri su diversi dispositivi. È quindi importante avere a disposizione uno strumento che permetta la gestione e la fruizione degli e-book in diversi formati.

Il presente progetto si propone di descrivere e sviluppare una applicazione che abbia le seguenti funzionalità (dettagliate nei paragrafi successivi):

- Gestione di una biblioteca di e-book;
- Salvataggio e caricamento della biblioteca;
- Lettura dei libri;
- Ricerca di informazioni nella biblioteca;
- **Stampa delle informazioni contenute nella biblioteca.**

#### Gestione di una biblioteca

La gestione della biblioteca prevede la visualizzazione della lista dei libri e alcune operazioni su di essa.

La visualizzazione deve essere in formato tabellare, elencando un libro per riga, con le relative informazioni:

- Titolo

- Autore
- Anno
- Percorso del file
- Numero di pagine

L'utente deve avere la possibilità di aggiungere, modificare e cancellare libri dalla biblioteca.

La prima operazione è l'**aggiunta** di un libro alla biblioteca. L'utente deve specificare il percorso del file sul disco, e inserire le diverse informazioni relative al libro. Si prevedano **almeno 3 formati** come ad esempio PDF, epub, HTML, mobi, ...

Opzionalmente, si deve dare la possibilità all'utente di selezionare più file o tutti i file di una directory che hanno una data estensione [2 punti].

Una seconda operazione è la **modifica** delle informazioni di un libro, che chiaramente può riguardare solo una parte se alcune informazioni si ritengono non modificabili (ad es. il percorso del file).

Infine, l'utente deve avere la possibilità di **eliminare** un libro dalla biblioteca (non dal filesystem).

### Salvataggio e caricamento della biblioteca

L'utente deve avere la possibilità di **salvare** la biblioteca (cioè l'elenco dei contenuti) su un file, specificandone il **nome**, e di **ricaricare** la biblioteca specificando il **nome** del file salvato in precedenza. Nel caso in cui si tenti di salvare la biblioteca in un file che esiste già, deve essere chiesto all'utente se desidera **sovrascrivere** il file esistente [2 punti].

### Lettura dei libri

Per la lettura dei libri, ogni formato può avere un comportamento diverso. Le possibilità sono:

1. Chiedere all'utente quale **applicazione esterna** vuole utilizzare per leggere un formato e salvare questa informazione nella configurazione del programma;
2. Usare l'applicazione di **default** associata a un formato;
3. **Implementare** un visualizzatore sfruttando una libreria software esistente.

Si implementi un visualizzatore per **almeno** un formato scelto.

Opzionalmente, si implementi un visualizzatore anche per un **secondo** formato a scelta [5 punti].

Per implementare questa caratteristica e le funzionalità collegate si suggerisce di utilizzare il **polimorfismo in Java**.

### Ricerca di informazioni nella biblioteca

L'utente deve avere la possibilità di effettuare delle **ricerche** nella biblioteca. La ricerca si basa su **testo libero** che può essere una parte del testo contenuto in una informazione.

La ricerca deve evidenziare la **prima cella/riga** che contiene il testo cercato; l'applicazione deve permettere all'utente di **continuare** la ricerca per evidenziare man mano le celle/righe **successive** che rispondono ai requisiti (ad esempio, tramite un bottone "successivo", o usando lo stesso bottone della ricerca).

### Stampa delle informazioni contenute nella biblioteca

Si dia all'utente la possibilità di **stampare** l'intera biblioteca. Si sfruttino le classi di libreria Java per stampare tramite una delle stampanti configurate dal sistema operativo [1 punto].

## MATERIALE UTILE

- Interfaccia Java `java.awt.print.Printable` e classe Java `java.awt.print.PrinterJob` per stampare su stampante. Si possono trovare tutorial online.
- Librerie <http://www.siegmann.nl/epublib> e <https://github.com/psiegman/epublib> per la gestione di file in formato epub.
- Librerie <http://pdfbox.apache.org/> e <http://stefanochizzolini.it/it/projects/clown/index.html> per la visualizzazione di e-book in PDF.