



Biblioteca



Simone Ruggeri
Lorenzo Ferrari



Introduzione



- Questo progetto riguarda la creazione di una biblioteca virtuale. L'obiettivo è fornire strumenti efficienti per catalogare i libri, monitorare i prestiti e gestire le risorse della biblioteca.
- L'interfaccia grafica sarà rivolta all'addetto alla biblioteca, che sarà l'unico ad averne accesso dal suo PC personale.
- Il sistema consentirà al bibliotecario di aggiungere e rimuovere libri dal catalogo, tracciare i prestiti effettuati dai membri della biblioteca e verificare eventuali scadenze non rispettate.
- Questo approccio centralizzato mira a migliorare l'organizzazione e l'efficienza operativa della biblioteca.



CRC: Libro e ElencoLibri

NAME CLASS: Libro	
RESPONSIBILITY	COLLABORATION
- tenere traccia del titolo, autore, genere, anno e se è in prestito o è stato ricercato.	
- permette di cambiare lo stato di disponibilità in caso di prenotazione o restituzione	

NAME CLASS: ElencoLibri	
RESPONSIBILITY	COLLABORATION
- tenere traccia di un elenco di libri	-Libro
- permettere di aggiungere, rimuovere, riordinare, ricercare libri dell'elenco	-Libro
- permettere di capire se un libro è presente o meno nell'elenco	-Libro



CRC: Prenotazione e Data

NAME CLASS: Prenotazione	
RESPONSIBILITY	COLLABORATION
- tenere traccia di un libro prenotato e di una data di scadenza	-Data -Libro

NAME CLASS: Data	
RESPONSIBILITY	COLLABORATION
- tenere traccia di giorno, mese, anno	
- calcolare i giorni trascorsi da una certa data	
- calcolare i giorni mancanti ad una determinata scadenza	



CRC: Biblioteca e Utente

NAME CLASS: Biblioteca	
RESPONSIBILITY	COLLABORATION
- tenere traccia di una lista di utenti e di vari elenchi di libri associati agli utenti	-Utente -ElencoLibri
- caricare e salvare tramite Json	-FileManager
- permette di aggiungere o rimuovere un Libro dal suo ElencoLibri	-Libro -ElencoLibri
- permette di aggiungere un utente nel caso in cui lo username non sia già presente	-Utente
- contrassegnare i libri come "ricercati"	-ElencoLibri -Libro

NAME CLASS: Utente	
RESPONSIBILITY	COLLABORATION
- tenere traccia di uno username e di una lista di prenotazioni	-Prenotazione
- permettere di prenotare e restituire a nome di un utente un libro	-Libro -Prenotazione
- capire se un utente ha già in prestito un libro o meno	-Libro



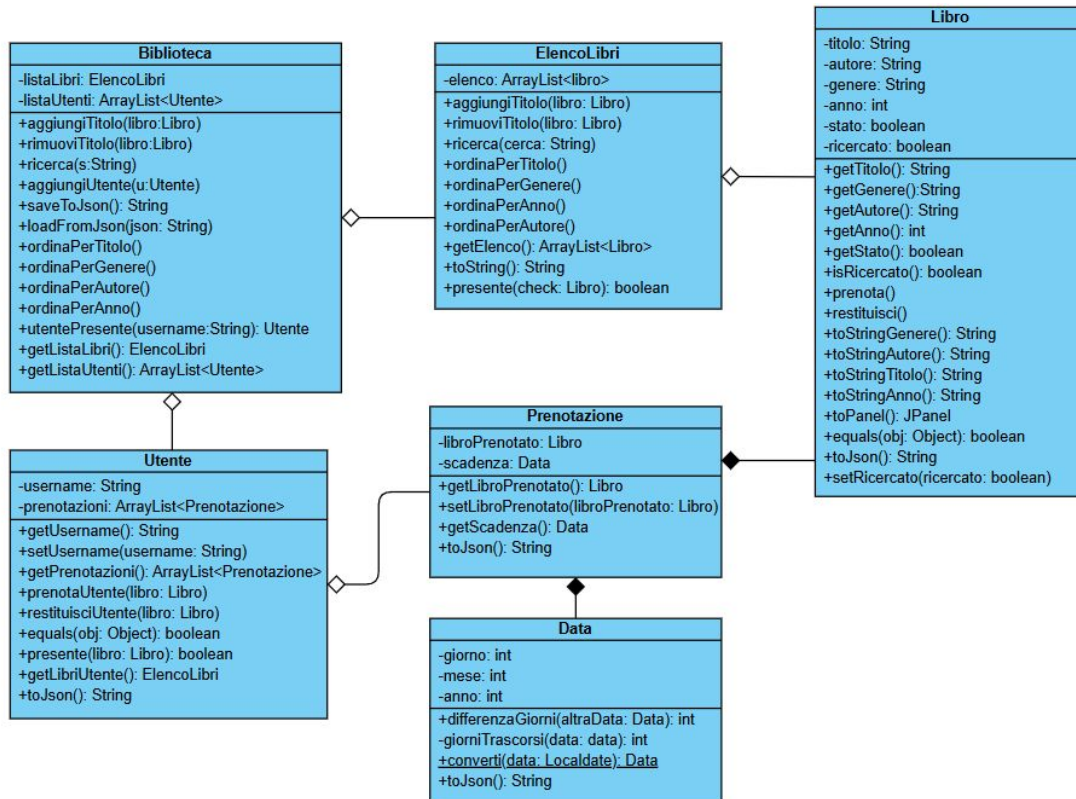
CRC: FileManager e BibliotecaTest

NAME CLASS: FileManager	
RESPONSIBILITY	COLLABORATION
- leggere da file	-Biblioteca
- scrivere su file	-Biblioteca

NAME CLASS: BibliotecaTest	
RESPONSIBILITY	COLLABORATION
- recuperare biblioteca da file	-Biblioteca -FileManager
- lanciare il programma	-Biblioteca

UML

- Diagramma UML che descrive la gestione dell'applicazione biblioteca senza la parte grafica e la gestione del salvataggio su file



Pacchetti

- **application:** contiene le classi che permettono di inizializzare la biblioteca
- **biblioteca:** contiene le classi addette alla gestione della biblioteca
- **GUI:** contiene le classi che gestiscono la parte grafica del progetto
- **settings:** contiene la classe Impostazioni, che viene consultata dalle altre classi

- ✓  application
 - >  BibliotecaTest.java
 - >  FileManager.java
- ✓  biblioteca
 - >  Biblioteca.java
 - >  Data.java
 - >  ElencoLibri.java
 - >  Libro.java
 - >  Prenotazione.java
 - >  Utente.java
- ✓  GUI
 - >  BibliotecaFrame.java
 - >  BibliotecaMenu.java
 - >  PannelloLibri.java
- ✓  settings
 - >  Impostazioni.java

Libro

> 📖 biblioteca

- La classe Libro contiene attributi legati alla descrizione di un libro: titolo, autore, genere e anno di pubblicazione.
- Esistono inoltre due attributi booleani: “stato”, che indica l’eventuale disponibilità del libro all’interno della biblioteca per un prestito, e “ricercato” che indica se il libro è stato ricercato attraverso il sistema di ricerca del programma.
- La classe contiene inoltre un metodo che restituisce un JPanel contenente tutte le informazioni relative al libro e un metodo per il salvataggio del file in Json.

Libro
-titolo: String -autore: String -genere: String -anno: int -stato: boolean -ricercato: boolean
+getTitolo(): String +getGenere():String +getAutore(): String +getAnno(): int +getStato(): boolean +isRicerca(): boolean +prenota() +restituisce() +toStringGenere(): String +toStringAutore(): String +toStringTitolo(): String +toStringAnno(): String +toPanel(): JPanel +equals(obj: Object): boolean +toJson(): String +setRicerca(ricercato: boolean)

Prenotazione

> 📖 biblioteca

- La classe Prenotazione ha due attributi: uno di tipo Libro, che contiene il libro che è stato prenotato dall'utente, e una scadenza, descritta sotto un tipo Data creato appositamente.
- Contiene un metodo per il salvataggio su file tramite Json e un metodo per impostare il libro prenotato

Prenotazione
-libroPrenotato: Libro -scadenza: Data
+getLibroPrenotato(): Libro +setLibroPrenotato(libroPrenotato: Libro) +getScadenza(): Data +toJson(): String

Utente

> 📖 biblioteca

- Un oggetto della classe Utente contiene un username e una lista di Prenotazioni
- Per la gestione delle prenotazioni ci sono due metodi, uno per prenotare un libro e uno per restituire un libro, che verrà cercato tra le prenotazioni effettuate dall'utente
- Per verificare la presenza di un libro tra le prenotazioni dell'utente esiste il metodo presente()
- Sono presenti inoltre un metodo che restituisce l'elenco dei libri in prestito all'utente in tipo ElencoLibri e un metodo per il salvataggio del file con Json

Utente
-username: String -prenotazioni: ArrayList<Prenotazione>
+getUsername(): String +setUsername(username: String) +getPrenotazioni(): ArrayList<Prenotazione> +prenotaUtente(libro: Libro) +restituisceUtente(libro: Libro) +equals(obj: Object): boolean +presente(libro: Libro): boolean +getLibriUtente(): ElencoLibri +toJson(): String

Data

> 📖 biblioteca

- La classe Data è una semplice classe con attributi giorno, mese ed anno, creata per evitare il problema del salvataggio della classe LocalDate
- Contiene un metodo che calcola i giorni trascorsi abbinato ad un metodo per calcolare i giorni di differenza con un'altra data
- Inoltre sono presenti un metodo static per convertire gli oggetti LocalDate e un metodo per il salvataggio file con Json

Data
-giorno: int -mese: int -anno: int
+differenzaGiorni(altraData: Data): int -giorniTrascorsi(data: data): int <u>+converti(data: Localdate): Data</u> +toJson(): String

ElencoLibri


> 📖 biblioteca

- La classe ElencoLibri permette di gestire comodamente un ArrayList di oggetti Libro
- Sono presenti metodi per aggiungere e rimuovere libri dall'elenco, controllare se un libro è presente e contrassegnarlo come "ricercato" dal sistema
- Esistono infine metodi che permettono di riordinare l'elenco in base al titolo, all'autore, al genere o all'anno di pubblicazione

ElencoLibri
-elenco: ArrayList<libro>
+aggiungiTitolo(libro: Libro) +rimuoviTitolo(libro: Libro) +ricerca(cerca: String) +ordinaPerTitolo() +ordinaPerGenere() +ordinaPerAnno() +ordinaPerAutore() +getElenco(): ArrayList<Libro> +toString(): String +presente(check: Libro): boolean



FileManager

>  application

- La classe FileManager gestisce il salvataggio su file e il recupero dalla memoria di un oggetto Biblioteca
- Utilizza la libreria open-source Json di Google per la serializzazione e la deserializzazione delle classi

FileManager

+leggiDaFile(filename: String): Biblioteca
+scriviSuFile(biblio: Biblioteca, filename: String)

BibliotecaTest


> application

- La classe BibliotecaTest prova a recuperare dalla directory indicata un file di salvataggio da convertire in oggetto Biblioteca
- Se non dovesse riuscirci, ne crea uno nuovo

BibliotecaTest
+main(a: String[])


```
public static void main(String[] a){
    FileManager filemanager = new FileManager();
    Biblioteca biblioteca = filemanager.leggiDaFile(Impostazioni.NOME_FILE_BIBLIOTECA);
    if (biblioteca == null)
        biblioteca = new Biblioteca();
    BibliotecaFrame frame = new BibliotecaFrame(Impostazioni.STRINGA_BIBLIOTECA, biblioteca);
    frame.setVisible(true);
    frame.setDefaultCloseOperation(JFrame.EXIT_ON_CLOSE);
}
```


Impostazioni

>  settings

- La classe Impostazioni è stata creata per inserire tutte le costanti del progetto e viene richiamata da tutte le classi

Impostazioni
<u>+NOME_FILE_BIBLIOTECA: String = "savings/biblioteca.json"</u>
<u>+TITOLO_PROGRAMMA: String = "BIBLIOTECA"</u>
<u>+FONT_TITOLO: String = "Times new roman"</u>
...

▼  savings

 biblioteca.json

BibliotecaFrame

> 🏠 GUI

- La classe BibliotecaFrame è la classe che estende JFrame e conterrà i vari pannelli
- Il metodo start() mostra a schermo il pannello iniziale e gestisce il pulsante di Start
- Il metodo menu() mostra a schermo il menu e il pannello principale dove verranno svolte le operazioni sulla biblioteca
- Il metodo salva() permette di richiamare un oggetto della classe FileManager nel momento in cui la finestra viene chiusa

BibliotecaFrame
-biblioteca: Biblioteca -filemanager: FileManager
+start() +salva() +menu()



BibliotecaMenu

> 🏠 GUI

- La classe BibliotecaMenu estende la classe JPanel e gestisce sette pulsanti con sette funzioni diverse
- La pressione di ogni pulsante comunica al pannello principale quale funzione la biblioteca dovrà svolgere

Visualizza Catalogo

Visualizza Utenti

Aggiungi Titolo

Rimuovi Titolo

Prenota Libro

Restituisci Libro

Esci

PannelloLibri

> 🏠 GUI

- La classe PannelloLibri è la classe che gestisce il pannello principale del frame
- In base alla selezione del menu, mostrerà il pannello corrispondente descritto nel metodo
- Viene tenuta nota del pannello mostrato con il parametro di tipo char “mode”

PannelloLibri
-biblioteca: Biblioteca -mode: char -selettoreOrdine: int
+prenota() +restituisce() +mostra() +ricerca(cerca: String) +aggiungi() +rimuovi() +utenti() +nuovoUtente() +salva() +vuoto(n: int) +pannello(oggetto: JComponent) +pannelloDimensionato(oggetto: JComponent) +resetPannelli()

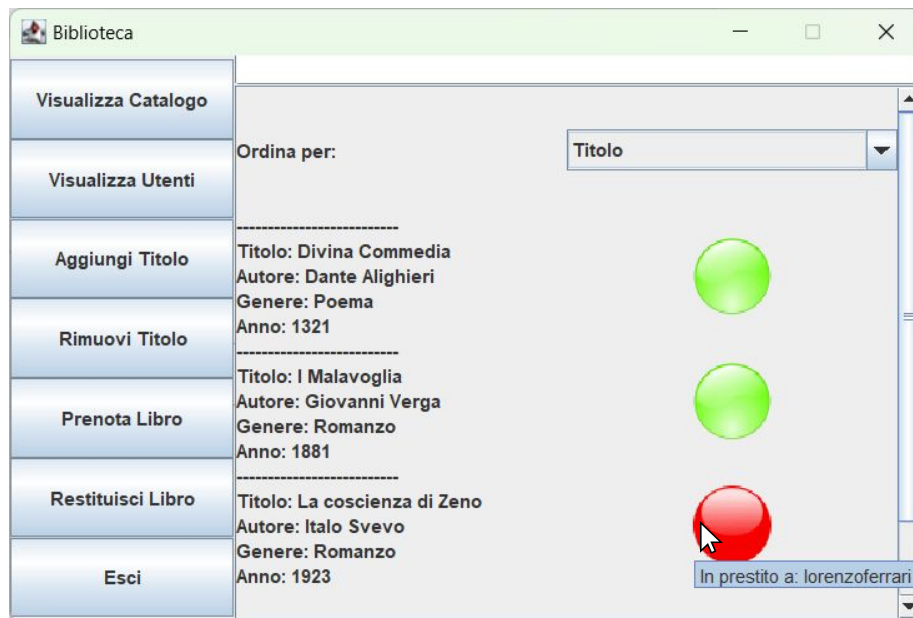
Pannello iniziale

- Tramite la pressione di un pulsante di start (posizionato al di sotto di un'immagine e di un'etichetta di testo per il titolo) è possibile passare alla visualizzazione del pannello menu (contenente dei pulsanti per la gestione delle varie attività) e del pannello principale.



Visualizza Catalogo

- Nel menù il primo pulsante presente avrà il compito di rendere visibile nel pannello principale il catalogo completo dei libri presenti nella biblioteca, con nome, titolo, genere, anno, e disponibilità del libro (se in prestito o disponibile).
- Possibilità di cercare un libro attraverso uno spazio di ricerca.
- Possibilità di ordinare il catalogo tramite un menù a “tendina” (per autore, anno, genere e titolo).



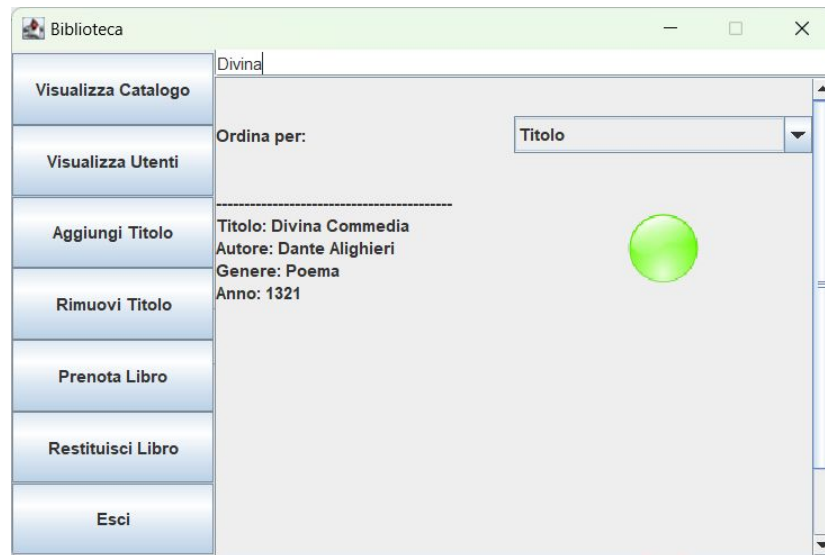
Visualizza Catalogo (ToolTip e ordina per)

- Sono presenti dei Tooltip quando si passa con il mouse sull'icona di stato (se verde viene indicata la disponibilità, se rossa viene indicato a che username quel libro è in prestito)
- Cambiando la modalità di visualizzazione tramite ComboBox viene riordinato il catalogo secondo quanto selezionato



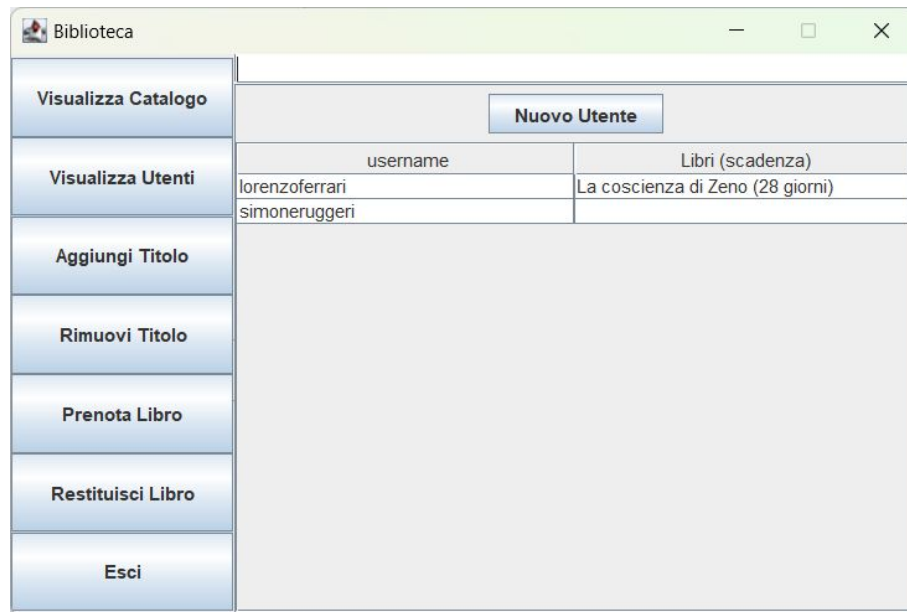
Ricerca Libro

- Tramite la barra di ricerca in alto, è possibile cercare e mostrare a schermo solo i libri che contengono la stringa inserita in uno dei campi (il tutto senza tenere in considerazione le lettere maiuscole)



Visualizza Utenti

- Il secondo pulsante del menu, mostra nel pannello principale tramite una JTable tutti gli username degli utenti registrati nella nostra biblioteca , l'elenco dei libri da loro prenotati, ognuno di essi seguito tra parentesi dai giorni mancanti alla scadenza.
- Nel caso di un nuovo utente, è possibile registrarlo e quindi inserirlo a tabella premendo il pulsante “nuovo utente” (mostrando a schermo un errore nel caso di username già in uso).



Nuovo Utente

Biblioteca

Visualizza Catalogo

Visualizza Utenti

Aggiungi Titolo

Rimuovi Titolo

Prenota Libro

Restituisci Libro

Esci

username:

nuovoutente

Aggiungi

Biblioteca

Visualizza Catalogo

Visualizza Utenti

Aggiungi Titolo

Rimuovi Titolo

Prenota Libro

Restituisci Libro

Esci

Nuovo Utente

username	Libri (scadenza)
lorenzoferrari	La coscienza di Zeno (28 giorni)
simoneruggeri	
nuovoutente	

Aggiungi Titolo

- La terza opzione del menu, serve per poter aggiungere all'interno della biblioteca un nuovo libro, inserendo le quattro voci in figura. Solo alla pressione del pulsante aggiungi il libro verrà inserito nel catalogo.

The screenshot shows a graphical user interface for a library system. On the left is a vertical menu with the following items: "Visualizza Catalogo", "Visualizza Utenti", "Aggiungi Titolo", "Rimuovi Titolo", "Prenota Libro", "Restituisci Libro", and "Esci". The "Aggiungi Titolo" option is highlighted. The main window area is titled "Biblioteca" and contains a form with the following fields: "Titolo:" with the value "1984", "Autore:" with the value "George Orwell", "Genere:" with the value "Romanzo", and "Anno:" with the value "1949". At the bottom right of the form is a blue button labeled "Aggiungi". A mouse cursor is pointing at this button.

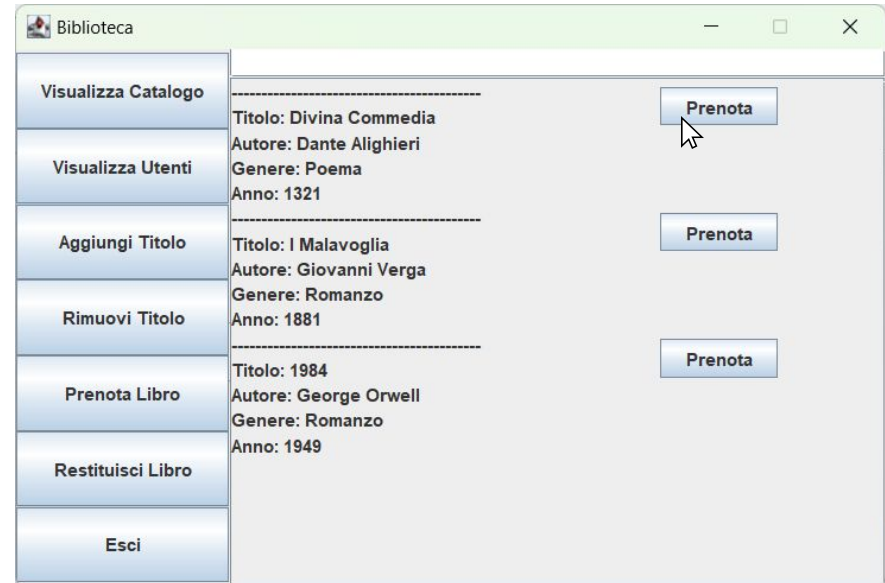
Rimuovi Titolo

- La quarta modalità è necessaria per poter rimuovere un libro definitivamente dal catalogo (questa azione è resa possibile solo in caso di disponibilità del libro e non nel caso in cui esso sia in prestito). Alla pressione del tasto rimuovi, il libro sarà cancellato dall'elenco.



Prenota Libro

- Il quinto pulsante, viene utilizzato per prenotare un libro a nome di un utente.
- Quindi dopo aver scelto quale tra i libri attualmente disponibili prenotare, sarà possibile, cliccando il pulsante, dichiarare poi a chi tra gli utente registrati assegnarlo.
- Dopo la scelta ne verrà cambiato lo stato e verrà aggiunto di conseguenza il libro all'elenco dei libri prenotati dall'utente.



Prenota Libro

Biblioteca

Visualizza Catalogo

Visualizza Utenti

Aggiungi Titolo

Rimuovi Titolo

Prenota Libro

Restituisci Libro

Esci

username:

lorenzoferrari

simoneruggeri

Biblioteca

Visualizza Catalogo

Visualizza Utenti

Aggiungi Titolo

Rimuovi Titolo

Prenota Libro

Restituisci Libro

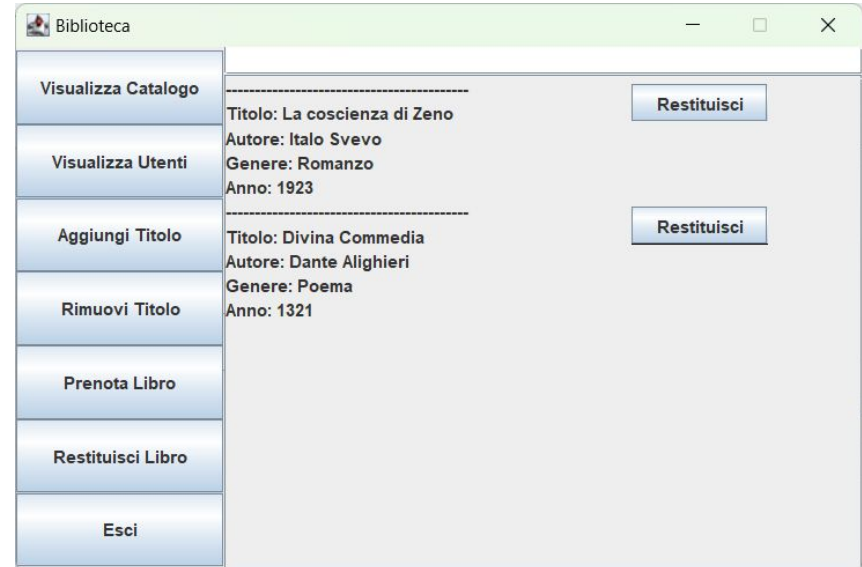
Esci

Nuovo Utente

username	Libri (scadenza)
lorenzoferrari	La coscienza di Zeno (28 giorni)
simoneruggeri	Divina Commedia(31 giorni)

Restituisci Libro

- L'ultima opzione serve invece per la restituzione, una volta consegnato il libro e premuto sul pulsante verrà modificato lo stato in "disponibile", permettendo una futura prenotazione.



Restituisci Libro

Biblioteca

Visualizza Catalogo

Visualizza Utenti

Aggiungi Titolo

Rimuovi Titolo

Prenota Libro

Restituisci Libro

Esci

Titolo: La coscienza di Zeno

Autore: Italo Svevo

Genere: Romanzo

Anno: 1923

Titolo: Divina Commedia

Autore: Dante Alighieri

Genere: Poema

Anno: 1321

Restituisci

Restituisci

In prestito a: simoneruggeri

Biblioteca

Visualizza Catalogo

Visualizza Utenti

Aggiungi Titolo

Rimuovi Titolo

Prenota Libro

Restituisci Libro

Esci

Nuovo Utente

username	Libri (scadenza)
lorenzoferrari	La coscienza di Zeno (28 giorni)
simoneruggeri	

Salvataggio File con la classe Gson


- La classe Gson appartiene alla libreria JSON di Google, ed è utilizzata per convertire oggetti Java in rappresentazioni JSON (serializzazione) e per convertire stringhe JSON in oggetti Java (deserializzazione). Questo strumento facilita la manipolazione e il trasferimento di dati strutturati all'interno delle applicazioni Java. Per questo motivo ogni classe della struttura biblioteca implementa un metodo toJson() che ha il compito di serializzare l'oggetto della classe.

▼ 📁 lib

📄 gson-2.10.1.jar

```
public String toJson() {  
    Gson gson = new Gson();  
    return gson.toJson(this);  
}
```


Salvataggio File

- Il salvataggio su file JSON viene richiamato nel momento della chiusura della finestra con il pulsante  in alto a destra.

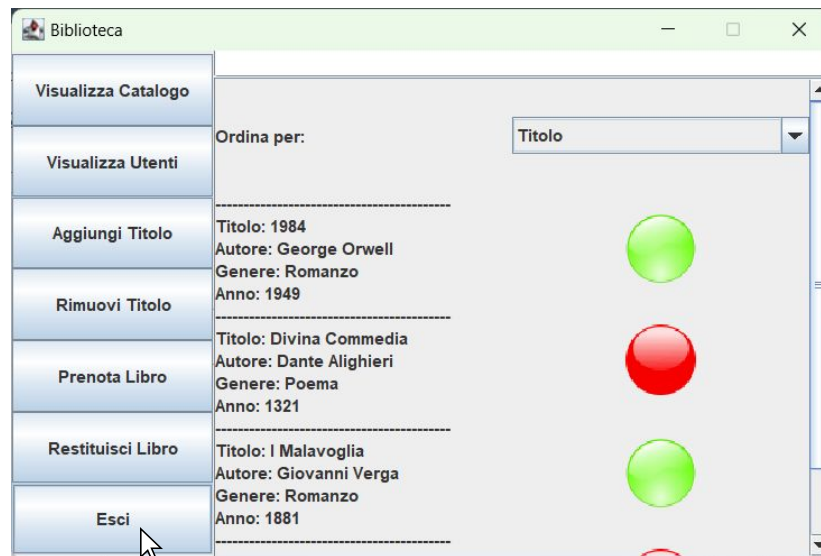
```
addWindowListener(new WindowAdapter() {  
    @Override  
    public void windowClosing(WindowEvent e) {  
        salva();  
    }  
});
```



Salvataggio File

- Il salvataggio su file JSON viene richiamato anche dalla pressione del pulsante “Esci” del pannello BibliotecaMenu.

```
esciButton.addActionListener(new ActionListener() {  
    public void actionPerformed(ActionEvent e) {  
        pannello.salva();  
        System.exit(0);  
    }  
});
```





Problematiche nel salvataggio in Json

- La corrispondenza tra un oggetto Libro nell'ElencoLibri di una Biblioteca e l'oggetto Libro nella Prenotazione di un utente veniva meno in seguito alla serializzazione in stringhe di rappresentazione JSON. A questo scopo la classe Biblioteca implementa il metodo loadFromJson(), che ricostruisce la listaLibri e ne attinge per ricostruire gli oggetti Prenotazione della listaUtenti.
- La classe Gson non riusciva ad accedere alle variabili protected della classe LocalDate, quindi è stata creata una classe Data personalizzata più semplice e finalizzata al progetto.

```
public void loadFromJson(String json) {
    Gson gson = new Gson();
    List<List<String>> bibliotecaJson = gson.fromJson(json, List.class);
    List<String> libriJson = bibliotecaJson.get(0);
    List<String> utentiJson = bibliotecaJson.get(1);

    for (String libroJson : libriJson) {
        Libro libro = gson.fromJson(libroJson, Libro.class);
        listaLibri.aggiungiTitolo(libro);
    }

    for (String utenteJson : utentiJson) {
        Utente utente = gson.fromJson(utenteJson, Utente.class);
        for (Prenotazione prenotazione : utente.getPrenotazioni())
            for (Libro libro : listaLibri.getElenco())
                if (prenotazione.getLibroPrenotato().equals(libro))
                    prenotazione.setLibroPrenotato(libro);
        listaUtenti.add(utente);
    }
}
```

Conclusioni



- Il progetto della biblioteca virtuale offre numerosi vantaggi, tra cui una gestione più efficiente del catalogo dei libri e un monitoraggio accurato dei prestiti.
- Tuttavia, presenta anche alcuni svantaggi, come la necessità di formazione del personale per l'uso del sistema e la scarsa autonomia e accessibilità lasciata agli utenti.
- Nonostante questi potenziali inconvenienti, una gestione centralizzata delle risorse della biblioteca garantisce uniformità, coerenza e qualità dei dati, che sono caratteristiche fondamentali nella gestione di un database come un archivio di libri.



FINE