

## Spiegazione del Diagramma ER

Questo schema rappresenta il database per la gestione di una biblioteca, con tre entità principali: Utenti, Elementi del catalogo (libri e riviste), e Prestiti.

### 1. Utenti

Ogni utente ha un ID univoco, nome, cognome, data di nascita e un numero di tessera (che può essere riassegnato se perso/rinnovato).

Perché?: Separare l'ID interno dal numero di tessera permette maggiore flessibilità nella gestione degli utenti.

### 2. Elementi del catalogo (Libri e Riviste)

elementi\_catalogo contiene le informazioni comuni: ISBN (codice univoco), titolo, anno di pubblicazione e numero di pagine.

libri aggiunge autore e genere.

riviste hanno invece la periodicità (settimanale, mensile, semestrale).

Perché questa struttura?: Invece di duplicare campi come ISBN e titolo in due tabelle separate, ho usato una tabella base (elementi\_catalogo) con due tabelle collegate (libri e riviste).

Così, un prestito può riferirsi a un elemento generico, che poi può essere un libro o una rivista senza complicazioni.

### 3. Prestiti

Ogni prestito registra:

Chi ha preso l'elemento (collegamento a utenti).

Cosa è stato prestato (collegamento a elementi\_catalogo).

Le date inizio, restituzione prevista (30 giorni dopo l'inizio) ed effettiva (se ancora non restituito, sarà NULL).

Perché queste date?:

Con data\_restituzione\_previsa ed effettiva, possiamo facilmente trovare prestiti scaduti (quando l'effettiva è NULL e la prevista è passata).

### 4. Ricerche e funzionalità

Lo schema supporta tutte le operazioni richieste:

Ricerca per ISBN/autore/titolo: Basta cercare in elementi\_catalogo (+ libri se serve l'autore).

Prestiti attivi per utente: Si cercano in prestiti quelli senza data\_restituzione\_effettiva.

Prestiti scaduti: Sono quelli con data\_restituzione\_prevista < oggi e effettiva = NULL.

Possibili miglioramenti

Se in futuro si aggiungono nuovi tipi (es. DVD), basta creare una nuova tabella collegata a elementi\_catalogo.

La periodicità delle riviste sarà un enum nel codice Java (ma nel database può essere una stringa o un intero).

