

PROGETTO BASI DI DATI

Matteo Squarzoni mat.2068240

Francesco Lapenna mat.2072134

1. ABSTRACT

La libreria Book Gateway è un importante centro vendita di libri, vanta dipendenti di alta professionalità, ognuno dei quali è specializzato in un determinato settore della libreria. La libreria infatti è suddivisa in settori, situati nei vari piani della libreria, dove si possono trovare i libri di una determinata categoria, ad esempio è presente una sezione “classici”, “fantasy” e molti altri. I libri vengono scritti da un autore oppure possono nascere anche dalla collaborazione di più autori, inoltre i libri vengono pubblicati da una casa editrice, la quale deve detenere il diritto di pubblicare le opere degli autori. Tutti i clienti della libreria possono dunque acquistare libri, ma nel caso decidano di diventare dei lettori “premium”, tramite l’acquisizione della tessera della libreria, sulla base dei punti accumulati sulla tessera, i clienti “premium” possono riscattare gadget esclusivi (non acquistabili) in omaggio. Inoltre i clienti premium hanno la possibilità di poter lasciare delle recensioni dove possono esprimere un punteggio di apprezzamento (da 1 a 5) per i libri che leggono, queste recensioni vengono conservate all’interno della base di dati della libreria che periodicamente pubblica statistiche di queste recensioni per aumentare le interazioni con i clienti. Nonostante il business principale della libreria sia quello della vendita dei libri, è possibile acquistare anche alcuni cd-dvd di vari film e serie tv situati in concomitanza della sezione “cassa”, tuttavia per questa tipologia di articoli momentaneamente non è possibile rilasciare alcuna recensione trattandosi di un business secondario.

2. ANALISI DEI REQUISITI

2.1 DESCRIZIONE

Si vuole creare una base di dati per la gestione della libreria.

La libreria vende alcuni prodotti tra i quali principalmente libri ma anche cd-dvd.

Un prodotto è descritto da:

- EAN-13, univoco per tipologia di prodotto
- Quantità in magazzino
- Prezzo
- Titolo / nome
- Anno di pubblicazione

In particolare un libro è descritto da:

- Numero pagine
- Copertina rigida / flessibile

In particolare un cd-dvd è descritto da:

- Durata in minuti

Ogni libro può essere scritto da uno o più autori e viene pubblicato da una casa editrice.

Un autore è descritto da:

- Codice Fiscale
- Nome
- Cognome
- Nazionalità
- Sesso

Una casa editrice è caratterizzata da:

- Nome
- P.IVA
- Città
- Email
- Anno di fondazione

All'interno della libreria ci sono varie sezioni che accolgono i libri di un determinato genere, dunque il nome della sezione corrisponde al genere dei libri che contiene. Nella sezione "cassa" si possono trovare alcuni scaffali con i cd-dvd. Ogni sezione si trova in un piano della libreria.

In particolare ogni sezione è descritta da:

- Nome
- Numero piano
- Numero scaffali

I dipendenti possono essere di due tipi: "esperti di libri" o "cassieri". Ogni esperto è specializzato in un determinato genere di libri. Ogni sezione viene gestita da un numero variabile di esperti. I cassieri possono gestire solamente la sezione "cassa".

In particolare un dipendente è descritto da:

- Codice fiscale
- Nome
- Cognome
- Età
- Stipendio
- Sesso

I clienti della libreria possono acquistare uno o più prodotti. Una vendita a un cliente può contenere uno o più prodotti.

La vendita è caratterizzata da:

- Data
- Prezzo totale

Un cliente può decidere di diventare cliente “premium” acquistando la tessera della libreria. Tramite gli acquisti accumulerà dei punti sulla tessera grazie ai quali potrà ricevere dei gadget omaggio una volta raggiunta una certa soglia di punti. Può riscattare un determinato gadget una sola volta.

Un cliente è caratterizzato da:

- Codice Fiscale
- Nome
- Cognome
- Telefono
- Email
- Indirizzo

Una tessera è caratterizzata da:

- Numero tessera
- Punti accumulati

Un gadget è caratterizzato da:

- Nome
- Costo in punti

Solo i clienti premium hanno la possibilità di lasciare recensioni sui libri. In particolare un cliente può lasciare una sola recensione per ogni libro che legge.

Ogni recensione viene descritta da:

- Punteggio (da 1 a 5)
- Commento
- Data

3. PROGETTAZIONE CONCETTUALE

3.1 LISTA ENTITÀ

Il Database è formato dalle seguenti tabelle. Tutti i campi sono NOT NULL tranne quelli specificati.

- **Prodotto**: rappresenta un prodotto vendibile
 - EAN-13: *string*
 - Quantità: *int*
 - Prezzo: *int*
 - Titolo: *string*
 - Anno: *int*
- **Libro**: rappresenta una specializzazione di prodotto
 - Numero pagine: *int*
 - Copertina rigida: *bool*
- **CD-DVD**: rappresenta una specializzazione di prodotto

- Durata: *int*
- **Autore:** rappresenta l'autore di un libro
 - CF: *string*
 - Nome: *string*
 - Cognome: *string*
 - Nazionalità: *string*
 - Sesso: *string*
- **Casa editrice:** rappresenta l'editore che pubblica libri
 - Nome: *string*
 - PIVA: *string*
 - Città: *string*
 - Email: *string*
 - Anno di fondazione: *int*
- **Sezione:** rappresenta una parte della libreria dove si trovano i libri di quel genere oppure la sezione cassa dove si trovano anche i cd-dvd
 - Nome: *string*
 - Numero piano: *int*
 - Numero scaffali: *int*
- **Dipendente:** rappresenta un dipendente della libreria
 - CF: *string*
 - Nome: *string*
 - Cognome: *string*
 - Età: *int*
 - Stipendio: *int*
 - Sesso: *string*
- **Esperto:** rappresenta una tipologia di dipendente specializzato in libri di una certa categoria impiegato presso una delle sezioni della libreria
- **Cassiere:** rappresenta una tipologia di dipendente impiegato nella sezione cassa
- **Vendita:** rappresenta la vendita ad un cliente
 - Data: *date*
 - Prezzo totale: *string*
- **Cliente:** rappresenta un cliente che acquista prodotti
 - CF: *string*
 - Nome: *string*
 - Cognome: *string*
 - Telefono: *string*
 - Email: *string*
 - Indirizzo: *string*
- **Base:** rappresenta una tipologia di cliente privo di tessera e che non può lasciare recensioni
- **Premium:** rappresenta una tipologia di cliente proprietario di tessera e con possibilità di lasciare recensioni
- **Tessera:** rappresenta una tessera di un cliente premium

- Numero: *string*
- Punti: *int*
- **Gadget**: rappresenta un oggetto non acquistabile che è possibile ricevere in omaggio con i punti guadagnati sulla tessera
 - Nome: *string*
 - Costo: *int*
- **Recensione**: rappresenta la recensione di un cliente su un libro
 - Punteggio: *int*
 - Commento: *string*
 - Data: *date*

3.2 LISTA RELAZIONI

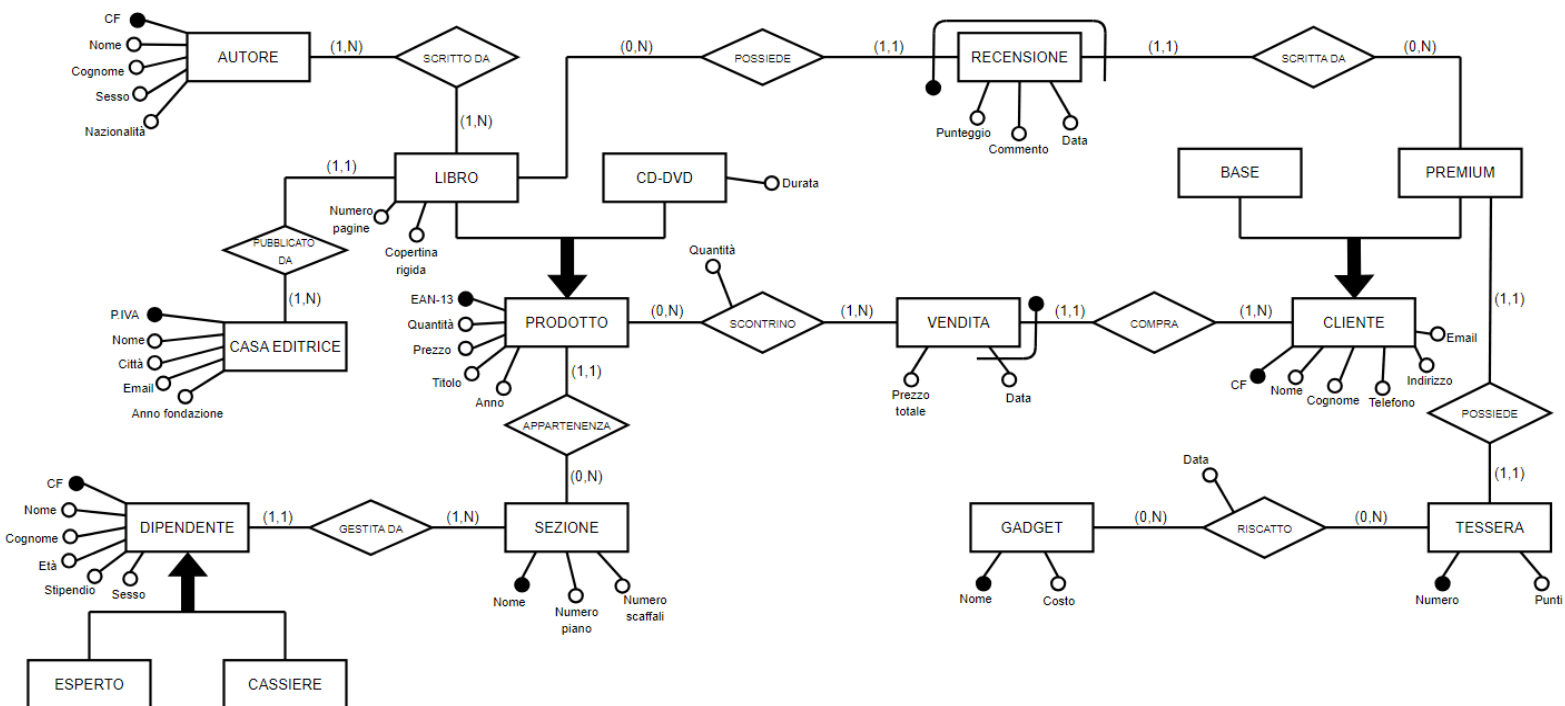
- **Autore-libro**: scrittura (N:M)
 - Un autore può scrivere uno o più libri
 - Un libro può essere scritto da uno o più autori
- **Editore-Libro**: (1:N)
 - Una casa editrice può pubblicare uno o più libri
 - Un libro è pubblicato da una sola casa editrice
- **Libro-Recensione**: (1:N)
 - Un libro può ricevere varie recensioni
 - Una recensione riguarda un solo libro
- **Recensione-Premium**: (1:N)
 - Un cliente premium può scrivere varie recensioni
 - Una recensione viene scritta da un solo cliente
- **Premium-Tessera**: (1:1)
 - Un utente premium ha una sola tessera
 - Ogni tessera è posseduta da un utente premium
- **Tessera-Gadget**: riscatto (N:M)
 - Una tessera può riscattare uno o più gadget
 - Un gadget può essere riscattato da una o più tessere
- **Prodotto-Vendita**: scontrino (N:M)
 - Una vendita può contenere uno o più prodotti
 - Un prodotto può essere venduto in una o più vendite
- **Vendita-Cliente**: (1:N)
 - Un cliente può effettuare una o più vendite
 - Una vendita viene effettuata da un cliente

- **Prodotto-Sezione:** (1:N)
 - Un prodotto può essere contenuto in una sola sezione
 - Una sezione può contenere uno o più prodotti
- **Sezione-Dipendente:** (1:N)
 - In una sezione possono lavorare uno o più dipendenti
 - Ogni dipendente lavora in una sola sezione

3.3 LISTA GENERALIZZAZIONI

- **Cliente** è una generalizzazione totale ed esclusiva di **Base** e **Premium**
- **Dipendente** è una generalizzazione totale ed esclusiva di **Esperto** e **Cassiere**
- **Prodotto** è una generalizzazione totale ed esclusiva di **Libro** e **CD-DVD**

3.4 SCHEMA E-R



Vincoli non esprimibili:

- Lo stesso gadget non può essere riscattato più di una volta dalla stessa tessera.
- Un cassiere può gestire solo la sezione “cassa”.

4. PROGETTAZIONE LOGICA E RISTRUTTURAZIONE

4.1 ANALISI DELLE RIDONDANZE

Analizzando lo schema E-R (visibile al punto 3.4) è possibile notare che l'attributo "Prezzo totale" dell'entità "Vendita" potrebbe essere calcolato dalla relazione "Scontrino". L'attributo "Prezzo totale" di una vendita non è altro che la somma dei prezzi di tutti i prodotti acquistati in quella singola vendita.

È necessario quindi analizzare le operazioni riguardante questo attributo per capire se rimuoverlo oppure no. Si suppone che il costo di una scrittura abbia costo doppio rispetto ad una lettura, mentre un check o un'operazione aritmetica costi come una lettura.

- **Operazione 1 (500 alla settimana):** inserimento di una nuova vendita
- **Operazione 2 (20 alla settimana):** controllo a campione il prezzo totale di vendita

Concetto	Costrutto	Volume
Prodotto	Entità	10'000
Vendita	Entità	1'250'000
Scontrino	Relazione	4'750'000
Cliente	Entità	50'000

Con ridondanza

- **Operazione 1**

Concetto	Costrutto	Accesso	Tipo
Prodotto	Entità	3	Lettura
Vendita	Entità	1	Scrittura
Vendita	Entità	3	Lettura
Scontrino	Relazione	3	Scrittura
Cliente	Entità	1	Lettura

Costo: 3500 accessi in lettura, 2000 accessi in scrittura + costo check (per verificare che i prezzi corrispondano tra il “prezzo totale” e il prezzo che mi potrei calcolare dalla relazione scontrino)

- **Operazione 2**

Concetto	Costrutto	Accesso	Tipo
Vendita	Entità	1	Lettura

Costo: 20 accessi in lettura

Costo settimanale totale: $3500 + 2000 \times 2 + 20 + \text{costo check (500)} + \text{costo operazione somma check (1500)} = 7520 + 500 + 1500 = \mathbf{9520}$

Senza ridondanza

- **Operazione 1**

Concetto	Costrutto	Accesso	Tipo
Prodotto	Entità	3	Lettura
Vendita	Entità	1	Scrittura
Vendita	Entità	3	Lettura
Scontrino	Relazione	3	Scrittura
Cliente	Entità	1	Lettura

Costo: 3500 accessi in lettura, 2000 accessi in scrittura

- **Operazione 2**

Concetto	Costrutto	Accesso	Tipo
Scontrino	Entità	3	Lettura
Vendita	Entità	1	Lettura
Prodotto	Entità	3	Lettura

Costo: 140 accessi in lettura + 60 (costo somma)

Costo settimanale totale: $3500 + 2000 \times 2 + (140+60) = 7700$

In questo caso conviene eliminare “Prezzo Totale” e calcolarlo quando viene richiesto.

4.2 ELIMINAZIONI DELLE GENERALIZZAZIONI

- **Dipendente:** è una generalizzazione totale ed esclusiva, legata all’entità Sezione. Dato che entrambe le classi figlie non hanno né relazioni né attributi, allora la generalizzazione è risolvibile banalmente andando ad accorpare le due classi figlie all’interno della classe padre, inserendo dunque un nuovo attributo “Incarico” che va ad indicare il tipo di dipendente (esperto o cassiere).
- **Prodotto:** è una generalizzazione totale ed esclusiva, legata all’entità Vendita. La classe figlia “Libro” ha alcuni attributi ed è in relazione molti a molti con “Autore” e uno a molti con “Casa Editrice”. L’altra classe figlia “CD-DVD” non presenta relazioni ma ha un proprio attributo. Dunque si è deciso di eliminare la generalizzazione sostituendola con la creazione di due nuove relazioni uno a uno (una tra “Prodotto” e “Libro” e l’altra tra “Prodotto” e “CD-DVD”). In questo modo si favorisce tra l’altro una minimizzazione dei campi NULL dato che di fatto vengono mantenute le entità “Libro” e “CD-DVD”.
- **Cliente:** è una generalizzazione totale ed esclusiva, legata all’entità Vendita. La classe figlia “Premium” è in relazione uno a molti con “Recensione” ed uno a uno con “Tessera”. L’altra classe figlia “Base” non presenta né relazioni né attributi. Dunque si è deciso di eliminare la generalizzazione accorpendo le due classi figlie in quella padre. Si sarebbe potuto aggiungere un flag booleano per indicare se il cliente è premium o meno, ma dato che in “Cliente” è presente una chiave esterna per il numero della tessera, se il numero è presente significa che il cliente è premium se invece il valore di questa chiave esterna è NULL allora il cliente è un cliente base. Si può dunque risparmiare il flag booleano che indica se il cliente è premium o meno.

4.3 PARTIZIONAMENTO/ACCORPAMENTO DI ENTITÀ E RELATIONSHIP

L'entità "Tessera" contiene il suo identificatore (numero della tessera) ed i punti accumulati da un cliente. Dunque dato che l'entità "Cliente" ha come chiave esterna il numero della tessera è possibile accorpare il rimanente campo di tessera, ossia i punti accumulati, all'interno di "Cliente" per risparmiare una tabella. Qualora un cliente sia un cliente base, dunque privo di tessera, allora il numero della tessera è settato a NULL e lo saranno anche i punti accumulati.

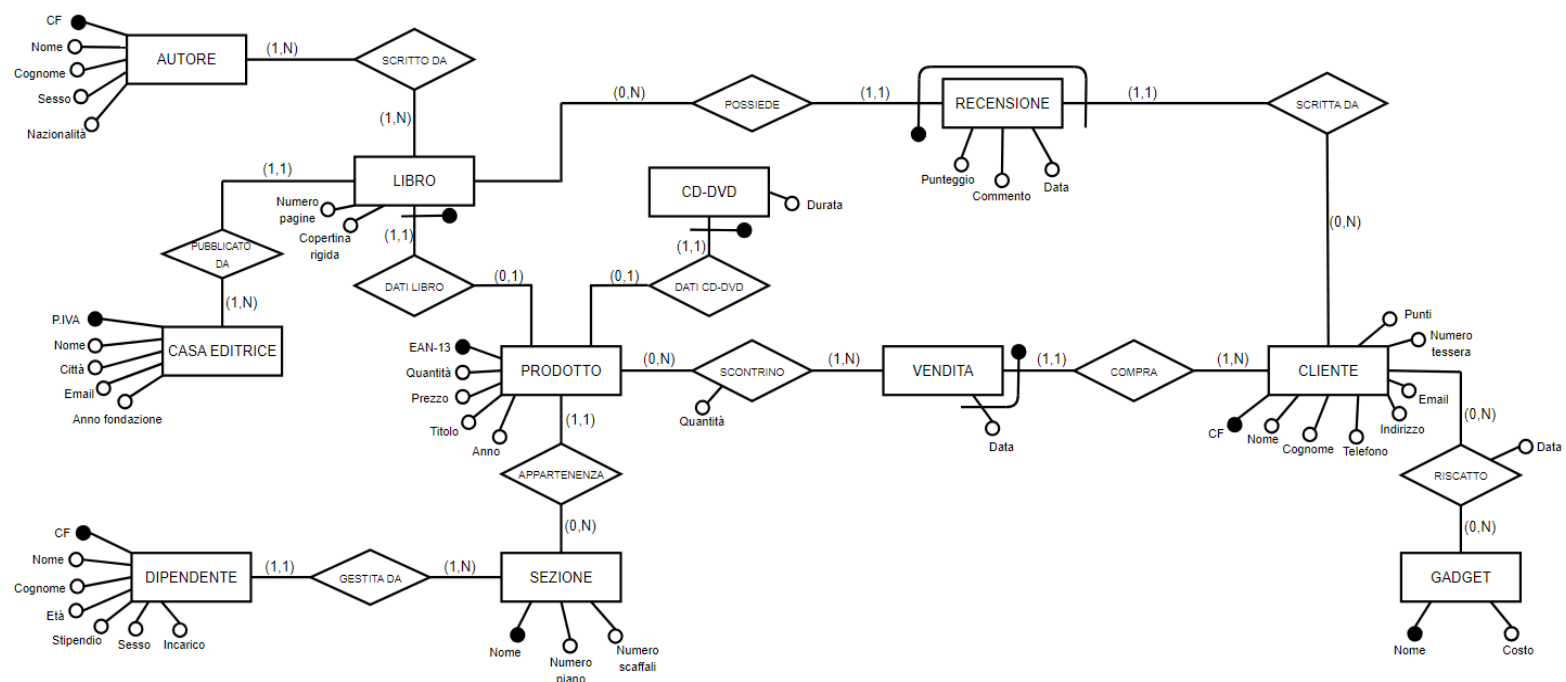
4.4 SCELTA DI IDENTIFICATORI PRIMARI

L'entità "Vendita" avrà come identificatore primario la composizione tra la chiave esterna che punta al codice fiscale di un cliente e l'attributo data (in formato esteso) dato che un cliente non potrà mai concretizzare due vendite nello stesso identico momento.

L'entità "Recensione" avrà come identificatore primario la composizione delle due chiavi esterne che fanno riferimento al cliente e al libro.

Le entità "Libro" e "CD-DVD" avranno come identificatore primario la chiave esterna che si riferisce al codice EAN13 del prodotto.

4.5 DIAGRAMMA ER RISTRUTTURATO



4.6 DESCRIZIONE SCHEMA RELAZIONALE

Sezione(Nome, NumeroPiano, NumeroScaffali)

Dipendente(CF, Nome, Cognome, Età, Stipendio, Sesso, Incarico, Sezione)

Prodotto(EAN13, Quantità, Prezzo, Titolo, Anno, Sezione)

Autore(CF, Nome, Cognome, Sesso, Nazionalità)

CasaEditrice(PIVA, Nome, Città, Email, AnnoFondazione)

Libro(EAN13, NumeroPagine, CopertinaRigida, CasaEditrice)

Scrittura(Autore, Libro)

Cd-Dvd(EAN13, Durata)

Gadget(Nome, Costo)

Cliente(CF, Nome, Cognome, Telefono, Indirizzo, Email, NumeroTessera, Punti)

Riscatto(Cliente, Gadget, Data)

Recensione(Cliente, Libro, Punteggio, Commento, Data)

Vendita(Cliente, Data)

Scontrino(Cliente, Data, Prodotto, Quantità)

4.7 VINCOLI DI INTEGRITÀ REFERENZIALE

Dipendente.Sezione -> Sezione.Nome

Prodotto.Sezione -> Sezione.Nome

Libro.CasaEditrice -> CasaEditrice.PIVA

Libro.EAN13 -> Prodotto.EAN13

Scrittura.Autore -> Autore.CF

Scrittura.Libro -> Libro.EAN13

Cd-Dvd.EAN13 -> Prodotto.EAN13

Riscatto.Cliente -> Cliente.CF

Riscatto.Gadget -> Gadget.Nome

Recensione.Cliente -> Cliente.CF

Recensione.Libro -> Libro.EAN13

Vendita.Cliente -> Cliente.CF

Scontrino.(Cliente, Data) -> Vendita.(Cliente, Data)

Scontrino.Prodotto -> Prodotto.EAN13

5. QUERY E INDICI

5.1 QUERY

- Query 1: Numero libri di autori [nazionalità] per ogni piano della libreria (in questo caso nazionalità = 'British')

```
1 SELECT sezione.numeropiano, COUNT(DISTINCT scrittura.libro) AS numLibri
2 FROM prodotto, scrittura, autore, sezione
3 WHERE prodotto.ean13 = scrittura.libro AND scrittura.autore = autore.cf
4     AND autore.nazionalita = 'British' AND prodotto.sezione = sezione.nome
5 GROUP BY sezione.numeropiano
6 ORDER BY sezione.numeropiano;
```

Data Output Messages Notifications

	numeropiano integer	numlibri bigint
1	0	2
2	2	20
3	3	91
4	4	27

- Query 2: Autori con almeno [tot] recensioni di punteggio massimo (in questo caso tot = '4')

```
1 SELECT nome, cognome FROM autore
2 WHERE cf IN
3     (SELECT autore.cf
4      FROM recensione, scrittura, autore
5      WHERE recensione.libro = scrittura.libro
6          AND scrittura.autore = autore.cf
7          AND recensione.punteggio = 5
8      GROUP BY (autore.cf)
9      HAVING COUNT(*) >= 4);
```

Data Output Messages Notifications

	nome character varying (25)	cognome character varying (25)
1	J.R.R.	Tolkien
2	Veronica	Roth
3	Rick	Riordan
4	Marie	Lu
5	Tahereh	Mafi
6	Holly	Black
7	Cassandra	Clare
8	Brandon	Sanderson

- **Query 3: I 5 clienti che hanno acquistato il maggior numero di libri dalla sezione [nome] (in questo caso nome = 'Horror')**

```

1 SELECT cliente.cf, cliente.nome, cliente.cognome, SUM(scontrino.quantita) AS numLibri
2 FROM cliente, scontrino, prodotto
3 WHERE cliente.cf = scontrino.cliente
4       AND scontrino.prodotto = prodotto.ean13
5       AND prodotto.sezione = 'Horror'
6 GROUP BY cliente.cf
7 ORDER BY numLibri DESC
8 LIMIT 5;

```

Data Output Messages Notifications

	cf [PK] character (16)	nome character varying (25)	cognome character varying (25)	numlibri bigint
1	ADIROB9151248103	Billie	Britten	4
2	TJFMHL9028834804	Lonnard	Dressell	4
3	QDRSAD0147962426	Olva	Trowsdale	2
4	RSXXEZ2572660446	Faye	Marks	2
5	WYVUBN4704788198	Rodger	Lemoir	2

- **Query 4: Casa editrice con sede a [città] che ha venduto il maggior numero di libri (in questo caso città = 'Milano')**

```

1 DROP VIEW IF EXISTS numLibriVendutiCaseEditriciPerCitta;
2 CREATE VIEW numLibriVendutiCaseEditriciPerCitta AS
3     SELECT casaeditrice.nome, casaeditrice.citta, SUM(scontrino.quantita) AS numLibri
4     FROM scontrino, libro, casaeditrice
5     WHERE scontrino.prodotto = libro.ean13
6           AND libro.casaeditrice = casaeditrice.piva
7     GROUP BY casaeditrice.nome, casaeditrice.citta;
8
9 SELECT * FROM numLibriVendutiCaseEditriciPerCitta
10 WHERE numLibri = (SELECT MAX(numLibri)
11                  FROM numLibriVendutiCaseEditriciPerCitta
12                  WHERE numLibriVendutiCaseEditriciPerCitta.citta = 'Milano')
13 AND numLibriVendutiCaseEditriciPerCitta.citta = 'Milano';

```

Data Output Messages Notifications

	nome character varying (25)	citta character varying (25)	numlibri bigint
1	Piemme	Milano	218

- **Query 5: Titolo del libro che ha incassato di più per la casa editrice [nome] (in questo caso nome = 'Mondadori')**

```

1 DROP VIEW IF EXISTS IncassiPerLibro_CasaEditrice;
2 CREATE VIEW IncassiPerLibro_CasaEditrice AS
3     SELECT libro.ean13, casaeditrice.nome AS casaeditrice, SUM(scontrino.quantita*prodotto.prezzo) AS incasso
4     FROM prodotto, libro, scontrino, casaeditrice
5     WHERE libro.ean13 = scontrino.prodotto
6           AND prodotto.ean13 = libro.ean13
7           AND libro.casaeditrice = casaeditrice.piva
8     GROUP BY libro.ean13, casaeditrice.nome;
9
10
11 SELECT prodotto.titolo, IncassiPerLibro_CasaEditrice.incasso
12 FROM prodotto, IncassiPerLibro_CasaEditrice
13 WHERE prodotto.ean13 = IncassiPerLibro_CasaEditrice.ean13
14     AND IncassiPerLibro_CasaEditrice.casaeditrice = 'Mondadori'
15     AND IncassiPerLibro_CasaEditrice.incasso = (SELECT MAX(incasso)
16                                                  FROM IncassiPerLibro_CasaEditrice
17                                                  WHERE IncassiPerLibro_CasaEditrice.casaeditrice = 'Mondadori');

```

Data Output
Messages
Notifications

	titolo character varying (60)	Incasso bigint
1	Oathbringer	981

Il codice di queste query è fornito all'interno del file query.sql

5.2 INDICI

```
CREATE INDEX idx_PunteggioLibro ON recensione (punteggio, libro);
```

L'indice idx_PunteggioLibro viene definito sulla tabella Recensione per velocizzare le operazioni della query 2, l'indicizzazione viene fatta prima per il punteggio e poi per libro, così facendo è possibile considerare solamente la parte di tabella dove le recensioni hanno punteggio massimo (punteggio = 5) riducendo così sensibilmente la porzione di tabella da valutare.

Non è possibile notare differenze in termini di tempi di esecuzione dato che le tabelle dovrebbero avere dimensioni molto più grandi rispetto a quelle del database in questione.

6. CODICE C

6.1 COMPILAZIONE

Per la compilazione del codice c immettere da terminale il comando:

- `gcc queryDB.c -o queryDB -I /usr/include/postgresql -lpq`

Per l'esecuzione del codice immettere successivamente:

- `./queryDB`

6.2 GUIDA ALL'UTILIZZO

All'interno del file c viene aperta una connessione con il database creato precedentemente (per creare il database eseguire su PgAdmin il file `init_tabelle.sql`). Successivamente è possibile digitare un numero da 1 a 5 per selezionare la query che si vuole eseguire, le query presenti sono tutte parametriche quindi è necessario inserire un parametro coerente con i dati presenti all'interno delle tabelle del database (vengono proposti al momento dell'inserimento del parametro alcuni esempi ammissibili). Una volta eseguita la query e visualizzati i risultati è possibile eseguire una nuova query a seconda della scelta che l'utente digita. Nel caso di risultati inconsistenti per la query selezionata, controllare il parametro inserito oppure provare ad eseguire nuovamente il programma se ha dato qualche problema con la creazione di qualche vista.