

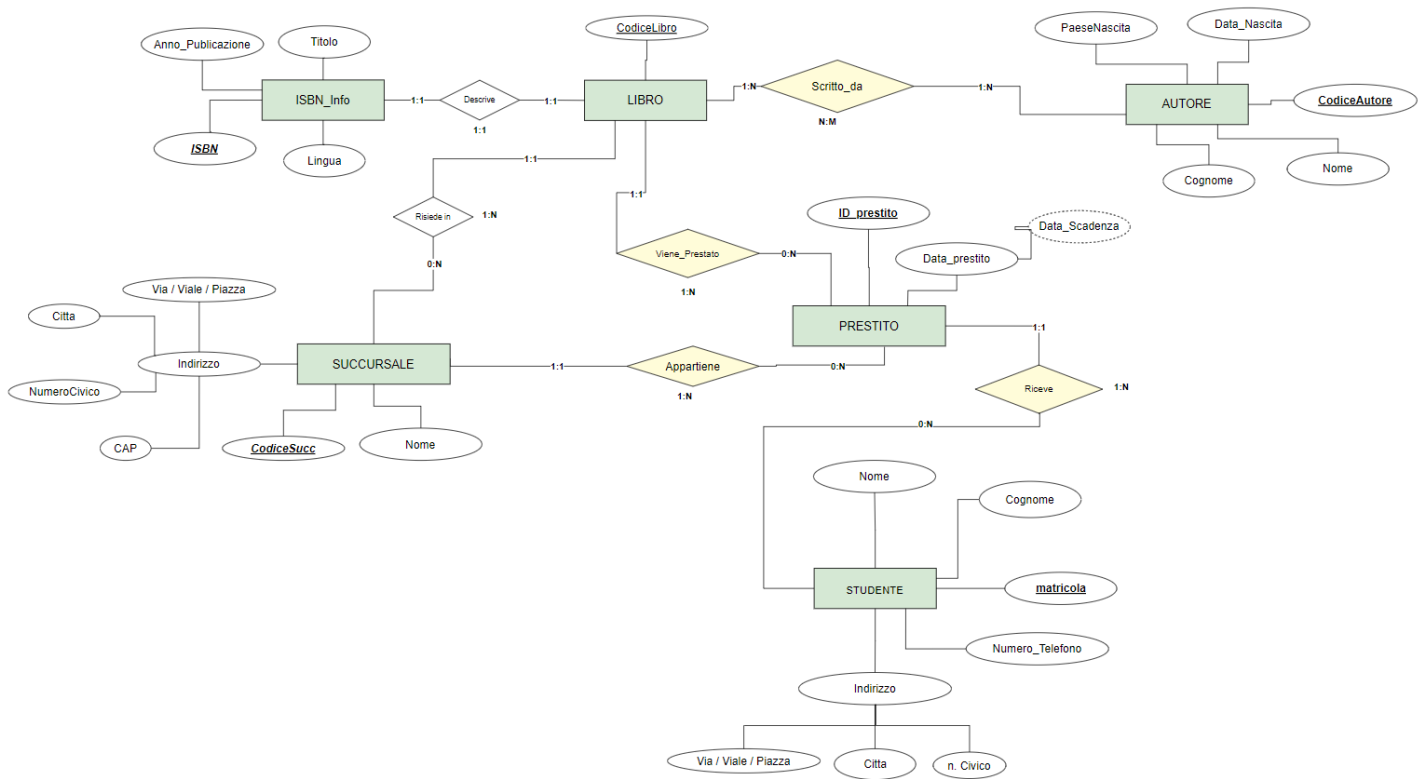
Relazione Progetto Basi dati

Edoardo Ponsanesi 166205
Enrico Albertini y636728

1. Definizione del problema

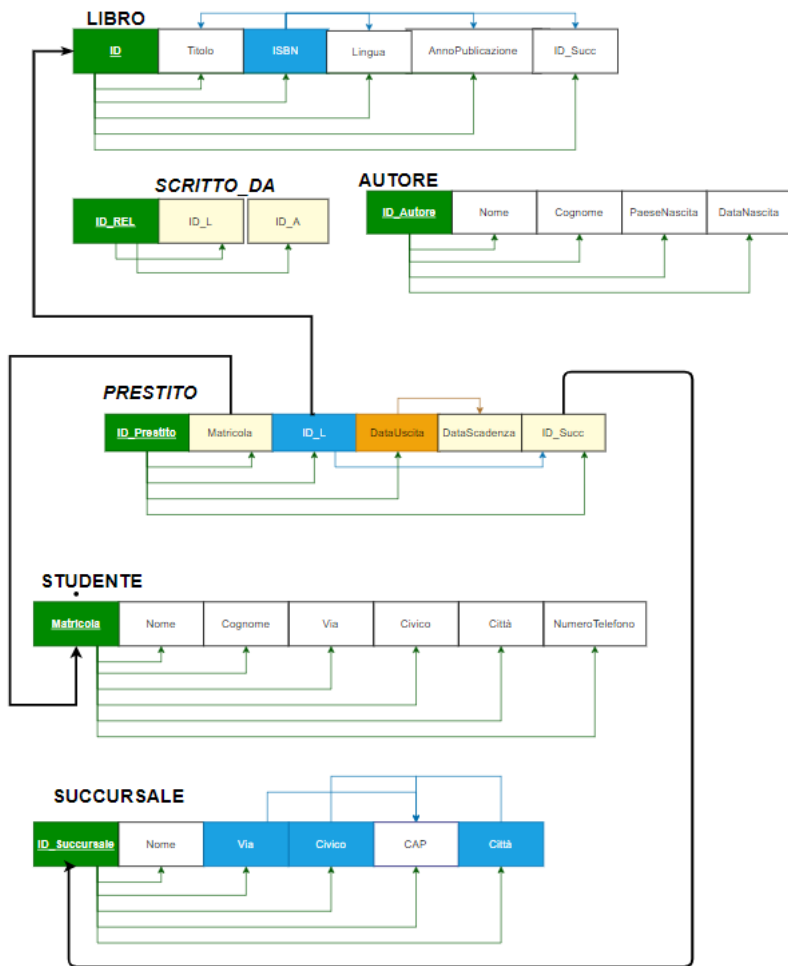
Immagine Modello ER
Immagine schema Pre-Normalizzazione
I

2. Modello ER



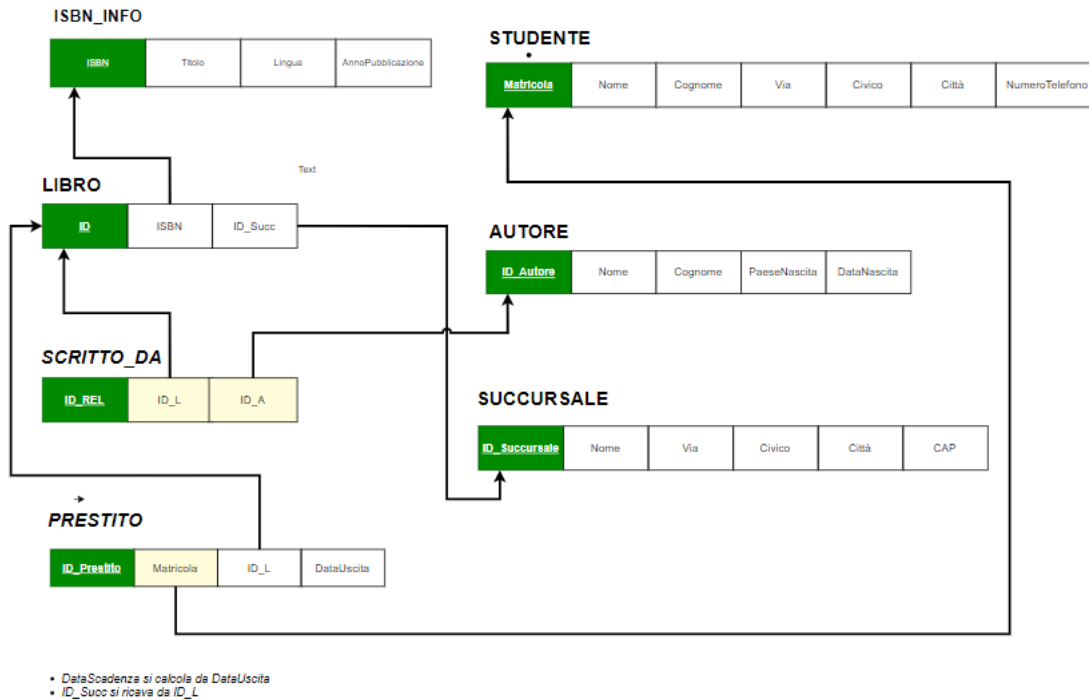
3. Modello relazionale in terza forma normale

Schema PRENORMALIZZAZIONE



Schema 3NF

```
FROM ISBN_Info AS info, Libro as l, Succursale as succ
WHERE info.TITOLO LIKE '%$nome_libro%' AND succ.ID_SUCC = l.ID_S;
```



4. Interrogazioni delle tracce in SQL con l' equivalente espressione scritta in Algebra Relazionale

[1]

Ricerca di un libro inserendo il titolo (anche parziale) - nel caso in cui nessun parametro venga specificato deve essere presentata la lista completa dei libri comprese le informazioni sintetiche del libro: titolo, isbn, in che succursale sono, ecc... (sintetiche - nome, cognome) sull'autore.

```
SELECT i.ISBN, i.TITOLO, i.LINGUA, s.NOME
FROM ISBN_Info AS i, Libro as l, Succursale as s
WHERE i.ISBN = l.ISBN AND i.TITOLO LIKE '%$nome_libro%'
AND l.ID_S = s.ID_SUCC
```

$$\rho_i(\text{ISBN_Info}), \rho_l(\text{Libro}), \rho_s(\text{Succursale})$$

$$\pi_{(\text{ISBN}, \text{TITOLO}, \text{LINGUA}, \text{NOME})}(\text{ISBN_Info} \bowtie \langle i.\text{ISBN} = l.\text{ISBN} \wedge i.\text{TITOLO} = \text{nome_libro} \rangle \text{Libro} \bowtie \langle l.\text{ID_S} = s.\text{ID_SUCC} \rangle \text{Succursale})$$

[2] Visualizzazione di tutti i libri di un determinato autore, eventualmente suddivisi per anno di pubblicazione.

```
SELECT i.TITOLO, i.ANNO_PUBBLICAZIONE, i.LINGUA, l.ISBN
FROM Libro AS l, ISBN_Info AS i
WHERE l.ISBN = i.ISBN
AND l.ID_LIBRO IN ( SELECT ID_L FROM Scritto_Da WHERE ID_A = $id_autore)
ORDER BY ANNO_PUBBLICAZIONE";
```

$$\rho_l(\text{Libro}), \rho_l(\text{Libro})$$

$$\text{LIBRI_AUTORE} \leftarrow \pi_{\langle \text{ID_L} \rangle}(\sigma_{\langle \text{ID_A} = \text{id_autore} \rangle}(\text{Scritto_Da}))$$

$$\text{INFO_LIBRI} \leftarrow \text{Libro} \bowtie \langle l.\text{ISBN} = i.\text{ISBN} \rangle \text{ISBN_Info} \bowtie \langle \text{ID_L} = \text{ID_LIBRO} \rangle \text{LIBRI_AUTORE}$$

$$\text{OUT} \leftarrow \pi_{\langle \text{TITOLO}, \text{ANNO_PUBBLICAZIONE}, \text{LINGUA}, \text{ISBN} \rangle}(\text{INFO_LIBRI})$$

[3]

Ricerca degli autori inserendo uno o più parametri (anche parziali), in forma libera o eventualmente guidata (per esempio menù a tendina con i soli valori

possibili).

```
SELECT NOME, COGNOME, ID_AUTORE, DATA_NASCITA, PAESE_NASCITA
FROM Autore
WHERE NOME LIKE '$nome_a%' AND COGNOME LIKE '$cognome_a%'
AND PAESE_NASCITA = $paese
```

$$AUTORI_RICHIESTI \leftarrow \sigma_{< NOME=nome_a \wedge COGNOME=cognome_a \wedge PAESE_NASCITA=paese >}(Autore)$$
$$OUT \leftarrow \pi_{< NOME, COGNOME, ID_AUTORE, DATA_NASCITA, PAESE_NASCITA >}(AUTORI_RICHIESTI)$$

[4]

Consultare l'elenco degli utenti della biblioteca (con le informazioni principali).

```
SELECT NOME, COGNOME, MATRICOLA, NUMERO_TELEFONO
FROM Studente
```

$$\pi_{< NOME, COGNOME, MATRICOLA, NUMERO_TELEFONO >}(Studente)$$

[5]

Ricerca di un utente della biblioteca e il suo storico dei prestiti (compresi quelli in corso).

```
SELECT p.ID_PRESTITO, p.DATA_USCITA, s.COGNOME, l.ISBN, l.ID_LIBRO
FROM Prestito AS p, Studente AS s, Libro AS l
WHERE MATRICOLA = '$matricola' AND p.MATRICOLA_S = s.MATRICOLA
AND p.ID_L = l.ID_LIBRO
```

$$JOIN_PSL \leftarrow (Prestito) \bowtie_{p.MATRICOLA=s.MATRICOLA \wedge s.MATRICOLA=matricola} (Studente) \bowtie_{p.ID_L=l.ID_LIBRO} (Libro)$$
$$OUT \leftarrow \pi_{< p.ID_PRESTITO, p.DATA_USCITA, s.COGNOME, l.ISBN, l.ID_LIBRO >}(JOIN_PSL)$$

[6]

Consultare lo storico dei prestiti comprese le informazioni (sintetiche - nome, cognome) sull'utente.

```
SELECT p.ID_PRESTITO, p.MATRICOLA_S, p.DATA_USCITA, s.NOME, s.COGNOME
FROM Prestito AS p, Studente AS s
WHERE p.MATRICOLA_S = s.MATRICOLA
```

$$PRESTITI_UTENTE \leftarrow (Prestito) \bowtie_{< p.MATRICOLA=s.MATRICOLA >} (Studente)$$
$$OUT \leftarrow \pi_{< p.ID_PRESTITO, p.MATRICOLA_S, p.DATA_USCITA, s.NOME, s.COGNOME >}(PRESTITI_UTENTE)$$

[7]

Ricerca dei prestiti effettuati in un range di date – nel caso in cui non vengano inserite date deve mostrare i prossimi in scadenza (quelli che scadranno in futuro) comprese le informazioni sintetiche sull'autore.

```
SELECT ID_PRESTITO, MATRICOLA_S, DATA_USCITA FROM Prestito
WHERE DATA_USCITA >= '$data_inizio'
AND DATA_USCITA <= '$data_fine'
```

$$IN_RANGE \leftarrow \sigma_{< DATA_USCITA \geq data_inizio \wedge DATA_USCITA \leq data_fine >}(Prestito)$$
$$OUT \leftarrow \pi_{< ID_PRESTITO, MATRICOLA_S, DATA_USCITA >}(IN_RANGE)$$

[8]

Statistiche (qui abbiamo deciso di utilizzare le join (per completezza))

[8.a] Numero di libri pubblicati in un determinato anno.

```
SELECT i.ANNO_PUBBLICAZIONE AS anno, COUNT(i.ISBN) AS numero_libri
FROM ISBN_Info AS i
```

```
GROUP BY i.ANNO_PUBBLICAZIONE
ORDER BY i.ANNO_PUBBLICAZIONE
```

$OUT \leftarrow ANNO_PUBBLICAZIONE \mathcal{F} ANNO_PUBBLICAZIONE, COUNT_{ISBN}(\pi_{<ANNO_PUBBLICAZIONE, ISBN>}(ISBN_Info))$
oppure (penso sia la stessa cosa)
 $OUT \leftarrow ANNO_PUBBLICAZIONE \mathcal{F} ANNO_PUBBLICAZIONE, COUNT_{ISBN}(ISBN_INFO)$

[8.b] Numero di prestiti effettuati in una determinata succursale.

```
SELECT s.NOME AS nome_succ, COUNT(p.ID_PRESTITO) AS numero_prestiti
FROM Succursale s
LEFT JOIN Libro l ON s.ID_SUCC = l.ID_S
LEFT JOIN Prestito p ON l.ID_LIBRO = p.ID_L
GROUP BY s.ID_SUCC
```

$JOIN_LSP \leftarrow (Succursale) \bowtie_{LEFT< s.ID_SUCC=l.ID_S >} (Libro) \bowtie_{LEFT< l.ID_LIBRO=p.ID_L >} (Prestito)$
 $OUT \leftarrow s.ID_SUCC \mathcal{F} s.NOME, COUNT_{p.ID_PRESTITO} (JOIN_LSP)$

La **prima LEFT JOIN** Questa parte esegue una LEFT JOIN tra Succursale e Libro (sulle colonne ID_S e ID_SUCC rispettivamente). Questa unione combina i dati delle succursali con i libri corrispondenti nella tabella "Libro".

- Se non ci sono libri corrispondenti, la succursale sarà comunque inclusa nel risultato.

La **seconda LEFT JOIN** Combina i dati di Libro con le istanze di Prestito corrispondente.

- Se non ci sono prestiti corrispondenti per un libro, il libro sarà comunque incluso nel risultato.

Infinie la **GROUP BY** raggruppa il risultato per NOME (della succursale) e ID_SUCC.

Possiamo quindi usare **COUNT** per calcolare il numero di prestiti per ogni succursale.

[8.c] Numero di libri pubblicati per autore.

```
SELECT a.ID_AUTORE, a.NOME AS nome, a.COGNOME AS cognome, COUNT(L.ID_LIBRO) AS numero_libri
FROM Autore a
JOIN Scritto_Da sd ON a.ID_AUTORE = sd.ID_A
JOIN Libro l ON sd.ID_L = l.ID_LIBRO
GROUP BY a.ID_AUTORE
```

$JOIN_ASL \leftarrow (Autore) \bowtie_{LEFT< A.ID_AUTORE=SD.ID_A >} (Scritto_Da) \bowtie_{LEFT< l.ID_L=p.ID_LIBRO >} (Libro)$
 $OUT \leftarrow ID_AUTORE \mathcal{F} ID_AUTORE, NOME, COGNOME, COUNT_{ID_LIBRO}(JOIN_ASL)$

La **prima LEFT JOIN** collega la tabella "Autore" con la tabella "Scritto_Da" utilizzando la condizione **A.ID_AUTORE = SD.ID_A** . Questo collegamento ci consente di associare gli autori ai loro scritti.

La **seconda "LEFT JOIN"** (tra ScrittoDa e Libro) ci consente di associare ogni libro con i corrispettivi scrittori.

La clausola **GROUP BY** raggruppa i risultati in base all'ID dell'autore, al nome e al cognome dell'autore.

Posso ora calcolare il conteggio dei libri scritti da ciascun autore tramite la funzione di aggregazione **COUNT**.

- Utilizziamo la **left join** invece che la **join naturale** in modo tale che **tutti** gli autori vengano inclusi nella query(*anche se non hanno scritto alcun libro*).
- Gli autori che non hanno scritto alcun libro compariranno ugualmente nel risultato (*con numero libri = 0*).