

Introduzione al linguaggio SQL

Structured Query Language (Linguaggio Strutturato di Interrogazione di Database)

prof. Cleto Azzani

IPSIA MORETTO BRESCIA

(2004)

Che cos'è un *Data Base*

(Base di dati)

- Insieme di Dati Strutturato in
 - Tabelle,
 - Record
 - Campi,
 - Relazioni
- Gestito da un Motore (Data Base Engine)

Alle Origini dei DataBase

- **CSV** (comma separated values) si tratta di file di testo nel quale i campi sono fra di loro separati da un carattere separatore : virgola, punto-virgola, TAB; i record vengono inseriti senza un ordine prestabilito.
- **ISAM** (Indexed Sequential Access Method), I record vengono inseriti in modo ordinato in fase di scrittura così da accelerare il processo di lettura; si introducono archivi sussidiari denominati “indici”
- **B-Trieve, DBIII, DBIV, Clipper, Paradox** appartengono alla categoria dei “Database non relazionali”.

Data Base Engine

- Elenco di alcuni fra i più affermati motori:
 - Microsoft Access
 - Borland BDE (Tabelle DBIII-Paradox)
 - Interbase (SQL) (multiplatforma Windows e Unix)
 - Microsoft SQL Server
 - MySQL (multiplatforma Windows e Unix)
 - Oracle SQL
 - Transact - SQL
 - PostgreSQL
 - SQLInformix
 - DBII SQL

Problematiche più ricorrenti

- **Ridondanza dei dati** (..spesso inserisco Cognome, Nome ed Indirizzo del medesimo soggetto in applicativi diversi che fanno riferimento a Basi Dati non relazionate fra loro; corro il rischio di inserire dati non uniformi..)
- **Uniformità dei dati** (..una volta inserisco IPSIA Moretto, un'altra volta inserisco I.P.S.I.A. Moretto, oppure IPIA "Moretto" ho fatto uso di dati non uniformi)
- **Indipendenza dalla piattaforma** (..spesso su sistemi diversi la stessa operazione si fa in modo notevolmente diverso e ciò crea problemi..)
- **Sicurezza delle transazioni** (..sto inserendo nel database tutte le fatture dell'ultimo semestre; va via la corrente: non so se l'ultima me l'ha presa o no! cosa faccio?, devo ricostruire l'intera tabella delle fatture per essere certo che non ci siano valori doppi o inseriti a metà??" ..)
- **Gestione corretta ambiente multiutente** (..Mi spieghi perché io sono certo di avere inserito l'ordine per le ultime dieci autovetture a nome del mio miglior cliente e quelle sono finite invece al peggior cliente di un'altro commerciale? eppure SO di averle fermate al magazzino io per primo..)

Il linguaggio SQL

(Structured Query Language)

- L'SQL è un "linguaggio" che permette di effettuare interrogazioni su un RDBMS (Relational DataBase Management System). Lo standard di riferimento prende il nome di ANSI SQL (SQL standardizzato della American National Standard Institute). Sebbene i produttori software affermino che il loro linguaggio è "ANSI Compliant" (compatibile con l'ANSI SQL) nella realtà i vari linguaggi presentano differenze fra di loro.

Esempio: Archivio Libri di una Biblioteca

- **Codice** (Codice Progressivo Numerico)
- **Autore** (Stringa di caratteri max. 30 caratteri)
- **Titolo** (Stringa di caratteri max. 50 caratteri)
- **Editore** (Stringa di caratteri max. 30 caratteri)
- **ISBN** (Codice numerico a 10 cifre)
- **Volume** (Indicazione Numerica Volume)

Creazione di una tabella

- CREATE TABLE "Libri"
(Codice INTEGER NOT NULL, Autore CHAR(30) NOT NULL, Titolo CHAR(50) NOT NULL, Editore CHAR(30) NULL, Isbn CHAR(10) NULL, Volume INTEGER NULL, PRIMARY KEY (Codice));
 - Char(x) - caratteri alfanumerici, x specifica il numero massimo di caratteri permessi nella campo (stringa di caratteri di lunghezza x).
 - Integer – dati interi, positivi o negativi, il cui valore massimo dipende dallo specifico DBMS.
 - Numeric(x, y) - numeri decimali, x esprime il numero massimo di cifre permesse, y il numero di cifre dopo la virgola.(Es.: Decimal(4,2) ammette come max il valore 99,99).
 - Date – formato data (tipo di dato specifico di ogni DBMS).
 - Logical – campo che può contenere solo uno valori: TRUE (vero) o FALSE (falso).
 - Primary Key – crea un indice primario per la nuova tabella;
- NB.: La clausola NOT NULL impone che il relativo campo non possa mai venire lasciato vuoto (deve sempre essere presente un valore); nel caso in cui si voglia invece permettere l' inserimento di record con quel determinato campo vuoto si usa la clausola NULL

Modifica della struttura di una tabella

(Aggiunge o Rimuove Campi della Tabella)

```
ALTER TABLE "Libri" ADD (Prezzo NUMERIC (5,2) NOT NULL);
```

- Aggiunge alla Tabella Libri il campo Prezzo di tipo Numeric(5,2);

```
ALTER TABLE "Libri" DROP Volume;
```

- Rimuove dalla Tabella Libri il campo Volume;

Inserimento dati in una tabella

(Tabella ristrutturata dopo le due istruzioni ALTER TABLE)

- INSERT INTO "Libri" VALUES (21, "Giustozzi-Monti-Zimuel", "Segreti spie codici cifrati", "Apogeo", "8873034837",16.53);
- INSERT INTO "Libri" (Codice, Autore,Titolo,Editore,Prezzo) VALUES (32, "Rubino-Zaccaria", "Manuale Z80", "Il Rostro", 4.53);

Query di Selezione Dati 1

- `SELECT * FROM "Libri";`
Seleziona tutti i campi presenti nella tabella Libri;
- `SELECT Autore,Titolo,Editore FROM "Libri";`
Seleziona solamente i campi elencati;
- `SELECT * FROM "Libri" WHERE (Editore = "Apogeo");`
Seleziona tutti i campi presenti nella tabella Libri dove l'editore è Apogeo;
- `SELECT Autore,Titolo,Editore FROM "Libri" WHERE (Prezzo > 10.0) ;`
Seleziona solamente i campi elencati dalla tabella Libri dove il prezzo supera i 10 €;

Query di Selezione Dati 2

La Clausola LIKE ed il carattere jolly %

- `SELECT * FROM "Libri" WHERE Editore LIKE "A%";`

Seleziona tutti i record presenti nella tabella Libri dove il nome dell'editore inizia con la lettera A; % svolge le funzioni di carattere jolly

- `SELECT * FROM "Libri" WHERE Editore LIKE "%A%";`

Seleziona tutti i record presenti nella tabella Libri dove il nome dell'editore contiene al suo interno la lettera A;

Query di Selezione Dati 3

La Clausola ORDER BY (ASC/DESC)

- `SELECT * FROM "Libri" ORDER BY Titolo ASC;`
Seleziona i record presenti nella tabella Libri in ordine crescente (ASCending) per Titolo
- `SELECT * FROM "Libri" ORDER BY Autore;`
Seleziona tutti i record presenti nella tabella Libri in ordine crescente per Autore
- `SELECT * FROM "Libri" ORDER BY Prezzo DESC;`
Seleziona tutti i record presenti nella tabella Libri in ordine decrescente di Prezzo

Query di Selezione Dati 4

Le Clausole IN e BETWEEN

- `SELECT * FROM "Libri" WHERE Editore IN ("Apogeo", "Hoepli");`
Seleziona tutti i record presenti nella tabella Libri dove l'editore è Apogeo oppure Hoepli;
- `SELECT * FROM "Libri" WHERE Prezzo BETWEEN 10 AND 15;`
Seleziona tutti i record presenti nella tabella Libri dove il prezzo è compreso fra 10€ e 15€;

SQL: Funzioni di Aggregazione 1

MAX, MIN, COUNT

- `SELECT COUNT (*) FROM "Libri" WHERE (Editore = "Apogeo");`
Conta il numero di record presenti nella tabella "Libri" pubblicati dalla casa editrice Apogeo;
- `SELECT MIN (Prezzo) FROM "Libri" WHERE (Editore = "Hoepli");`
Individua il prezzo minimo, nella tabella "Libri", dei volumi pubblicati dalla casa editrice Hoepli;
- `SELECT MAX (Prezzo) FROM "Libri" WHERE (Editore = "Hoepli");`
Individua il prezzo massimo, nella tabella "Libri", dei volumi pubblicati dalla casa editrice Hoepli;

SQL: Funzioni di Aggregazione 2

SUM, AVG

- `SELECT SUM (Prezzo) FROM "Libri" WHERE (Editore = "Hoepli");`
Calcola la somma dei prezzi, nella tabella "Libri", dei volumi pubblicati dalla casa editrice Hoepli;
- `SELECT AVG (Prezzo) FROM "Libri" WHERE (Editore = "Apogeo");`
Individua il prezzo medio, nella tabella "Libri", dei volumi pubblicati dalla casa editrice Apogeo;

Query di Cancellazione Dati

Cancella uno o più record in una tabella

- `DELETE FROM "Libri" WHERE (Editore = "Apogeo");`
Cancella dalla tabella "Libri" tutti i record dove l'editore è Apogeo;
- `DELETE FROM "Libri";`
Cancella dalla tabella "Libri" tutti i record;

Query di Aggiornamento Dati

Modifica uno o più record in una tabella

```
UPDATE "Libri" SET Prezzo = 10.50 WHERE (Codice = 32);
```

Aggiorna il campo Prezzo della tabella "Libri" dove il Codice vale 32;

```
UPDATE "Libri" SET Codice = 23, Autore = "Cleto Azzani", Titolo = "Corso  
Introduttivo MPU MC6800", Editore = "Motorola Semiconductors", ISBN =  
"0000000000", Prezzo = 12.40 WHERE (Codice = 32);
```

Aggiorna i campi Codice, Autore, Titolo, Editore, ISBN, Prezzo della tabella "Libri" con i nuovi valori che seguono la frase SET ma solamente all'interno del record dove il Codice vale 32;

Bibliografia

- "Introduction to Structured Query Language ver. 4.71" di James Hoffmann,
<http://w3.one.net/~jhoffman/sqltut.htm>
- Appunti sui DataBase Relazionali e sul linguaggio SQL di Pietro Suffritti,
<http://www.suffritti.it>
- Local SQL Help di Borland Delphi5 Professional