

EP01_Introduzione ai DB e alle tabelle

I sistemi di gestione delle basi di dati, noti con l'acronimo DBMS (Database Management System), rappresentano l'infrastruttura fondamentale attraverso cui si creano e gestiscono le basi di dati. Questi sistemi si distinguono principalmente in due categorie architetturali che riflettono differenti filosofie di organizzazione dei dati.

I DBMS relazionali costituiscono la categoria più diffusa e si fondono su un principio organizzativo preciso: i dati vengono salvati in tabelle che possono essere potenzialmente collegate tra loro attraverso relazioni logiche. Questa struttura tabellare permette di stabilire connessioni tra diverse entità informative, creando una rete di dati interconnessi che rispecchia la complessità delle relazioni del mondo reale. Il linguaggio utilizzato per interagire con questi sistemi è SQL, ovvero Structured Query Language, che fornisce una sintassi standardizzata per comunicare con il database.

In opposizione a questo modello si collocano i DBMS non relazionali, che adottano paradigmi organizzativi alternativi. Questi sistemi organizzano i dati secondo modelli non tabellari, come ad esempio le collezioni di documenti, e non utilizzano SQL come linguaggio di interrogazione. La scelta tra le due architetture dipende dalla natura dei dati da gestire e dalle esigenze specifiche dell'applicazione.

Indipendentemente dall'architettura scelta, ogni database deve garantire quattro funzionalità fondamentali che ne definiscono l'utilità pratica. La memorizzazione costituisce la capacità primaria di un database, permettendo di rendere persistenti i dati in modo che rimangano duraturi e accessibili nel tempo. Questa persistenza rappresenta un requisito essenziale per qualsiasi sistema informativo che debba conservare informazioni oltre la durata di una singola esecuzione del programma.

L'organizzazione rappresenta un'altra caratteristica distintiva dei database moderni. Un database non si limita a memorizzare dati in modo casuale, ma li organizza secondo concetti strutturali ben definiti che vanno oltre la semplice sequenza cronologica. Questa organizzazione concettuale permette di

rappresentare relazioni complesse e gerarchie informative che riflettono la struttura logica del dominio applicativo

La ricerca costituisce la terza funzionalità essenziale e rappresenta uno degli aspetti più utilizzati nella pratica. La maggior parte dei database dispone di linguaggi di interrogazione interni che rendono particolarmente semplice recuperare l'oggetto desiderato attraverso query strutturate. Un esempio tipico potrebbe essere la ricerca di tutti i documenti appartenenti a una specifica categoria tematica, operazione che il sistema deve essere in grado di eseguire in modo efficiente anche su grandi volumi di dati.

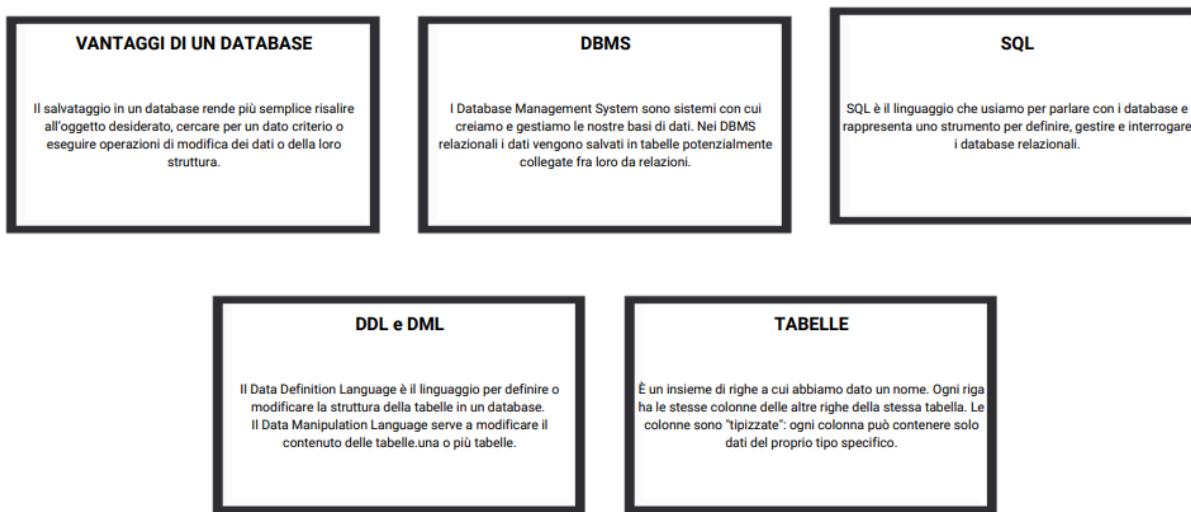
Infine, la modifica rappresenta la capacità di un database di evolvere nel tempo. Attraverso un database è possibile eseguire operazioni di modifica sia dei dati che della struttura stessa. Questa flessibilità permette al sistema di adattarsi a requisiti mutevoli, consentendo ad esempio di aumentare lo stipendio di un dipendente specifico o di aggiungere il campo della specializzazione per tutti i dottori presenti nel sistema.

Nel contesto dei database relazionali che utilizzano SQL, il linguaggio si articola in diverse categorie di comandi che servono scopi distinti. Il DDL, acronimo di Data Definition Language, rappresenta l'insieme dei comandi utilizzati per modificare o definire la struttura delle tabelle. Questi comandi operano a livello dello schema del database e includono istruzioni per creare nuove tabelle, alterare quelle esistenti, eliminarle completamente o rimuovere tutti i dati mantenendo la struttura. Le operazioni DDL sono tipicamente irreversibili e modificano la struttura del database senza influenzare direttamente i dati contenuti nelle tabelle. Queste istruzioni vengono generalmente eseguite con minore frequenza rispetto ad altri tipi di comandi e sono tipicamente di competenza degli amministratori di database.

Il DML, ovvero Data Manipulation Language, costituisce invece l'insieme dei comandi dedicati alla modifica del contenuto delle tabelle. Mentre il DDL definisce la struttura, il DML manipola i dati all'interno di quella struttura predefinita. Attraverso il DML è possibile inserire nuovi record, aggiornare informazioni esistenti, eliminare righe specifiche e soprattutto recuperare dati attraverso interrogazioni. A differenza del DDL, le operazioni DML hanno un impatto diretto sui dati contenuti nel database e possono essere reversibili attraverso meccanismi di controllo transazionale. Questi comandi vengono eseguiti con maggiore

frequenza e sono tipicamente utilizzati da sviluppatori di applicazioni e utenti finali che necessitano di interagire quotidianamente con i dati.

SQL nel suo complesso rappresenta il linguaggio generale che comprende tutti i comandi necessari per interrogare e gestire database relazionali. La distinzione tra DDL e DML evidenzia la separazione concettuale tra la definizione della struttura dati e la manipolazione del contenuto, due aspetti che richiedono competenze e privilegi diversi all'interno di un'organizzazione. Questa separazione permette di mantenere un controllo granulare sulle operazioni che diversi utenti possono eseguire, garantendo sia la flessibilità operativa che la sicurezza e l'integrità del sistema informativo



Il comando CREATE TABLE rappresenta l'istruzione fondamentale del Data Definition Language che permette di creare una nuova tabella all'interno di un database già esistente. Questo comando costituisce un esempio paradigmatico di operazione DDL poiché definisce la struttura dati senza ancora manipolare alcun contenuto. La sintassi richiede di specificare il nome della tabella seguito dalla definizione di ogni colonna con il rispettivo tipo di dato e dimensione. Nel processo di creazione è possibile definire vincoli strutturali come le chiavi primarie, che identificano univocamente ogni riga della tabella, oltre ad altri attributi come il formato delle date. La chiave primaria assume particolare rilevanza in quanto garantisce l'unicità di ogni record e costituisce il fondamento

per stabilire relazioni con altre tabelle del database.[ppl-ai-file-upload.s3.amazonaws]w3schools+2

Il comando INSERT appartiene invece al Data Manipulation Language e rappresenta l'operazione attraverso cui si inseriscono nuove righe all'interno di una tabella. Con questo comando diventa possibile aggiungere nuovi dati specificando i valori per ciascuna colonna definita nella struttura della tabella. La sintassi richiede di indicare il nome della tabella seguito dai nomi delle colonne interessate e dai corrispondenti valori da inserire. È possibile inserire uno o più record in un'unica operazione, specificando i valori tra parentesi e separando le diverse righe con virgole. Questo comando opera esclusivamente sul contenuto senza modificare la struttura sottostante della tabella.dev+1[ppl-ai-file-upload.s3.amazonaws]

Il comando UPDATE consente di modificare i contenuti esistenti all'interno di una tabella. Attraverso questa istruzione DML è possibile aggiornare i valori di una o più colonne per specifici record che soddisfano determinate condizioni. La clausola WHERE assume un ruolo cruciale in quanto specifica quali righe devono essere modificate, evitando di alterare indiscriminatamente l'intera tabella. L'assenza della clausola WHERE comporterebbe infatti l'aggiornamento di tutti i record presenti nella tabella, operazione raramente desiderata nella pratica. Il comando permette di correggere errori, aggiornare informazioni divenute obsolete o apportare modifiche necessarie mantenendo inalterata la struttura della tabella. [secoda][ppl-ai-file-upload.s3.amazonaws]

Il comando DELETE serve a cancellare i contenuti di una tabella rimuovendo specifiche righe. Questa operazione DML permette di eliminare record che soddisfano determinate condizioni espresse attraverso la clausola WHERE. La specificazione della condizione risulta fondamentale poiché in sua assenza il comando procederebbe alla cancellazione di tutti i record presenti nella tabella, svuotandola completamente pur mantenendone la struttura. Questo comportamento evidenzia la differenza concettuale tra cancellare dati e cancellare la struttura stessa, operazione quest'ultima che richiederebbe invece un comando DDL come DROP.mssqltips+1[ppl-ai-file-upload.s3.amazonaws]

Il comando SELECT costituisce l'operazione di interrogazione per eccellenza nel linguaggio SQL. Attraverso questa istruzione è possibile effettuare query che restituiscono dati filtrati in base a criteri specificati. La clausola WHERE permette

di definire condizioni che i record devono soddisfare per essere inclusi nel risultato dell'interrogazione. Il SELECT rappresenta una forma particolare di comando DML in quanto non modifica i dati ma si limita a recuperarli e presentarli secondo le specifiche della query. L'interrogazione può coinvolgere una o più tabelle e restituisce come risultato un insieme di dati che soddisfano le condizioni imposte, permettendo così di estrarre informazioni significative dalla massa di dati memorizzati nel database.[\[ppl-ai-file-upload.s3.amazonaws\]](#)[\[secoda\]](#)

La distinzione tra comandi DDL come CREATE TABLE e comandi DML come INSERT, UPDATE, DELETE e SELECT riflette la separazione fondamentale tra la definizione della struttura dati e la manipolazione del contenuto. Mentre il DDL opera a livello dello schema definendo come i dati devono essere organizzati, il DML lavora all'interno di quella struttura predefinita per aggiungere, modificare, rimuovere o consultare i dati effettivamente memorizzati. Questa architettura stratificata permette di mantenere separati gli aspetti strutturali da quelli operativi, garantendo maggiore controllo e sicurezza nella gestione delle basi di dati.[\[w3schools+3\]](#)[\[ppl-ai-file-upload.s3.amazonaws\]](#)

CREATE TABLE

Questo comando crea una nuova tabella in un database già esistente ed è un esempio di DDL (Data Definition Language).

```
USE test;
CREATE TABLE Person
id int primary key, name varchar(100), surname
varchar(100), dateofbirth date
```

INSERT

Con questo comando è possibile, ad esempio, inserire nuove righe all'interno di una tabella. È un esempio di DML (Data Manipulation Language).

```
INSERT INTO Person
(id, name, surname, dateofbirth)
VALUES (1, 'F', 'P', '1980-05-02'),
(2, 'S', 'R', '1995-01-01'),
(3, 'A', 'B', '1996-01-01');
```

UPDATE

Con questo comando è possibile modificare i contenuti di una tabella.

```
UPDATE Person set dateofbirth="1995-02-01"
where id=2;
```

DELETE

Con questo comando è possibile cancellare i contenuti di una tabella.

```
DELETE FROM Person where id=3;
```

SELECT

Con questo comando è possibile effettuare una "query" ossia una interrogazione dei contenuti di una tabella che ci restituisce come risultato i dati filtrati.

```
SELECT name, surname from Person where
dateofbirth < '1990-01-01';
```