

Operatori vari

Between

È un operatore di SQL utilizzato nelle query che mi permette di impostare un intervallo che viene delimitato da 2 limiti (**compresi**). I valori possono essere numerici, text o date.

Formato

```
SELECT column name (s)
FROM table_name
WHERE column name BETWEEN value1 AND value 2;
```

Esempio su tblTracks

```
SELECT *
FROM tblTracks
WHERE Milliseconds BETWEEN 130000 AND 180000;
```

IN

È un operatore che permette di indicare più valori inclusi in un insieme. È utilizzato per abbreviare una serie di condizioni legate dall'operatore OR

Formato

```
SELECT column_name(s)
FROM table_name
WHERE column_name IN (value1, value2, ...);
```

Esempio su tblTracks

```
SELECT *
FROM tblTracks
```

```
WHERE Composer IN ('Nelson Angelo', 'Kurt Cobain', 'R.E.M.');
```

SQL spinto!! Per il momento solo per chi è coraggioso!

```
SELECT column_name(s)
```

```
FROM table_name
```

```
WHERE column_name IN (SELECT STATEMENT);
```

Like

È un operatore che permette di **ricercare** all'interno di una colonna uno specifico **pattern** utilizzando caratteri speciali in sostituzione di **uno o più caratteri**. Questi caratteri speciali vengono anche chiamati **Caratteri Jolly** proprio perché hanno la funzione di sostituire altri caratteri.

I caratteri

- **%** → Sostituisce zero, uno o più caratteri
- **_** → Sostituisce **un solo** carattere

Formato

```
SELECT column1, column2, ...
```

```
FROM table_name
```

```
WHERE columnN LIKE pattern;
```

Esempi

```
SELECT *
```

```
FROM tblTracks
```

```
WHERE Composer LIKE 'N%';
```

```
SELECT *  
FROM tblTracks  
WHERE Composer LIKE 'Ni%';
```

```
SELECT *  
FROM tblTracks  
WHERE Composer LIKE 'Ni_____';
```

Il JOIN

Una clausola **JOIN** viene utilizzata per combinare righe di due o più tabelle, in base a una colonna correlata tra di loro.

La parola chiave **INNER JOIN** seleziona i record che hanno valori corrispondenti in **entrambe** le tabelle.

Formato

```
SELECT column_name(s)  
FROM table1  
INNER JOIN table2 ON table1.column_name=table2.column_name;
```

Esempio tblTracks

```
SELECT *  
FROM tblTracks  
INNER JOIN tblAlbums ON tblTracks.AlbumId=tblAlbums.idAlbum  
WHERE Title='Audioslave' AND Name LIKE 'S%';
```

(tblAlbums.title quando abbiamo title in più classi)

Esempio 2: visualizzare per l'album Audio Slave dove il nome della traccia inizia per S e voglio vedere il nome, il prezzo e il titolo.

```
SELECT tblTracks.Name, tblTracks.UnitPrice, tblAlbums.Title  
FROM tblTracks  
INNER JOIN tblAlbums ON tblTracks.AlbumId=tblAlbums.idAlbum  
WHERE Title='Audioslave' AND Name LIKE 'S%';
```

(tblAlbums.title quando abbiamo title in più classi)

Dove mettiamo le query?

Le mettiamo su views. Si potrà lavorare su di esse anche su blazor.

Funzioni di aggregazione

Count

Conta quanti valori ci sono in una tabella. Conta il numero di righe presenti in una tabella, la cardinalità di una relazione. Primo caso: non vengono contati i null, solo i valori del campo specificato (restituisce un numero).

Secondo caso: vengono contati tutti i valori.

```
SELECT COUNT (nomecampo)
```

```
FROM nometabella;
```

Oppure

```
SELECT COUNT (*)
```

```
FROM nometabella;
```

Oppure

```
SELECT COUNT (*) AS [Numero Prodotti Commercializzati]
```

```
FROM nometabella;
```

Sum

Somma tutti i valori numerici contenuti in una colonna.

```
SELECT SUM (scorte)
```

```
FROM tblProdotti
```

Oppure

```
SELECT SUM (scorte) AS [Totale Scorte]
```

```
FROM tblProdotti
```

Avg

Fa la media aritmetica (o pesata) dei valori contenuti in una determinata tabella.

```
SELECT AVG (PrezzoUnitario)
```

```
FROM tblProdotti;
```

Oppure

```
SELECT AVG (PrezzoUnitario*Scorte)
```

```
FROM tblProdotti;
```

Min e Max

Restituisce il minimo o il massimo tra i valori