

SQL: LIKE Condition

SELECT campi

FROM tabella

WHERE campo **LIKE** [stringa con caratteri jolly]

consente di utilizzare i caratteri jolly per eseguire la corrispondenza del modello in una query.

La condizione LIKE viene utilizzata nella clausola WHERE di un'istruzione SELECT, INSERT, UPDATE o DELETE.

Caratteri Jolly:

jolly	Spiegazione
%	Consente di sostituire una parte di stringa di qualsiasi lunghezza (compresa la lunghezza zero)
-	Permette di sostituire un singolo carattere
[]	Permette di far corrispondere qualsiasi carattere tra parentesi [] (ad esempio, [abc] corrisponde ai caratteri a, b, o c)
[^]	Permette di far corrispondere qualsiasi carattere non nelle parentesi [^] (ad esempio, [^ abc] corrisponde a qualsiasi carattere che non sia a, b, o c caratteri)

Esempio: utilizzo di % jolly nella condizione LIKE

Spieghiamo come funziona il carattere jolly % nella condizione SQL LIKE. Ricorda che il carattere jolly % corrisponde a qualsiasi stringa di qualsiasi lunghezza (compresa la lunghezza zero).

In questo primo esempio, vogliamo trovare tutti i record nella tabella dei clienti in cui il *cognome* del cliente inizia con "J".

In questo esempio, abbiamo una tabella chiamata *clienti* con i seguenti dati:

4000	Jackson	Joe	techonthenet.com
5000	Fui	Jane	digminecraft.com
6000	Ferguson	Samantha	bigactivities.com
7000	Reynolds	Allen	checkyourmath.com
8000	Anderson	Paige	NULL
9000	Johnson	Derek	techonthenet.com

Immettere la seguente istruzione SQL:

```
SELECT *  
  
FROM clienti  
  
WHERE cognome LIKE 'J%'  
  
ORDER BY cognome;
```

Saranno selezionati 2 record. Questi sono i risultati che dovresti vedere:

```
4000 Jackson Joe    techonthenet.com  
9000 Johnson Derek techonthenet.com
```

Questo esempio restituisce i record nella tabella dei *clienti* in cui il *cognome* inizia con "J"

Poiché la condizione LIKE non è sensibile al maiuscolo / minuscolo, la seguente istruzione SQL restituisce gli stessi risultati:

```
SELECT *  
  
FROM clienti  
  
WHERE cognome LIKE 'j%'  
  
ORDER BY cognome;
```

Utilizzo di più caratteri jolly % nella condizione LIKE

È inoltre possibile utilizzare la % wildcard più volte con la condizione LIKE.

Utilizzando la stessa tabella *clienti* con i seguenti dati:

```
4000 Jackson Joe    techonthenet.com  
5000 Fui    Jane    digminecraft.com  
6000 Ferguson Samantha bigactivities.com  
7000 Reynolds Allen    checkyourmath.com  
8000 Anderson Paige    NULL  
9000 Johnson Derek    techonthenet.com
```

Proviamo a trovare tutti i valori *last_name* dalla tabella dei *clienti* in cui il *cognome* contiene la lettera 'e'.
Immettere la seguente istruzione SQL:

```
SELECT *  
  
FROM clienti  
  
WHERE cognome LIKE '%e%'
```

Saranno selezionati 3 record. Questi sono i risultati che dovresti vedere:

cognome
Anderson
Ferguson
Reynolds

In questo esempio, i cognomi Anderson, Ferguson e Reynolds contengono la lettera "e".

Esempio: utilizzo di _ (trattino jolly) nella condizione LIKE

Quindi, spieghiamo come funziona la _ wildcard (carattere jolly) nella condizione LIKE.

Ricorda che il _ (trattino jolly) sostituisce esattamente un carattere, a differenza del carattere jolly % .

Utilizzando la tabella delle *categorie* con i seguenti dati:

category_id	categoria
25	gastronomia
50	Produrre
75	Forno
100	Merchandise generale
125	Tecnologia

Proviamo a trovare tutti i record dalla tabella delle *categorie* in cui *category_id* è lungo 2 cifre e termina con '5'. Immettere la seguente istruzione SQL:

```
SELECT *  
  
FROM categorie  
  
WHERE category_id LIKE '_5';
```

Saranno selezionati 2 record. Questi sono i risultati che dovresti vedere:

category_id	categoria
25	gastronomia
75	Forno

Utilizzo di caratteri jolly multipli _ nella condizione LIKE

Se si desidera associare un valore a 3 cifre terminato con "5", è necessario utilizzare il carattere jolly _ due volte. È possibile modificare la query come segue:

```
SELECT *
FROM categorie
WHERE category_id LIKE '__5'; /** due trattini **/
```

Ora restituirai il valore *category_id* di 125:

category_id	categoria
125	Tecnologia

Esempio: utilizzo dell'operatore NOT con la condizione LIKE

Successivamente, esaminiamo un esempio di come utilizzare l'[operatore NOT](#) con la condizione LIKE.

In questo esempio, abbiamo una tabella denominata *fornitori* con i seguenti dati:

id fornitore	città	stato
100 Microsoft	Redmond	Washington
200 Google	vista delle montagne	California
300 Oracolo	Redwood City	California
400 Kimberly-Clark	Irving	Texas
500 Tyson Foods	Springdale	Arkansas
600 SC Johnson	Racine	Wisconsin
700 Dole Food Company	Westlake Village	California
800 Alimenti di fiori	Thomasville	Georgia
900 Arti elettroniche	Redwood City	California

Cerchiamo tutti i record nella tabella dei *fornitori* in cui il *supplier_name* **non** contiene la lettera "o". Immettere la seguente istruzione SQL:

```
SELECT *
FROM fornitori
WHERE fornitore NOT LIKE '%o%';
```

Ci sarà 1 record selezionato. Questi sono i risultati che dovresti vedere:

id fornitore	città	stato
400 Kimberly-Clark	Irving	Texas

In questo esempio, nella tabella dei *fornitori* esiste un solo record in cui il campo *fornitore* non contiene la lettera "o".

Esempio: utilizzo del carattere jolly [] (parentesi quadre parentesi)

Il carattere jolly [] (parentesi quadre) → ciò che è contenuto nelle parentesi quadre sono i caratteri che vengono considerati.

Per esempio:

```
SELECT *  
FROM employees  
WHERE first_name LIKE 'Sm[iy]th';
```

Questo esempio di condizione di SQL Server restituirebbe tutti i dipendenti il cui *first_name* è lungo 5 caratteri, in cui i primi due caratteri sono "Sm" e gli ultimi due caratteri è "th" e il terzo carattere è "i" o "y". Quindi, in questo caso, corrisponderebbe a "Smith" o a "Smyth".

Esempio: utilizzo del carattere jolly [^] (parentesi quadre con ^ carattere jolly)

Ciò che è contenuto nelle parentesi quadre sono caratteri a cui NON vuoi corrispondere.

Per esempio:

```
SELECT *  
FROM employees  
WHERE first_name LIKE 'Sm[^iy]th';
```

Questo esempio di condizione di SQL Server restituirebbe tutti i dipendenti il cui *first_name* è lungo 5 caratteri, in cui i primi due caratteri sono "Sm" e gli ultimi due caratteri è "th" e il terzo carattere non è né "i" né "y". Quindi, in questo caso, corrisponderebbe a valori come "Smath", "Smeth", "Smoth", ecc. Ma non corrisponderebbe né a "Smith" né a "Smyth".

Escaping caratteri jolly (% e _) in SQL

Esempio: come si trovano tutti i record con un campo “parte” che contiene un carattere di sottolineatura?

```
SELECT *  
FROM tabella  
WHERE parte LIKE '% _%'
```

non funziona come vorremmo, perché i due % (percento) vengono interpretati come caratteri jolly!

Supponiamo di avere i seguenti dati:

```
PARTE  
-----  
test1  
TEST_1  
Test 1%  
test2  
test_2  
Test 2%
```

e vogliamo scrivere una query per restituire TEST_1 e test_2 perché hanno un trattino.

Dobbiamo dire al motore SQL che il carattere di sottolineatura dovrebbe essere considerato un carattere di sottolineatura semplice, non un carattere jolly.

L'espressione ESCAPE nella [clausola LIKE](#) permette di ottenere proprio questo: togliere il significato speciale dai caratteri jolly.

Quindi (se decidiamo di usare ad esempio il carattere barra davanti al carattere di sottolineatura come carattere di escape) il seguente codice fa al caso nostro:

```
SELECT *  
FROM tabella  
WHERE parte LIKE '% \_%' ESCAPE '\\'
```

Esercizio: estrarre “Test 1%” e “Test 2%”