

Riassunto Finale di MySQL

Prof. Francesco Gobbi
I.I.S.S. Galileo Galilei, Ostiglia

16 giugno 2025

Ogni comando MySQL è presentato con la sua struttura generale e un esempio concreto.

1. SELECT

Schema generale:

```
SELECT colonna1, colonna2, ...  
FROM nome_tabella  
[WHERE condizione]  
[ORDER BY colonna]  
[LIMIT numero];
```

Esempio:

```
SELECT nome, cognome  
FROM studenti  
WHERE classe = 5  
ORDER BY cognome  
LIMIT 10;
```

2. INSERT

Schema generale:

```
INSERT INTO nome_tabella (colonna1, colonna2, ...)  
VALUES (valore1, valore2, ...);
```

Esempio:

```
INSERT INTO studenti (nome, cognome, classe)  
VALUES ('Anna', 'Bianchi', 5);
```

3. UPDATE

Schema generale:

```
UPDATE nome_tabella  
SET colonna1 = valore1, colonna2 = valore2, ...  
WHERE condizione;
```

Esempio:

```
UPDATE studenti  
SET classe = 4  
WHERE nome = 'Anna' AND cognome = 'Bianchi';
```

4. DELETE

Schema generale:

```
DELETE FROM nome_tabella  
WHERE condizione;
```

Esempio:

```
DELETE FROM studenti  
WHERE classe = 3;
```

5. CREATE TABLE

Schema generale:

```
CREATE TABLE nome_tabella (  
    colonna1 TIPO [VINCOLO],  
    colonna2 TIPO [VINCOLO],  
    ...  
);
```

Esempio:

```
CREATE TABLE studenti (  
    id INT PRIMARY KEY AUTO_INCREMENT,  
    nome VARCHAR(50) NOT NULL,  
    cognome VARCHAR(50) NOT NULL,  
    classe INT  
);
```

6. ALTER TABLE

Schema generale:

```
ALTER TABLE nome_tabella  
AZIONE ...;
```

Esempio (aggiunta colonna):

```
ALTER TABLE studenti  
ADD COLUMN email VARCHAR(100);
```

7. DROP TABLE

Schema generale:

```
DROP TABLE nome_tabella;
```

Esempio:

```
DROP TABLE studenti;
```

8. WHERE

Schema generale (usato nelle query):

```
... FROM nome_tabella  
WHERE condizione;
```

Esempio:

```
SELECT * FROM studenti  
WHERE classe = 5 AND cognome = 'Rossi';
```

9. ORDER BY

Schema generale:

```
... FROM nome_tabella  
ORDER BY colonna [ASC|DESC];
```

Esempio:

```
SELECT nome, cognome FROM studenti  
ORDER BY cognome ASC;
```

10. GROUP BY e HAVING

Schema generale:

```
SELECT colonna, FUNZIONE_AGGREGATA(colonna2)  
FROM nome_tabella  
GROUP BY colonna  
[HAVING condizione_su_gruppi];
```

Esempio:

```
SELECT classe, COUNT(*) AS numero_studenti  
FROM studenti  
GROUP BY classe  
HAVING numero_studenti > 10;
```

11. JOIN (equi-join con WHERE)

Schema generale:

```
SELECT ...  
FROM tabella1, tabella2  
WHERE tabella1.colonna_fk = tabella2.colonna_pk  
[AND ...];
```

Esempio:

```
SELECT s.nome, c.nome AS classe  
FROM studenti AS s, classi AS c  
WHERE s.classe_id = c.id;
```

12. Funzioni di Aggregazione

Schema generale:

```
SELECT FUNZIONE_AGGREGATA(colonna)  
FROM nome_tabella  
[WHERE condizione];
```

Esempio:

```
SELECT AVG(voto) AS media_voti, MAX(voto) AS voto_massimo  
FROM esami;
```

13. Subquery

Schema generale:

```
SELECT ...  
FROM nome_tabella  
WHERE colonna IN (  
    SELECT ... FROM altra_tabella WHERE condizione  
);
```

Esempio:

```
SELECT nome, cognome  
FROM studenti  
WHERE classe_id IN (  
    SELECT id FROM classi WHERE nome = '5A'  
);
```

14. LIMIT e OFFSET

Schema generale:

```
SELECT ...  
FROM nome_tabella  
LIMIT numero  
[OFFSET n];
```

Esempio:

```
SELECT * FROM studenti  
ORDER BY nome  
LIMIT 5 OFFSET 10;
```

15. CREATE INDEX

Schema generale:

```
CREATE INDEX nome_indice  
ON nome_tabella (colonna);
```

Esempio:

```
CREATE INDEX idx_cognome  
ON studenti (cognome);
```

16. Vincoli: PRIMARY KEY, FOREIGN KEY, UNIQUE, CHECK, NOT NULL

Schema generale (in CREATE TABLE):

```
CREATE TABLE nome_tabella (  
  colonna1 TIPO PRIMARY KEY,  
  colonna2 TIPO UNIQUE,  
  colonna3 TIPO NOT NULL,  
  colonna4 TIPO CHECK (condizione),  
  colonna5 TIPO,  
  FOREIGN KEY (colonna5) REFERENCES altra_tabella(colonna_pk)  
);
```

Esempio:

```
CREATE TABLE iscrizioni (  
  id INT PRIMARY KEY AUTO_INCREMENT,  
  studente_id INT NOT NULL,  
  FOREIGN KEY (studente_id) REFERENCES studenti(id),  
  UNIQUE (studente_id)  
);
```