

PRONTUARIO SINTASSI COMANDI SQL

| COMANDO | SEMANTICA |
|--------------------------------|----------------------------------------------------------------------------------------------------------|
| CREATE DATABASE nome_database; | Crea un nuovo database |
| DROP DATABASE nome_database; | Elimina un database esistente |
| CREATE TABLE nome_tabella | Creare una nuova tabella |
| ALTER TABLE nome_tabella | Modificare la struttura di una tabella (aggiunge, rimuove, rinomina colonne, cambia tipo di una colonna) |
| DROP TABLE nome_tabella | Elimina una tabella esistente |
| INSERT INTO nome_tabella | Inserisce valori in certi campi di una nuova riga |
| UPDATE nome_tabella | Cambia valori in certi campi di una riga esistente |
| DELETE FROM nome_tabella | Elimina una riga in base a certi criteri |
| SELECT | Estrapola campi, righe, raggruppa, ordina, calcola |
| GRANT | Garantisce privilegi sulle tabelle agli utenti |
| REVOKE | Revoca privilegi sulle tabelle agli utenti |

_____ COMANDI _____ NB: gli elementi tra [] sono Opzionali; _____

CREATE TABLE *nome_tabella*

(
 nome_campo1 *tipo_di_dato(size)* [vincoli di campo] ,
 nome_campo1 2 *tipo_di_dato(size)* [vincoli di campo] ,
 nome_campo1 3 *tipo_di_dato(size)* [vincoli di campo] ,

 [PRIMARY KEY(nome_campo, [nome_campo, ..., nome_campo])]
 [FOREIGN KEY(nome_campo) REFERENCES nome_tabella[(nome_campo)]]
 [UNIQUE(nome_campo)]
 [CHECK(expr*)]
);

i vincoli di campo possono essere:

| | |
|-----------------------|--------------------------------------------------------------------|
| NOT NULL | La colonna non può memorizzare valori nulli |
| AUTO_INCREMENT | Il valore di tipo numerico si incrementa ad ogni inserimento |
| UNSIGNED | Senza segno e cioè positivo |
| DEFAULT | Valore di Default per la colonna |
| UNIQUE | Ogni riga avrà un valore unico sulla colonna |
| PRIMARY KEY | chiave primaria, UNIQUE e NOT NULL insieme, |
| FOREIGN KEY | Chiave esterna, assicura integrità referenziale a un'altra tabella |
| CHECK | Vincolo sulle colonne |

INSERT INTO nome_tabella VALUES (valore1, valore2, valore3,...); oppure:

INSERT INTO nome_tabella (*colonna1,colonna2,colonna3,...*) VALUES (valore1, valore2, valore3,...);

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

DELETE FROM nome_tabella WHERE *condizione**;

```

SELECT [ALL | DISTINCT] lista_campi_separati_da_virgola
FROM lista_nomi_tabelle_separati_da_virgola
[WHERE condizione*]
[GROUP BY lista_campi_separati_da_virgola]
[HAVING condizione*]
[ORDER BY lista_campi_separati_da_virgola [ASC | DESC] ]
[LIMIT numero_di_righe [OFFSET numero_iniziale] ]

```

nella clausola SELECT è possibile usare al posto di lista_campi_separati_da_virgola

- * che vuol dire tutti i campi
- funzioni di aggregazione sui domini separati da virgola (da soli o con campi delle tabelle, vedi sotto)
- alias di campi tramite: AS alias_per_il_campo a destra del campo
- espressione con semplici calcoli aritmetici (+, -, *, /) tra (valori dei) campi

*la condizione che si inserisce nel WHERE, nell'HAVING o anche come espressione nel CHECK (vedi CREATE TABLE) deve essere sempre di uno dei seguenti tipi:

- operando **operatoreBinario** operando
- **operatoreUnario** operando
- operando **operatoreUnario**

dove:

- operando può essere: un nome di campo, un valore, una condizione (racchiusa tra parentesi)
- operatore può essere:
 - di confronto: <, <=, >, >=, =, != (diverso)
 - logico: AND, OR, NOT (operatore unario)
 - operatore LIKE
 - BETWEEN, [NOT] IN, IS [NOT] NULL con le seguenti sintassi:
 - nome_campo BETWEEN valore1 AND valore2,
 - nome_campo [NOT] IN ('valore1', 'valore2', ...)
 - nome_campo IS [NOT] NULL

Funzioni di aggregazione sui domini:

si possono usare nella clausola SELECT singolarmente o separati da virgola al posto di lista_campi_separati_da_virgola

se usate assieme a uno o più campi delle tabelle è obbligatorio l'uso del GROUP BY

| | |
|----------------------------|-----------------------------------------------------------------------------------------------------------------------------|
| COUNT(*) | Riporta il numero di righe della tabella |
| COUNT(nome_campo) | Riporta il numero di valori non nulli del campo |
| COUNT(DISTINCT nome_campo) | Riporta il numero di valori univoci non nulli del campo |
| SUM(nome_campo) | Somma i valori della colonna |
| MIN(nome_campo) | Calcola il minimo valore della colonna |
| MAX(nome_campo) | Calcola il massimo valore della colonna |
| AVG(nome_campo) | Calcola la media aritmetica tra tutti i valori della colonna, applicato solo a campi numerici, i valori nulli sono ignorati |