**ALTER TABLE** 9

**ALTER:** 8

**AVG** 5

**CHECK TABLE :** 7

**COMMIT** 7

**COUNT :** 5

**CREATE :** 8

**CREATE TABLE** 9

**CREATE VIEW** 7

**Delete :** 2

**DROP** 8

**DROP TABLE** 9

**DROP VIEW** 7

**Funzioni di aggregazione :** 5

**GRANT** 7

**GROUP BY e HAVING :** 6

**Insert :** 2

**Join :** 4

**MAX** 5

**MIN** 5

**Operatori BETWEEN e LIKE :** 3

**Operatori IN e IS NULL :** 3

**ORDER BY :** 6

**RECOVER TABLE :** 7

**RENAME** 8

**RENAME TABLE** 9

**REPAIR TABLE :** 7

**REVOKE** 7

**ROLLBACK** 7

**SAVEPOINT** 7

**Select :** 3

**Subquery :** 6

**SUM** 5

**Tabelle** 9

**TRUNCATE** 8

**TRUNCATE TABLE** 9

**Update :** 2

**DML (Data Manipulation Language)**

**Insert :**

*L’Istruzione Insert viene utilizzata per inserire nuovi record in una tabella.*

INSERT INTO table\_name (column1, column2, column3, …)

VALUES (value1, value2, value3, …);

**Update :**

*L’Istruzione Update viene utilizzata per modificare i record esistenti in una tabella.*

UPDATE table\_name

SET column1 = value1, column2 = value2, …

WHERE condition;

**Delete :**

*L’Istruzione Delete viene utilizzata per eliminare i record esistenti in una tabella.*

INSERT INTO nomeTabella (colonna1, colonna2, colonna3, …)

VALUES (value1, value2, value3, …);

**QL (Query Language)**

**Select :**

*Il costrutto Select è composto da:.*

* *\_Target list*
* *\_FROM clause*
* *\_WHERE clause*

*Sia nella target list che nella from possono essere inseriti gli alias.*

SELECT column1

FROM table1 AS t1

WHERE condizione;

**Operatori BETWEEN e LIKE :**

*L’operatore BETWEEN serve per verificare che un attributo sia in un certo intervallo.*

SELECT column1

FROM table1 AS t1

WHERE t1.valore BETWEEN 160 && 170;

*L’operatore LIKE serve per effettuare una ricerca su stringhe o caratteri simili.*

SELECT column1,column2

FROM table1 AS t1

WHERE t1.column1 LIKE R\* || t1.column2 LIKE G\*;

**Operatori IN e IS NULL :**

*L’operatore IN confronta un attributo con i valori contenuti all’interno dell’operando.*

SELECT column1

FROM table1 AS t1

WHERE t1.column1 IN (1, 2, 3);

*L’operatore IS NULL serve per vedere se ho un’istanza con un attributo nullo.*

SELECT column1

FROM table1 AS t1

WHERE t1.column1 IS NULL;

**Join :**

*Il costrutto Join serve a combinare i dati di due o più tabelle basandosi su alcune condizioni logiche.*

*Esistono diversi tipi di Join:*

* *INNER JOIN prende solo i record in comune.*
* *LEFT JOIN prende anche i record non comuni della prima tabella nominata*
* *RIGHT JOIN prende anche i record non comuni della seconda tabella nominata*
* *FULL JOIN prende tutti i record non comuni alle tabelle*

SELECT column1, column2

FROM table1 AS t1 INNER JOIN table2 AS t2 ON t1.column1 = t2.column1

WHERE condition;

SELECT column1, column2

FROM table1 AS t1, table2 as t2

WHERE t1.column1 = t2.column1 && condition;

**Funzioni di aggregazione :**

**COUNT :**

*L’operatore COUNT serve per contare le istanze che ha un’entità (conta anche i valori null).*

SELECT COUNT(column1) AS tot

FROM table\_name;

**AVG :**

*L’operatore AVG serve per calcolare la media.*

SELECT AVG(column1) AS average

FROM table\_name;

**MAX :**

*L’operatore MAX restituisce il valore più alto contenuto nella colonna in questione.*

SELECT MAX(column1) AS max

FROM table\_name;

**MIN :**

*L’operatore MIN restituisce il valore più basso contenuto nella colonna in questione.*

SELECT MIN(column1) AS min

FROM table\_name;

**SUM :**

*L’operatore SUM restituisce la somma dei valori contenuti in una colonna.*

SELECT MIN(column1) AS min

FROM table\_name;

**ORDER BY :**

*La parola chiave ORDER BY viene utilizzata per ordinare il set di risultati in ordine crescente o decrescente. Per ordinare i record in ordine decrescente, utilizzare la parola chiave DESC.*

SELECT column1, column2, …

FROM table\_name

ORDER BY column1, column2, … ASC|DESC;

**GROUP BY e HAVING :**

*L’istruzione GROUP BY raggruppa righe con gli stessi valori in righe di riepilogo, l’istruzione GROUP BY viene spesso utilizzata con funzioni aggregate per raggruppare il set di risultati per una o più colonne.*

SELECT column\_name(s)

FROM table\_name

WHERE condition

GROUP BY column\_name(s);

*La clausola HAVING è stata aggiunta a SQL perché non è stato possibile utilizzare la parola chiave WHERE con le funzioni di aggregazione. Si usa solo quando è presente una GROUP BY.*

SELECT column\_name(s)

FROM table\_name

WHERE condition

GROUP BY column\_name(s)

HAVING condition

ORDER BY column\_name(s);

**Subquery :**

*Le Subquery o query annidate sono semplicemente delle query dentro altre query.*

SELECT column\_name(s)

FROM table\_name

WHERE condition > (SELECT column\_name(s)

FROM table\_name

WHERE condition);

**DCL (Data Control Language)**

**COMMIT :**

*con la COMMIT le modifiche effettuate in una transazione diventano permanenti.*

**ROLLBACK :**

*Operazione opposta a quella della COMMIT, riporta lo stato del database all’ultima COMMIT eseguita, annullando, tutte le operazioni effettuate dopo quel momento.*

**SAVEPOINT :**

*Specificare un punto all’interno di una transazione alla quale poi sarà possibile effettuare un ROLLBACK.*

**GRANT :**

*per concedere il permesso( es. Inserimento in tabella, modifica o eliminazione ).*

**REVOKE :**

*per revocare il permesso( es. cancellazione di una tabella ).*

**RECOVER TABLE :**

*consente di recuperare una tabella da una copia di sicurezza.*

**CHECK TABLE :**

*controlla la corrispondenza dei dati tra le tabelle e i suoi indici.*

**REPAIR TABLE :**

*Ricostruisce gli indici quando il comando CHECK non ha funzionato.*

**DROP VIEW :**

*elimina lista.*

**CREATE VIEW :**

*crea lista.*

**DDL (Data Definition Language)**

**CREATE :**

*Per creare database e oggetti simili (tabella, indice, viste, procedura di archiviazione, funzione e trigger)***.**

**ALTER:**

*Modifica la struttura del database esistente.*

**DROP :**

*Cancella gli oggetti dal database.*

**TRUNCATE :**

*Rimuove tutti i record da una tabella, inclusi tutti gli spazi allocati per i record.*

**RENAME :**

*Rinomina un oggetto.*

**Tabelle:**

**CREATE TABLE :**

CREATE TABLE table\_name(

Attr1 tipodiDato(grandezza),

Attr2 tipodiDato(grandezza),

… ,

primary key(attrChiave),

foreign key(attrMigrato) references table\_name2(attrMigrato));

**ALTER TABLE :**

ALTER TABLE table\_name ADD COLUMN attr3 tipodiDato(grandezza);

**DROP TABLE :**

DROP TABLE table\_name

**ALTER TABLE :**

**TRUNCATE TABLE :**

TRUNCATE TABLE table\_name

**ALTER TABLE :**

**RENAME TABLE :**

RENAME TABLE old\_name TO new\_name

**ALTER TABLE :**