Retrieving data

SELECT * FROM **t**;
A simple query to retrieve all rows and columns from *t*

SELECT **c1, c2** FROM **t**;
Query data in columns *c1*, *c2* from a table

SELECT **c1, c2** FROM **t** ORDER BY **c1**;
Query data ordering them by *c1*

SELECT * FROM **t** ORDER BY **c1** ASC;
SELECT * FROM **t** ORDER BY **c1** DESC;
Sort the result set in ascending or descending order

SELECT **c1, c2** FROM **t** LIMIT **number** OFFSET **offset**;
Query *the number* of rows skipping *the offset* of rows

SELECT **c1, c2** FROM **t**
WHERE **c1** BETWEEN **low** AND **high**;
Query rows where value in *c1* is between two values

SELECT **c1, c2** FROM **t1** WHERE **c1** [!] LIKE **pattern**;
Query rows using simple pattern matching % and _ that is processed quickly

SELECT **c1, c2** FROM **t** WHERE c1 [!] IN **v1, v2**;
Query rows where the value of *c1* is **[not]** in the list of *v1 and v2*

SELECT **c1, c2** FROM **t** WHERE **c1** IS [!] NULL;
Check if values in c1 is NULL or not

Retrieving data from multiple tables

SELECT **c1, c2** FROM **t1** UNION [ALL]
SELECT **c1, c2** FROM **t2**;
Combine rows from two queries

SELECT **c1, c2** FROM **t1** INTERSECT
SELECT **c1, c2** FROM **t2**;
Return the intersection of two queries

SELECT **c1, c2** FROM **t1** EXCEPT
SELECT **c1, c2** FROM **t2**;
Subtract a result set from another result set

SELECT **c1, c2** FROM **t1** INNER JOIN **t2**
ON **t1.c1=t2.c3**;
Inner join t1 and t2
If t1.c1!=t2.c3 the row will not be in the result

SELECT **c1, c2** FROM **t1** LEFT JOIN **t2**
ON **t1.c1=t2.c3**;
Left join t1 and t1

SELECT **c1, c2** FROM **t1** LEFT JOIN **t2**
ON **t1.c1=t2.c3**;
Right join t1 and t2

Deleting data

DELETE FROM **t** WHERE **c1=v1**;
Delete subset of rows in the table *t*

TRUNCATE TABLE **t** CASCADE;
DELETE FROM **t**;
Remove all data in the table *t*

Modifying data

INSERT INTO **t(c1, c2)** VALUES**(v1, v2)**;
Insert one row into a table

INSERT INTO **t(c1, c2)** VALUES
(v1, v2), (v1, v2), (v1, v2), (v1, v2), (v1, v2);
Insert multiple rows into a table (5 in the example above)

UPDATE **t** SET **c1=v1**, **c2=v2** WHERE **c3=v3**;
Update values in the columns c1, c2 that match the condition

INSERT INTO **t(c1, c2)** SELECT **c3**, **c4** FROM **t2**;
Insert rows from columns c3 and c4 from t2
into columns c1 and c2 in t1 respectively

Notes

t, **t1**, **t2** ... **tn** – table names
c, **c1**, **c2** ... **cn** – column names
v, **v1**, **v2** ... **vn** – values of different type
value – another values
[!] – optional negative condition
[ALL] – optional all condition

Retrieving data

SELECT * FROM **t**;
A simple query to retrieve all rows and columns from *t*

SELECT **c1, c2** FROM **t**;
Query data in columns *c1*, *c2* from a table

SELECT **c1, c2** FROM **t** ORDER BY **c1**;
Query data ordering them by *c1*

SELECT * FROM **t** ORDER BY **c1** ASC;
SELECT * FROM **t** ORDER BY **c1** DESC;
Sort the result set in ascending or descending order

SELECT **c1, c2** FROM **t** LIMIT **number** OFFSET **offset**;
Query *the number* of rows skipping *the offset* of rows

SELECT **c1, c2** FROM **t**
WHERE **c1** BETWEEN **low** AND **high**;
Query rows where value in *c1* is between two values

SELECT **c1, c2** FROM **t1** WHERE **c1** [!] LIKE **pattern**;
Query rows using simple pattern matching % and _ that is processed quickly

SELECT **c1, c2** FROM **t** WHERE c1 [!] IN **v1, v2**;
Query rows where the value of *c1* is **[not]** in the list of *v1 and v2*

SELECT **c1, c2** FROM **t** WHERE **c1** IS [!] NULL;
Check if values in c1 is NULL or not

Retrieving data from multiple tables

SELECT **c1, c2** FROM **t1** UNION [ALL]
SELECT **c1, c2** FROM **t2**;
Combine rows from two queries

SELECT **c1, c2** FROM **t1** INTERSECT
SELECT **c1, c2** FROM **t2**;
Return the intersection of two queries

SELECT **c1, c2** FROM **t1** EXCEPT
SELECT **c1, c2** FROM **t2**;
Subtract a result set from another result set

SELECT **c1, c2** FROM **t1** INNER JOIN **t2**
ON **t1.c1=t2.c3**;
Inner join t1 and t2
If t1.c1!=t2.c3 the row will not be in the result

SELECT **c1, c2** FROM **t1** LEFT JOIN **t2**
ON **t1.c1=t2.c3**;
Left join t1 and t1

SELECT **c1, c2** FROM **t1** LEFT JOIN **t2**
ON **t1.c1=t2.c3**;
Right join t1 and t2

Deleting data

DELETE FROM **t** WHERE **c1=v1**;
Delete subset of rows in the table *t*

TRUNCATE TABLE **t** CASCADE;
DELETE FROM **t**;
Remove all data in the table *t*

Modifying data

INSERT INTO **t(c1, c2)** VALUES**(v1, v2)**;
Insert one row into a table

INSERT INTO **t(c1, c2)** VALUES
(v1, v2), (v1, v2), (v1, v2), (v1, v2), (v1, v2);
Insert multiple rows into a table (5 in the example above)

UPDATE **t** SET **c1=v1**, **c2=v2** WHERE **c3=v3**;
Update values in the columns c1, c2 that match the condition

INSERT INTO **t(c1, c2)** SELECT **c3**, **c4** FROM **t2**;
Insert rows from columns c3 and c4 from t2
into columns c1 and c2 in t1 respectively

Notes

t, **t1**, **t2** ... **tn** – table names
c, **c1**, **c2** ... **cn** – column names
v, **v1**, **v2** ... **vn** – values of different type
value – another values
[!] – optional negative condition
[ALL] – optional all condition