1. Retrieve Locations of specific features

SELECT place, latitude, longitiude, depth

FROM earthquakes\_table

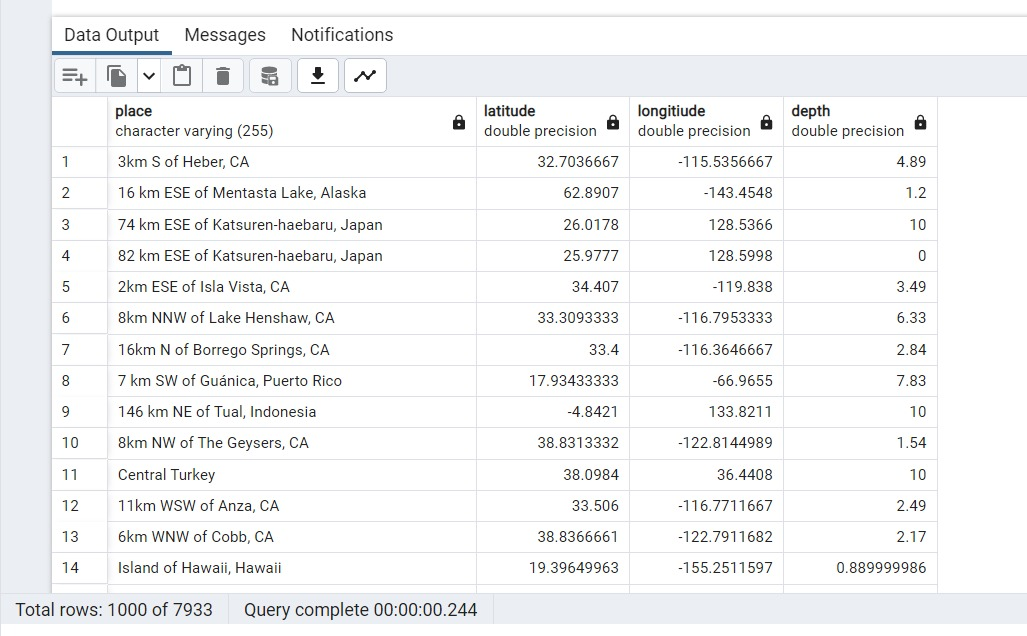
WHERE mag < 8.5 AND depth < 12;

The SQL query you provided selects the columns "place", "latitude", "longitude", and "depth" from a table called "earthquakes\_table". It filters the results based on two conditions using the WHERE clause:

1.mag < 8.5: This condition selects earthquakes with a magnitude less than 8.5.

2.depth < 12: This condition selects earthquakes with a depth less than 12 kilometers.

So the query will return all earthquakes in the table with a magnitude less than 8.5 and a depth less than 12 kilometers, along with their respective places, latitudes, longitudes, and depths.



2. Calculate Distance between points

select

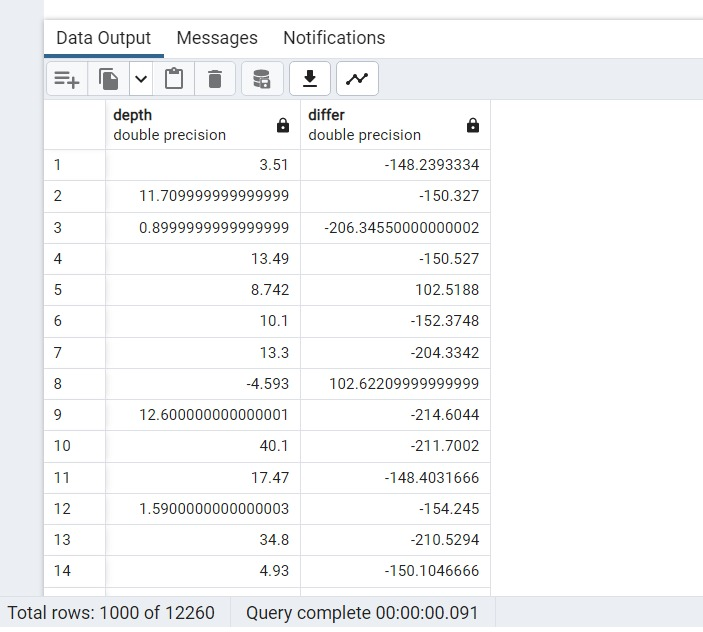
depth - deptherror as Depth,

longitiude - latitude as Differ

FROM earthquakes\_table ;

The SQL query you provided selects the calculated difference between two columns "depth - deptherror" and "longitude - latitude" from the "earthquakes\_table" table. The first expression subtracts the "deptherror" value from the "depth" value for each earthquake, and the resulting value is aliased as "Depth". The second expression subtracts the "latitude" value from the "longitude" value for each earthquake, and the resulting value is aliased as "Differ".

The query will return a table with two columns: "Depth" and "Differ". The "Depth" column will contain the calculated difference between the "depth" and "deptherror" values for each earthquake, and the "Differ" column will contain the calculated difference between the "longitude" and "latitude" values for each earthquake.

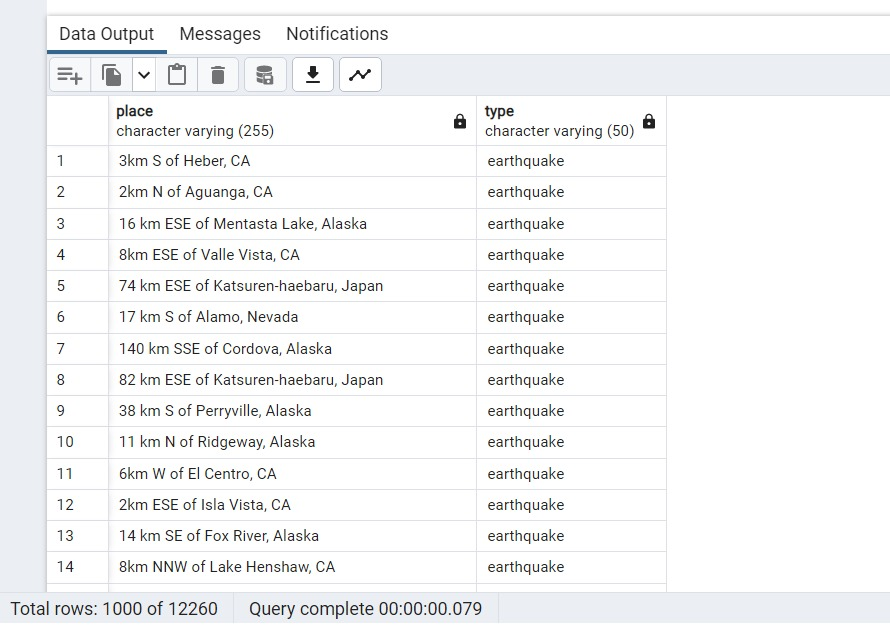


3. Calculate Areas of Interest

select place, type from public.earthquakes\_table;

The SQL query you provided selects the "place" and "type" columns from the "earthquakes\_table" table. The table is specified to be in the "public" schema, so the full table name is "public.earthquakes\_table". The "place" column likely contains the location where each earthquake occurred, and the "type" column likely contains information about the type of earthquake (e.g., tectonic, volcanic, etc.).

The query will return a table with two columns: "place" and "type". The "place" column will contain the location information for each earthquake, and the "type" column will contain the earthquake type information for each earthquake.



4. Analyze the queries

BEGIN;

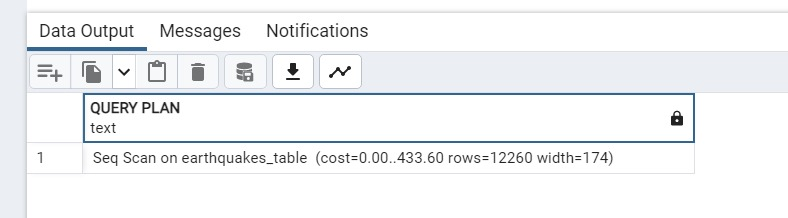
EXPLAIN

select \* from public.earthquakes\_table ;

The SQL query you provided starts a transaction with the BEGIN statement and then uses the EXPLAIN statement to get information about how the database will execute a SELECT statement that follows.

The SELECT statement itself retrieves all columns from the "earthquakes\_table" table in the "public" schema. The EXPLAIN statement provides information about how the database will execute the SELECT statement, such as which indexes will be used and the estimated cost of the operation.

So the EXPLAIN statement will return information about how the database will execute the SELECT statement, but it will not return any data from the table itself. The transaction started with the BEGIN statement is still open and must be either committed or rolled back before any changes to the database are made permanent.

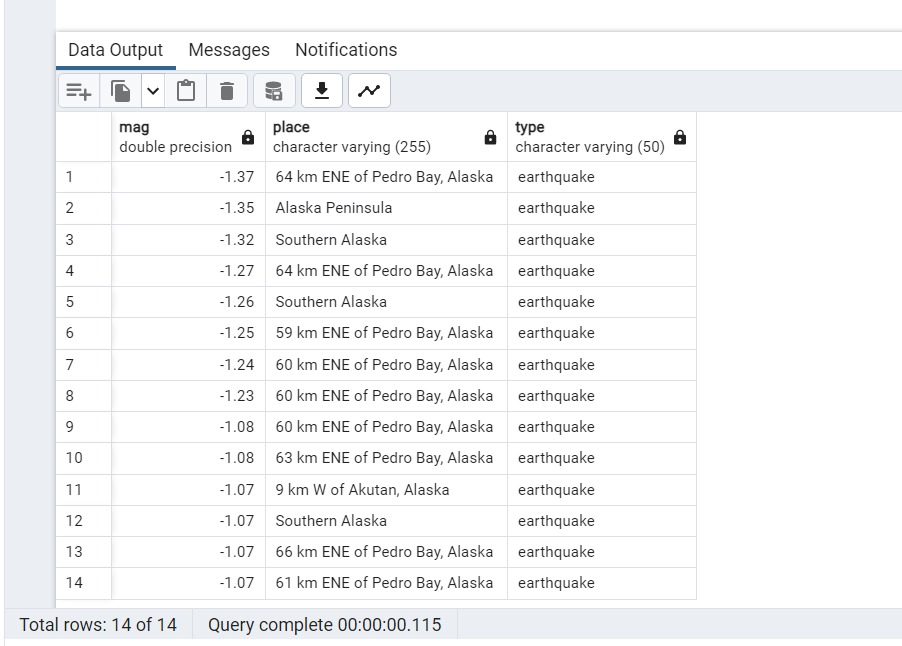


5. Sorting and Limit Executions

select mag,place, type from public.earthquakes\_table order by mag asc limit 14;

The SQL query you provided selects the "mag", "place", and "type" columns from the "earthquakes\_table" table in the "public" schema. The results are ordered by the "mag" column in ascending order (i.e., from smallest to largest magnitude), and the LIMIT clause limits the number of results returned to 14.

So the query will return the 14 earthquakes with the smallest magnitude values in the table, along with their respective places and types. The results will be sorted in ascending order based on the magnitude values.



6. Optimize the queries to speed up execution time

CREATE INDEX idx\_place ON earthquakes\_table(place);

SELECT place, COUNT(\*) as num\_earthquakes, AVG(mag) as avg\_magnitude

FROM (

SELECT \*

FROM earthquakes\_table

WHERE place IS NOT NULL

) AS filtered

GROUP BY place

HAVING COUNT(\*) = 10

ORDER BY num\_earthquakes DESC;

The SQL query you provided creates an index on the "place" column of the "earthquakes\_table" table in the "public" schema using the CREATE INDEX statement. The index is named "idx\_place".

The query then selects the "place" column, the count of earthquakes grouped by place, and the average magnitude of earthquakes grouped by place from the "earthquakes\_table" table. The subquery filters out any rows where the "place" column is NULL. The resulting dataset is then grouped by the "place" column, and only groups that have a count of 10 are selected. The results are then ordered by the count of earthquakes in each group in descending order.

Overall, the query returns a list of places where there have been exactly 10 earthquakes, along with the number of earthquakes and the average magnitude of those earthquakes. The index created at the beginning of the query can speed up the execution of this query if the "place" column is frequently used for grouping and filtering.

A screenshot of a computer

Description automatically generated with medium confidence

7. N-Optimization of queries

SELECT place, mag

FROM earthquakes\_table

WHERE latitude > 30 AND longitiude < -100

AND mag > 3.0

ORDER BY mag ASC

LIMIT 13;

The SQL query you provided selects the "place" and "mag" columns from the "earthquakes\_table" table in the "public" schema. The query filters the results using three conditions in the WHERE clause:

latitude > 30: This condition selects earthquakes with a latitude greater than 30.

longitude < -100: This condition selects earthquakes with a longitude less than -100.

mag > 3.0: This condition selects earthquakes with a magnitude greater than 3.0.

The results are then ordered by magnitude in ascending order and the LIMIT clause limits the number of results returned to 13.

So the query will return the 13 earthquakes that occurred in locations with a latitude greater than 30 and a longitude less than -100, and have a magnitude greater than 3.0. The results will be sorted in ascending order based on the magnitude values. The "place" column will contain the location information for each earthquake, and the "mag" column will contain the magnitude values for each earthquake.

