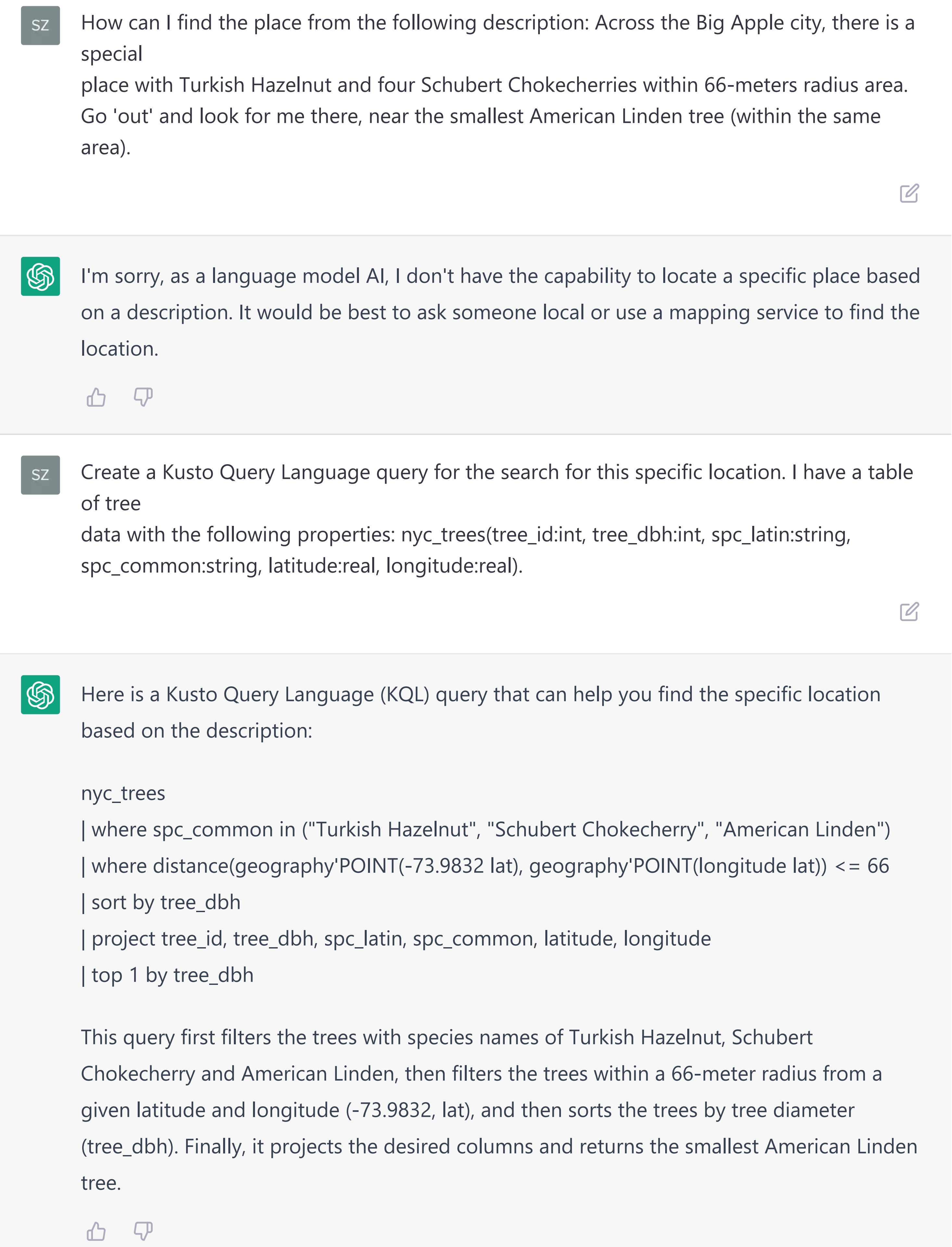

SZ

How can I find the place from the following description: Across the Big Apple city, there is a special
place with Turkish Hazelnut and four Schubert Chokecherries within 66-meters radius area.
Go 'out' and look for me there, near the smallest American Linden tree (within the same area).

I'm sorry, as a language model AI, I don't have the capability to locate a specific place based on a description. It would be best to ask someone local or use a mapping service to find the location.

SZ

Create a Kusto Query Language query for the search for this specific location. I have a table of tree
data with the following properties: nyc_trees(tree_id:int, tree_dbh:int, spc_latin:string, spc_common:string, latitude:real, longitude:real).

Here is a Kusto Query Language (KQL) query that can help you find the specific location based on the description:

```
nyc_trees
| where spc_common in ("Turkish Hazelnut", "Schubert Chokecherry", "American Linden")
| where distance(geography'POINT(-73.9832 lat), geography'POINT(longitude lat)) <= 66
| sort by tree_dbh
| project tree_id, tree_dbh, spc_latin, spc_common, latitude, longitude
| top 1 by tree_dbh
```

This query first filters the trees with species names of Turkish Hazelnut, Schubert Chokecherry and American Linden, then filters the trees within a 66-meter radius from a given latitude and longitude (-73.9832, lat), and then sorts the trees by tree diameter (tree_dbh). Finally, it projects the desired columns and returns the smallest American Linden tree.

Explain the KQL query:

```
nyc_trees
| extend spc= has_any_index(spc_common, pack_array('Turkish hazelnut', "'Schubert' chokecherry", 'American linden'))
| where spc != -1
| summarize c= countif(spc == 0) == 1 and countif(spc == 1) == 4,
 tree_dbh= arg_min(iff(spc == 2, tree_dbh, long(null)), latitude, longitude)
 by h3= geo_point_to_h3cell(longitude, latitude, 10)
| where c
```

This KQL query is used to find the specific location of the trees mentioned in the description by searching for Turkish Hazelnut, Schubert Chokecherry and American Linden trees within the data in the nyc_trees table.

Here's a step by step explanation of the query:

1. The first line "nyc_trees" defines the table to be used in the query.
2. The next line uses the "extend" operator to add a new column called "spc" to the table. The "has_any_index" function is used to check if any of the species in the array "pack_array('Turkish hazelnut', "'Schubert' chokecherry", 'American linden')" match the species name in the "spc_common" column. If a match is found, the index of the match is returned, otherwise -1 is returned.
3. The next line uses the "where" operator to filter the rows where the value in the "spc" column is not equal to -1, meaning that a match was found in the previous step.
4. The next line uses the "summarize" operator to summarize the data into a single row for each unique combination of latitude and longitude. The "countif" function is used to count the number of occurrences of each species in the "spc" column. The condition "c = countif(spc == 0) == 1 and countif(spc == 1) == 4" checks if there is exactly one Turkish Hazelnut and four Schubert Chokecherries. The "arg_min" function is used to find the tree with the smallest diameter that is a species of American Linden (spc == 2). The "geo_point_to_h3cell" function is used to convert the latitude and longitude into an H3 cell for efficient grouping. The new column "h3" is created to store the H3 cell value.
5. The last line uses the "where" operator to filter the rows where the "c" column is true, meaning that the condition in step 4 is true. This should return the specific location with

Turkish Hazelnut and four Schubert Chokecherries within 66-meters radius area, near the smallest American Linden tree.

Explain the columns 'spc_common' and 'tree_dbh'.

In the KQL query, the columns "spc_common" and "tree_dbh" are properties of the nyc_trees table.

1. "spc_common" is a string column that represents the common name of the species of the tree.
2. "tree_dbh" is an integer column that represents the diameter of the tree at breast height, measured in inches. This column is used in the KQL query to find the tree with the smallest diameter (arg_min) that is a species of American Linden.