cases_spaces

```
SELECT DISTINCT street,
    -- Trim off unwanted characters from street
    TRIM(street, '0123456789 #/.') AS cleaned_street
 FROM evanston311
ORDER BY street;
```

Explanation:

- This SQL query selects distinct street names from the evanston311 table. It then uses the TRIM function to remove leading and trailing characters (digits 0-9, space, #, ., /) from each street name, creating a new column called cleaned_street. Finally, it orders the results alphabetically by the original street name.

```
-- Count rows where description includes 'trash' or 'garbage' regardless of case
SELECT COUNT(description)
 FROM evanston311
WHERE description ILIKE '%trash%' OR description ILIKE '%garbage%';
```

Explanation:

- This SQL query counts the number of rows in the evanston311 table where the description column contains either “trash” or “garbage”, case-insensitively. ILIKE is used for case-insensitive pattern matching, and % acts as a wildcard matching any sequence of characters. The OR condition ensures that rows containing either term are included in the count.

```
-- Select categories containing Trash or Garbage
SELECT category
 FROM evanston311
 -- Use LIKE
WHERE category LIKE '%Trash%' OR category LIKE '%Garbage%';
```

Explanation:

- This SQL query selects the category column from the evanston311 table. It filters the results to include only categories that contain either “Trash” or “Garbage” using the LIKE operator with wildcards (%). The OR condition ensures that rows containing either string are included in the output.

```
-- Count rows
SELECT COUNT(description)
 FROM evanston311
```

```
-- description contains trash or garbage (any case)
WHERE (description ILIKE '%trash%'
  OR description ILIKE '%garbage%')
-- category does not contain Trash or Garbage
 AND category NOT LIKE '%Trash%'
 AND category NOT LIKE '%Garbage%';
```

Explanation:

- This SQL query counts the number of rows in the evanston311 table where the description column contains either “trash” or “garbage” (case-insensitive), but the category column does *not* contain “Trash” or “Garbage”. This likely aims to identify instances where the description indicates a trash/garbage issue, but it’s not properly categorized.

```
-- Count rows with each category
SELECT category, COUNT(*)
 FROM evanston311
WHERE (description ILIKE '%trash%'
  OR description ILIKE '%garbage%')
 AND category NOT LIKE '%Trash%'
 AND category NOT LIKE '%Garbage%'
-- What are you counting?
GROUP BY category
ORDER BY count DESC
LIMIT 10;
```

Explanation:

- This SQL query counts the occurrences of different categories in the evanston311 table, focusing on entries related to trash or garbage. It filters the data to include rows where the description contains “trash” or “garbage” but excludes rows where the category itself already contains “Trash” or “Garbage”. The results are grouped by category, ordered by count in descending order, and limited to the top 10 most frequent categories. Essentially, it’s identifying categories that are *not* primarily about trash/garbage but are frequently associated with them based on the description.