Data Scientist Role Play: Profiling and Analyzing the Yelp Dataset Coursera Worksheet

This is a 2-part assignment. In the first part, you are asked a series of questions that will help you profile and understand the data just like a data scientist would. For this first part of the assignment, you will be assessed both on the correctness of your findings, as well as the code you used to arrive at your answer. You will be graded on how easy your code is to read, so remember to use proper formatting and comments where necessary.

In the second part of the assignment, you are asked to come up with your own inferences and analysis of the data for a particular research question you want to answer. You will be required to prepare the dataset for the analysis you choose to do. As with the first part, you will be graded, in part, on how easy your code is to read, so use proper formatting and comments to illustrate and communicate your intent as required.

For both parts of this assignment, use this "worksheet." It provides all the questions you are being asked, and your job will be to transfer your answers and SQL coding where indicated into this worksheet so that your peers can review your work. You should be able to use any Text Editor (Windows Notepad, Apple TextEdit, Notepad ++, Sublime Text, etc.) to copy and paste your answers. If you are going to use Word or some other page layout application, just be careful to make sure your answers and code are lined appropriately.
In this case, you may want to save as a PDF to ensure your formatting remains intact for you reviewer.

Part 1: Yelp Dataset Profiling and Understanding

1. Profile the data by finding the total number of records for each of the tables below:

i. Attribute table = 10000
ii. Business table = 10000
iii. Category table = 10000
iv. Checkin table = 10000
v. elite_years table = 10000
vi. friend table = 10000
vii. hours table = 10000
viii. photo table = 10000
ix. review table = 10000
x. tip table = 10000
xi. user table = 10000

2. Find the total distinct records by either the foreign key or primary key for each table. If two foreign keys are listed in the table, please specify which foreign key.

i. Business = 10000 for primary key
ii. Hours = 1562
iii. Category = 2643
iv. Attribute = 1115
v. Review = 8090 for businessID, 9581 for userID, 10000 for primary key
vi. Checkin = 493
vii. Photo = 6493 for businessID, 10000 for primary key
viii. Tip = 537 for userID, 3979 for businessID
ix. User = 10000 for primary key
x. Friend = 11
xi. Elite_years = 2780

Note: Primary Keys are denoted in the ER-Diagram with a yellow key icon.

3. Are there any columns with null values in the Users table? Indicate "yes," or "no."

```
SELECT count(*)
FROM user
WHERE (
    name is NULL or review_count is NULL or yelping_since is NULL or useful
    is NULL or funny is NULL or cool is NULL or fans is NULL or average_stars
    is NULL or compliment_hot is NULL or compliment_more is NULL or
    compliment_profile is NULL or compliment_cute is NULL or
    compliment_list is NULL or compliment_note is NULL or
    compliment_plain is NULL or compliment_cool is NULL or
    compliment_funny is NULL or compliment_writer is NULL or
    compliment_photos is NULL
    )
```

4. For each table and column listed below, display the smallest (minimum), largest (maximum), and average (mean) value for the following fields:

i. Table: Review, Column: Stars
min: 1 max:5 avg: 3.7082

ii. Table: Business, Column: Stars
min: 1 max:5 avg: 3.6549

iii. Table: Tip, Column: Likes
min: 0 max:2 avg: 0.0144

iv. Table: Checkin, Column: Count

min: 1 max:53 avg: 1.9414

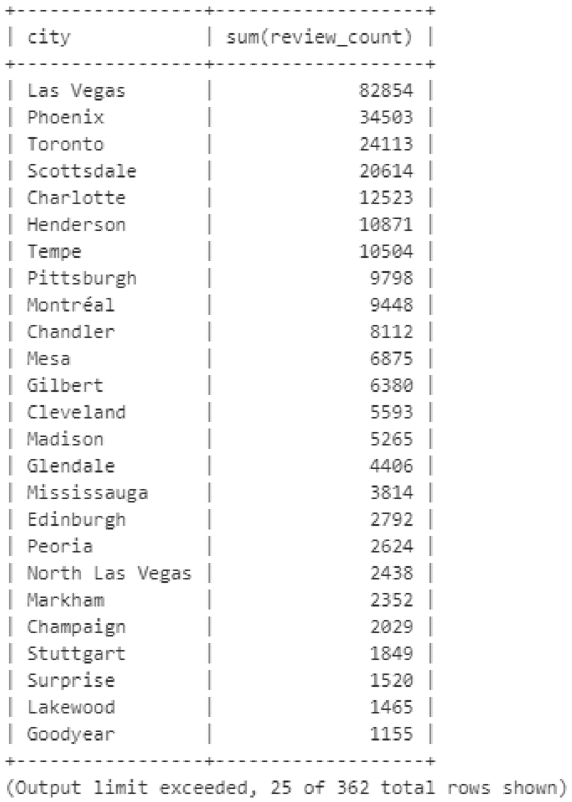
v. Table: User, Column: Review_count
min: 0 max:2000 avg: 24.2995

5. List the cities with the most reviews in descending order:

SQL code used to arrive at answer:

```
SELECT city, sum(review_count)
FROM business b
GROUP BY city
ORDER BY sum(review_count) desc
```

Copy and Paste the Result Below:



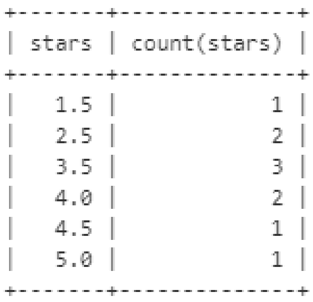
```
+-----------------+-------------------+
| city            | sum(review_count) |
+-----------------+-------------------+
| Las Vegas       |             82854 |
| Phoenix         |             34503 |
| Toronto         |             24113 |
| Scottsdale      |             20614 |
| Charlotte       |             12523 |
| Henderson       |             10871 |
| Tempe           |             10504 |
| Pittsburgh      |              9798 |
| Montréal        |              9448 |
| Chandler        |              8112 |
| Mesa            |              6875 |
| Gilbert         |              6380 |
| Cleveland       |              5593 |
| Madison         |              5265 |
| Glendale        |              4406 |
| Mississauga     |              3814 |
| Edinburgh       |              2792 |
| Peoria          |              2624 |
| North Las Vegas |              2438 |
| Markham         |              2352 |
| Champaign       |              2029 |
| Stuttgart       |              1849 |
| Surprise        |              1520 |
| Lakewood        |              1465 |
| Goodyear        |              1155 |
+-----------------+-------------------+
(Output limit exceeded, 25 of 362 total rows shown)
```

6. Find the distribution of star ratings to the business in the following cities:
i. Avon

SQL code used to arrive at answer:

```
SELECT stars, count(stars)
FROM business
WHERE city = 'Avon'
GROUP BY stars
```

Copy and Paste the Resulting Table Below (2 columns – star rating and count):



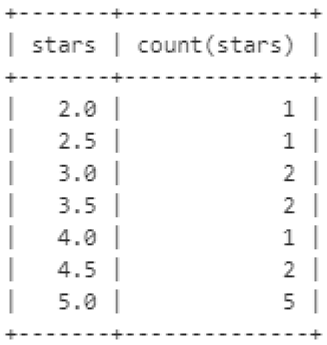
```
+-------+--------------+
| stars | count(stars) |
+-------+--------------+
|   1.5 |            1 |
|   2.5 |            2 |
|   3.5 |            3 |
|   4.0 |            2 |
|   4.5 |            1 |
|   5.0 |            1 |
+-------+--------------+
```

ii. Beachwood

SQL code used to arrive at answer:

```
SELECT stars, count(stars)
FROM business
WHERE city = 'Beachwood'
GROUP BY stars
```

Copy and Paste the Resulting Table Below (2 columns – star rating and count):



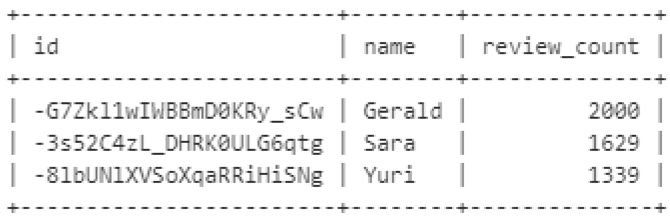
```
+-------+--------------+
| stars | count(stars) |
+-------+--------------+
|   2.0 |            1 |
|   2.5 |            1 |
|   3.0 |            2 |
|   3.5 |            2 |
|   4.0 |            1 |
|   4.5 |            2 |
|   5.0 |            5 |
+-------+--------------+
```

7. Find the top 3 users based on their total number of reviews:

SQL code used to arrive at answer:

```
SELECT id, name, review_count
FROM user
GROUP BY id
ORDER BY review_count desc
LIMIT 3
```

Copy and Paste the Result Below:



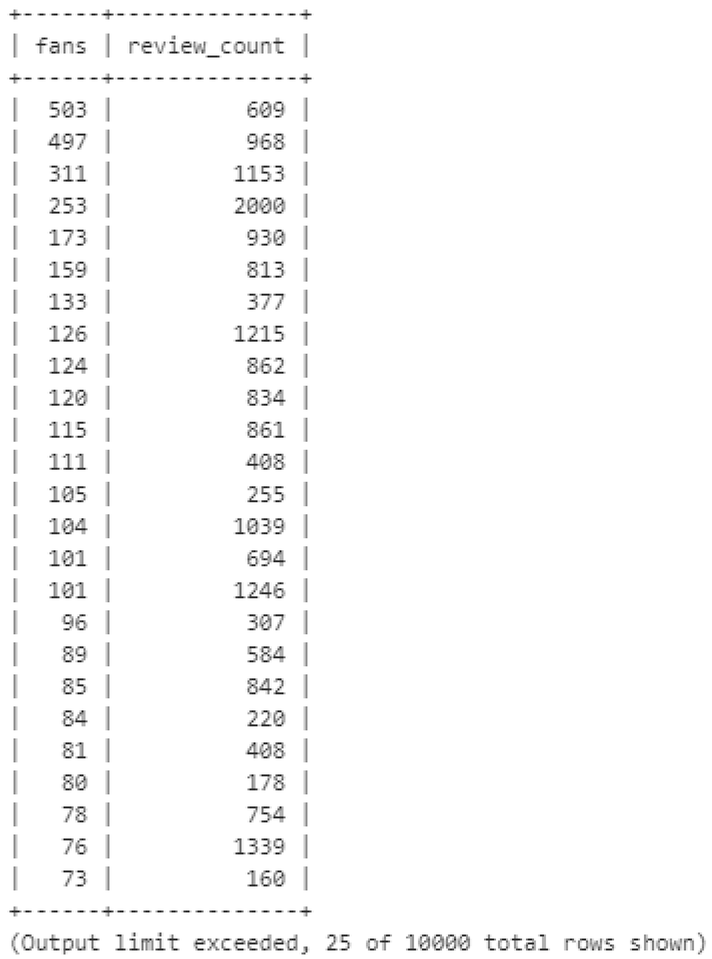
```
+------------------------+--------+--------------+
| id                     | name   | review_count |
+------------------------+--------+--------------+
| -G7Zkl1wIWBBmD0KRy_sCw | Gerald |         2000 |
| -3s52C4zL_DHRK0ULG6qtg | Sara   |         1629 |
| -8lbUNlXVSoXqaRRiHiSNg | Yuri   |         1339 |
+------------------------+--------+--------------+
```

8. Does posing more reviews correlate with more fans? Please explain your findings and interpretation of the results:

SQL code and result

```
SELECT fans, review_count
FROM user
ORDER BY fans desc, review_count
```

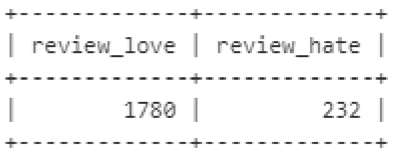
By checking the number of reviews and fans in descending order of fans, we can see that the records which have more fans, also have more reviews.



```
+------+--------------+
| fans | review_count |
+------+--------------+
|  503 |          609 |
|  497 |          968 |
|  311 |         1153 |
|  253 |         2000 |
|  173 |          930 |
|  159 |          813 |
|  133 |          377 |
|  126 |         1215 |
|  124 |          862 |
|  120 |          834 |
|  115 |          861 |
|  111 |          408 |
|  105 |          255 |
|  104 |         1039 |
|  101 |          694 |
|  101 |         1246 |
|   96 |          307 |
|   89 |          584 |
|   85 |          842 |
|   84 |          220 |
|   81 |          408 |
|   80 |          178 |
|   78 |          754 |
|   76 |         1339 |
|   73 |          160 |
+------+--------------+
(Output limit exceeded, 25 of 10000 total rows shown)
```

9. Are there more reviews with the word "love" or with the word "hate" in them?

Answer:



```
+-------------+-------------+
| review_love | review_hate |
+-------------+-------------+
|        1780 |         232 |
+-------------+-------------+
```

There are more reviews with the word "love" (1780) than "hate" (232).

SQL code used to arrive at answer:

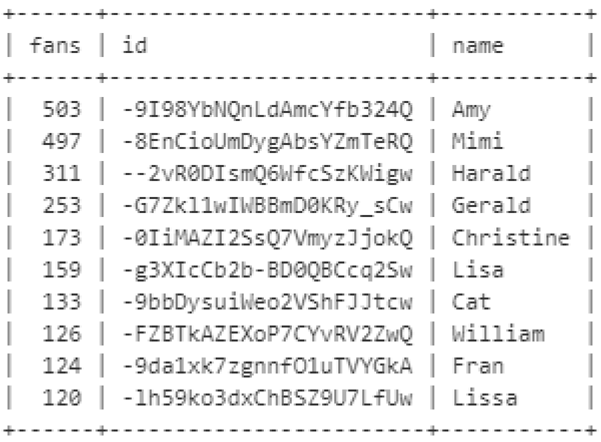
```
SELECT count(id) as review_love,
      (SELECT count(id)
      FROM review
      WHERE text like '%hate%') as review_hate
FROM review
WHERE text like '%love%'
```

10. Find the top 10 users with the most fans:

SQL code used to arrive at answer:

```
SELECT fans, id, name
FROM user
ORDER BY fans desc
LIMIT 10
```

Copy and Paste the Result Below:



```
+------+------------------------+-----------+
| fans | id                     | name      |
+------+------------------------+-----------+
|  503 | -9I98YbNQnLdAmcYfb324Q | Amy       |
|  497 | -8EnCioUmDygAbsYZmTeRQ | Mimi      |
|  311 | --2vR0DIsmQ6WfcSzKWigw | Harald    |
|  253 | -G7Zkl1wIWBBmD0KRy_sCw | Gerald    |
|  173 | -0IiMAZI2SsQ7VmyzJjokQ | Christine |
|  159 | -g3XIcCb2b-BD0QBCcq2Sw | Lisa      |
|  133 | -9bbDysuiWeo2VShFJJtcw | Cat       |
|  126 | -FZBTkAZEXoP7CYvRV2ZwQ | William   |
|  124 | -9da1xk7zgnnfO1uTVYGkA | Fran      |
|  120 | -1h59ko3dxChBSZ9U7LfUw | Lissa     |
+------+------------------------+-----------+
```

Part 2: Inferences and Analysis

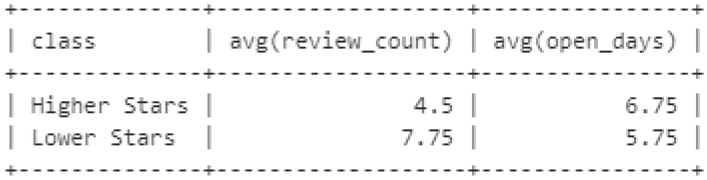
1. Pick one city and category of your choice and group the businesses in that city or category by their overall star rating. Compare the businesses with 2-3 stars to the businesses with 4-5 stars and answer the following questions. Include your code.

**(If we filter the businesses with BOTH city and category, then less than 10 records are left and not appropriate for analysis, so only city 'Charlotte' is chosen below)

(If we analyze the hours COLUMN only)

```
SELECT class, avg(review_count), avg(open_days)
FROM
--Find out how many days are opened per week
(SELECT name, class, review_count, count(day) as open_days
FROM
   (SELECT name, hours, review_count,
                   CASE
                   WHEN hours like 'Mon%' THEN 1
                   WHEN hours like 'Tue%' THEN 2
                   WHEN hours like 'Wed%' THEN 3
                   WHEN hours like 'Thur%' THEN 4
                   WHEN hours like 'Fri%' THEN 5
                   WHEN hours like 'Sat%' THEN 6
                   ELSE 7
                   END as day,
                   CASE
                   WHEN stars between 2 and 3.5 THEN 'Lower Stars'
                   WHEN stars between 4 and 5 THEN 'Higher Stars'
                   END as class
   FROM hours h INNER JOIN BUSINESS b ON h.business_id = b.id
   WHERE CITY = 'Charlotte')
GROUP BY name)
GROUP BY class
```

Result:



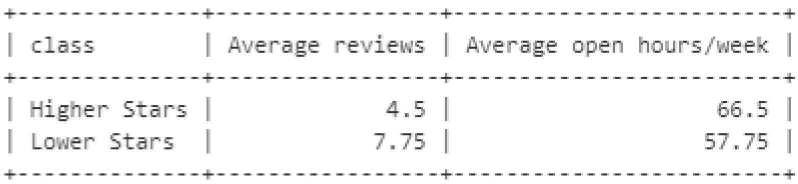
```
+--------------+-------------------+-----------------+
| class        | avg(review_count) | avg(open_days) |
+--------------+-------------------+-----------------+
| Higher Stars |               4.5 |            6.75 |
| Lower Stars  |              7.75 |            5.75 |
+--------------+-------------------+-----------------+
```

(If we analyze the EXACT OPEN HOURS PER WEEK)

```
SELECT class, avg(review_count) as 'Average reviews', avg(totalopenhours) as
'Average open hours/week'
FROM
--Get the total opening hours per week
  (Select id, name, city, hours, class, review_count, sum(open_hours) as
  totalopenhours
  FROM
--Get the opening hours
    (Select id, name, city, hours, class, review_count,
    strftime('%H:%M', close_time) - strftime('%H:%M', open_time) as
    open_hours
    FROM
--Getting the opening and closing time
        (SELECT id, name, city, hours, class, review_count,
        CASE
        WHEN length(hourwithoutweek) = 10
        THEN "0"||substr(hourwithoutweek,1,4)
        ELSE substr(hourwithoutweek,1,5)
        END as open_time,

        CASE
        WHEN length(hourwithoutweek) = 10
        THEN substr(hourwithoutweek,6,5)
        ELSE substr(hourwithoutweek,7,5)
        END as close_time
        FROM
--Trim the week days in hours column, group them by stars
            (SELECT id, name, city, hours,
            CASE
            WHEN stars between 2 and 3.5 THEN 'Lower Stars'
            WHEN stars between 4 and 5 THEN 'Higher Stars'
            END as class,
            review_count, trim(hours,
            'MonTuesWednesThursFriSaturSunday|') as hourwithoutweek
            FROM hours INNER JOIN business on id = business_id))
            WHERE city = 'Charlotte')
            GROUP BY id)
```

Result:



```
+--------------+-----------------+------------------------+
| class        | Average reviews | Average open hours/week |
+--------------+-----------------+------------------------+
| Higher Stars |             4.5 |                   66.5 |
| Lower Stars  |            7.75 |                  57.75 |
+--------------+-----------------+------------------------+
```

i. Do the two groups you chose to analyze have a different distribution of hours?

From the result, we can find that the businesses with lower stars have shorter open hours per week(57.75/5.75), while those with higher stars have longer open hours(66.5/6.75). Therefore, in Charlotte, longer opening hours would lead to better star ranking.

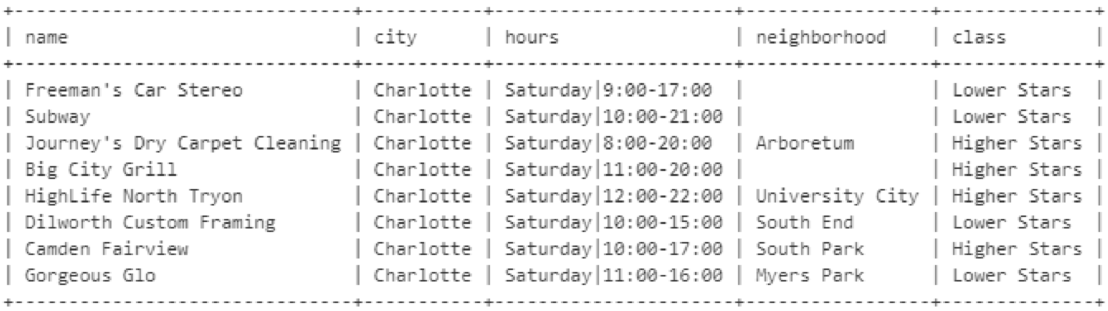
ii. Do the two groups you chose to analyze have a different number of reviews?

From the result, we can see that the average number of reviews the businesses with 2-3 stars get are 7.75, and those with 4-5 stars are 4.5. Therefore, in Charlotte, fewer reviews seem to lead to better star ranking. As the difference is not quite a lot, so it might not really matter.

iii. Are you able to infer anything from the location data provided between these two groups? Explain.

```
SELECT name, city, hours, neighborhood ,
      CASE
      WHEN stars between 2 and 3.5 THEN 'Lower Stars'
      WHEN stars between 4 and 5 THEN 'Higher Stars'
      END as class
FROM hours INNER JOIN business on id = business_id
WHERE city = 'Charlotte'
GROUP BY id
```

Result:



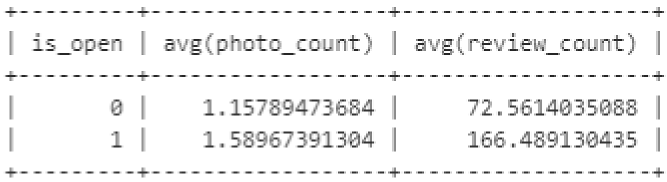
```
+-----------------------------+-----------+----------------------+-----------------+--------------+
| name                        | city      | hours                | neighborhood    | class        |
+-----------------------------+-----------+----------------------+-----------------+--------------+
| Freeman's Car Stereo        | Charlotte | Saturday|9:00-17:00  |                 | Lower Stars  |
| Subway                      | Charlotte | Saturday|10:00-21:00 |                 | Lower Stars  |
| Journey's Dry Carpet Cleaning | Charlotte | Saturday|8:00-20:00  | Arboretum       | Higher Stars |
| Big City Grill              | Charlotte | Saturday|11:00-20:00 |                 | Higher Stars |
| HighLife North Tryon        | Charlotte | Saturday|12:00-22:00 | University City | Higher Stars |
| Dilworth Custom Framing     | Charlotte | Saturday|10:00-15:00 | South End       | Lower Stars  |
| Camden Fairview             | Charlotte | Saturday|10:00-17:00 | South Park      | Higher Stars |
| Gorgeous Glo                | Charlotte | Saturday|11:00-16:00 | Myers Park      | Lower Stars  |
+-----------------------------+-----------+----------------------+-----------------+--------------+
```

As some records in the neighborhood column is null, and the others are different, so it seems that location data is not relevant to stars.

2. Group business based on the ones that are open and the ones that are closed. What differences can you find between the ones that are still open and the ones that are closed? List at least two differences and the SQL code you used to arrive at your answer.

```
SELECT is_open, avg(photo_count), avg(review_count)
FROM
   (SELECT b.id, is_open, count(p.id) as photo_count, review_count
   FROM business b INNER JOIN photo p ON b.id = p.business_id
   GROUP BY b.id)
GROUP BY is_open
```

Result:



```
+---------+-----------------+------------------+
| is_open | avg(photo_count) | avg(review_count) |
+---------+-----------------+------------------+
|       0 |   1.15789473684 |    72.5614035088 |
|       1 |   1.58967391304 |    166.489130435 |
+---------+-----------------+------------------+
```

i. Difference 1:

For the businesses which have photos, those are opened have more average photos (1.59) than those are closed (1.16).

ii. Difference 2:

Obviously those are opened have more reviews (166.49) than those are closed (72.56)

3. For this last part of your analysis, you are going to choose the type of analysis you want to conduct on the Yelp dataset and are going to prepare the data for analysis.

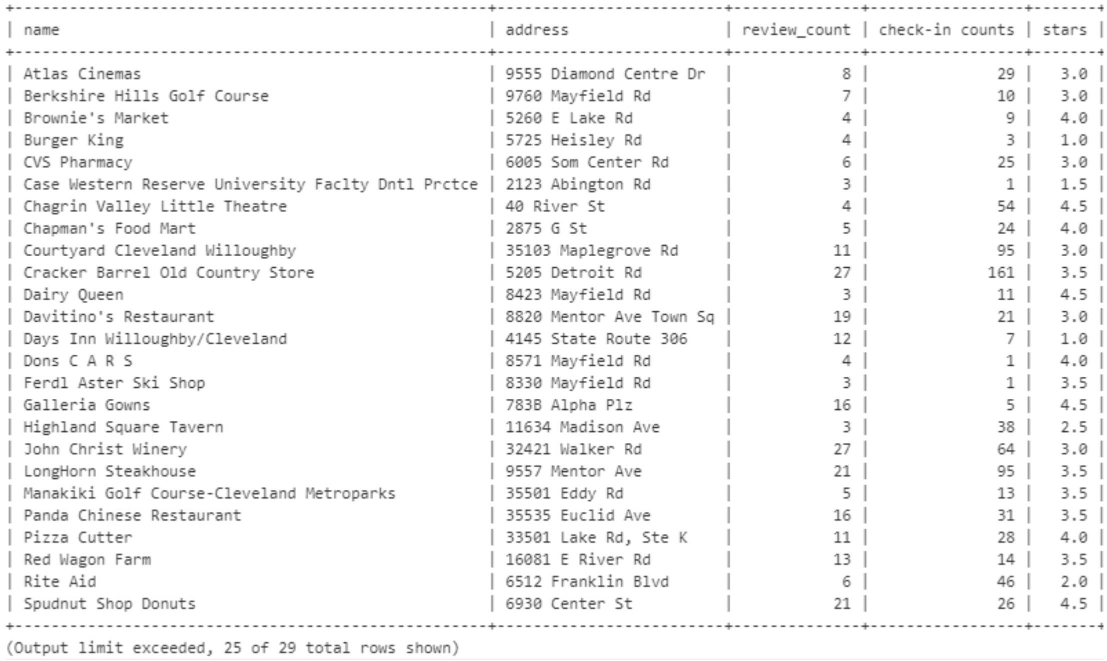
i. Indicate the type of analysis you chose to do:

I'm going to conduct an analysis about predicting how many check-in a business will have, as check-in usually indicate the consuming behavior, and also an important marketing behavior, therefore I believe that predicting how many check-in they will have is a useful analysis.

ii. Write 1-2 brief paragraphs on the type of data you will need for your analysis and why you chose that data:

To conduct the analysis, location data is necessary and directly related. At the same time, stars is usually treat as the quality of the business, so it might help attracting user to check-in.
Also, how many reviews they get might also related to the check-in behavior.

iii. Output of your finished dataset:



```
+----------------------------------------------------+-------------------------+--------------+-----------------+-------+
| name                                               | address                 | review_count | check-in counts | stars |
+----------------------------------------------------+-------------------------+--------------+-----------------+-------+
| Atlas Cinemas                                      | 9555 Diamond Centre Dr  |            8 |              29 |   3.0 |
| Berkshire Hills Golf Course                        | 9760 Mayfield Rd        |            7 |              10 |   3.0 |
| Brownie's Market                                   | 5260 E Lake Rd          |            4 |               9 |   4.0 |
| Burger King                                        | 5725 Heisley Rd         |            4 |               3 |   1.0 |
| CVS Pharmacy                                       | 6005 Som Center Rd      |            6 |              25 |   3.0 |
| Case Western Reserve University Faclty Dntl Prctce | 2123 Abington Rd        |            3 |               1 |   1.5 |
| Chagrin Valley Little Theatre                      | 40 River St             |            4 |              54 |   4.5 |
| Chapman's Food Mart                                | 2875 G St               |            5 |              24 |   4.0 |
| Courtyard Cleveland Willoughby                     | 35103 Maplegrove Rd     |           11 |              95 |   3.0 |
| Cracker Barrel Old Country Store                   | 5205 Detroit Rd         |           27 |             161 |   3.5 |
| Dairy Queen                                        | 8423 Mayfield Rd        |            3 |              11 |   4.5 |
| Davitino's Restaurant                              | 8820 Mentor Ave Town Sq |           19 |              21 |   3.0 |
| Days Inn Willoughby/Cleveland                      | 4145 State Route 306    |           12 |               7 |   1.0 |
| Dons C A R S                                       | 8571 Mayfield Rd        |            4 |               1 |   4.0 |
| Ferdl Aster Ski Shop                               | 8330 Mayfield Rd        |            3 |               1 |   3.5 |
| Galleria Gowns                                     | 783B Alpha Plz          |           16 |               5 |   4.5 |
| Highland Square Tavern                             | 11634 Madison Ave       |            3 |              38 |   2.5 |
| John Christ Winery                                 | 32421 Walker Rd         |           27 |              64 |   3.0 |
| LongHorn Steakhouse                                | 9557 Mentor Ave         |           21 |              95 |   3.5 |
| Manakiki Golf Course-Cleveland Metroparks          | 35501 Eddy Rd           |            5 |              13 |   3.5 |
| Panda Chinese Restaurant                           | 35535 Euclid Ave        |           16 |              31 |   3.5 |
| Pizza Cutter                                       | 33501 Lake Rd, Ste K    |           11 |              28 |   4.0 |
| Red Wagon Farm                                     | 16081 E River Rd        |           13 |              14 |   3.5 |
| Rite Aid                                           | 6512 Franklin Blvd      |            6 |              46 |   2.0 |
| Spudnut Shop Donuts                                | 6930 Center St          |           21 |              26 |   4.5 |
+----------------------------------------------------+-------------------------+--------------+-----------------+-------+
(Output limit exceeded, 25 of 29 total rows shown)
```

iv. Provide the SQL code you used to create your final dataset:

```
SELECT name, address, review_count, sum(count) as 'check-in counts', stars
FROM business b INNER JOIN checkin c ON c.business_id = b.id
GROUP BY name
```