YELP DATASET ANALYSIS

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

SOLUTON: -

Sample code (including NULL values):

```
select count(*) as
total_records
from attribute;
```

```
+---------------+
| total_records |
+---------------+
|         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

ii. Hours = 1562

iii. Category = 2643

iv. Attribute = 1115

v. Review = (**id** = 10000, **business_id** = 8090, **user_id** = 9581)

vi. Checkin = 493

vii. Photo = (**business_id** = 6493, **id** = 10000)

viii. Tip = (**business_id** = 3979, **id** = 537)

ix. User = 10000

x. Friend = 11

xi. Elite_years = 2780

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

SOLUTION: -

i. SELECT COUNT (distinct id) from business

ii. SELECT COUNT (distinct business_id) from hours

iii. SELECT COUNT (distinct business_id) from Category

iv. SELECT COUNT (distinct business_id) from Attribute

v. SELECT COUNT (distinct id), count(distinct business_id), COUNT(distinct user_id) from review

vi. SELECT COUNT (distinct business_id) from Checkin

vii. SELECT COUNT (distinct id), count(distinct business_id) from photo

viii. SELECT COUNT (distinct business_id),count (distinct user_id) from tip

ix. SELECT COUNT (distinct id) from User

SELECT COUNT (distinct user_id) from Friend

SELECT COUNT (distinct user_id) from Elite_yers

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

Answer:

No

SQL code used to arrive at answer:

```
SELECT COUNT(* ) from User
where id is NULL OR
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:

SOLUTION: -

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

QUERIES FOR ABOVE ANSWERS: -

i. SELECT min(stars),max(stars), avg(stars) from Review

ii. SELECT min(stars),max(stars), avg(stars) from Business

iii. SELECT min(likes),max(likes), avg(likes) from Tip

iv. SELECT min(count),max(count), avg(count) from Checkin

v. SELECT min(Review_count),max(Review_count), avg(Review_count) from User

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

SQL code used to arrive at answer:

```
SELECT city, SUM(review_count) AS 'Most_Reviews'
FROM business
GROUP BY city
ORDER BY Most_Reviews DESC;
```

Copy and Paste the Result Below:

city	Most_Reviews
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, sum(review_count) FROM business

WHERE city = 'Avon'

GROUP BY stars

Copy and Paste the Resulting Table Below (2 columns â€“ star rating and count):

stars	sum(review_count)
1.5	10
2.5	6
3.5	88
4.0	21
4.5	31
5.0	3

ii. Beachwood

SQL code used to arrive at answer:

SELECT stars, sum(review_count) FROM business

WHERE city = 'Beachwood'

GROUP BY stars

Copy and Paste the Resulting Table Below (2 columns â€“ star rating and count):

stars	sum(review_count)
2.0	8
2.5	3
3.0	11
3.5	6
4.0	69
4.5	17
5.0	23

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 AS 'Total_Number_Of_Reviews'
FROM user
ORDER BY Total_Number_Of_Reviews DESC
LIMIT 3;
```

Copy and Paste the Result Below:

id	name	Total_Number_Of_Reviews
-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:

SOLUTION: -

No, Hence posing more reviews do not correlate with more fans.

Here is my findings and interpretation that clarify that posing more reviews do not correlate with more fans.

SELECT id, name, review_count, fans, yelping_since

FROM user

ORDER BY fans desc

id	name	review_count	fans	yelping_since
-9I98YbNQnLdAmcYfb324Q	Amy	609	503	2007-07-19 00:00:00
-8EnCioUmDygAbsYZmTeRQ	Mimi	968	497	2011-03-30 00:00:00
--2vR0DIsmQ6WfcSzKWigw	Harald	1153	311	2012-11-27 00:00:00
-G7Zkl1wIWBBmD0KRy_sCw	Gerald	2000	253	2012-12-16 00:00:00
-0IiMAZI2SsQ7VmyzJjokQ	Christine	930	173	2009-07-08 00:00:00
-g3XIcCb2b-BD0QBCcq2Sw	Lisa	813	159	2009-10-05 00:00:00
-9bbDysuiWeo2VShFJJtcw	Cat	377	133	2009-02-05 00:00:00
-FZBTkAZEXoP7CYvRV2ZwQ	William	1215	126	2015-02-19 00:00:00
-9da1xk7zgnnfO1uTVYGkA	Fran	862	124	2012-04-05 00:00:00
-lh59ko3dxChBSZ9U7LfUw	Lissa	834	120	2007-08-14 00:00:00
-B-QEUESGWHPE_889WJaeg	Mark	861	115	2009-05-31 00:00:00
-DmqnhW4Omr3YhmnigaqHg	Tiffany	408	111	2008-10-28 00:00:00
-cv9PPT7IHux7XUc9dOpkg	bernice	255	105	2007-08-29 00:00:00
-DFCC64NXgqrxlO8aLU5rg	Roanna	1039	104	2006-03-28 00:00:00
-IgKkE8JvYNWeGu8ze4P8Q	Angela	694	101	2010-10-01 00:00:00
-K2Tcgh2EKX6e6HqqIrBIQ	.Hon	1246	101	2006-07-19 00:00:00
-4viTt9UC44lWCFJwleMNQ	Ben	307	96	2007-03-10 00:00:00
-3i9bhfvrM3F1wsC9XIB8g	Linda	584	89	2005-08-07 00:00:00
-kLVfaJytOJY2-QdQoCcNQ	Christina	842	85	2012-10-08 00:00:00
-ePh4Prox7ZXnEBNGKyUEA	Jessica	220	84	2009-01-12 00:00:00
-4BEUkLvHQntN6qPfKJP2w	Greg	408	81	2008-02-16 00:00:00
-C-l8EHSLXtZZVfUAUhsPA	Nieves	178	80	2013-07-08 00:00:00
-dw8f7FLaUmWR7bfJ_Yf0w	Sui	754	78	2009-09-07 00:00:00
-8lbUNlXVSoXqaRRiHiSNg	Yuri	1339	76	2008-01-03 00:00:00
-0zEEaDFIjABtPQni0XlHA	Nicole	161	73	2009-04-30 00:00:00

(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:

The word ‘Love’ has more reviews than ‘Hate’.

SQL code used to arrive at answer:

SELECT COUNT(*) AS 'HATE'

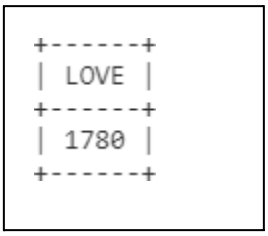
FROM review

WHERE text LIKE '%hate%';

SELECT COUNT(*) AS 'LOVE'

FROM review

WHERE text LIKE '%love%';



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

SQL code used to arrive at answer:

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

Copy and Paste the Result Below:

id	name	fans
-9I98YbNQnLdAmcYfb324Q	Amy	503
-8EnCioUmDygAbsYZmTeRQ	Mimi	497
--2vR0DIsmQ6WfcSzKWigw	Harald	311
-G7Zkl1wIWBBmD0KRy_sCw	Gerald	253
-0IiMAZI2SsQ7VmyzJjokQ	Christine	173
-g3XIcCb2b-BD0QBCcq2Sw	Lisa	159
-9bbDysuiWeo2VShFJJtcw	Cat	133
-FZBTkAZEXoP7CYvRV2ZwQ	William	126
-9da1xk7zgnnfO1uTVYGkA	Fran	124
-lh59ko3dxChBSZ9U7LfUw	Lissa	120

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.

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

YES

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

YES

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

Hence, based on the findings, we can see that there can be a connection between a company's rating and its location. Businesses with a high rating are likely to be close neighbors. Additionally, their working hours are similar. Additionally, companies with longer working hours typically receive higher ratings.

name	category	city	zipcode	hours
Charlie D's Catfish & Chicken	Restaurants	Phoenix	85034	Saturday|11:00-18:00
Bootleggers Modern American Smokehouse	Restaurants	Phoenix	85028	Saturday|11:00-22:00
Gallagher's	Restaurants	Phoenix	85024	Saturday|9:00-2:00
McDonald's	Restaurants	Phoenix	85004	Saturday|5:00-0:00

rating	reviews
4-5 stars	7
4-5 stars	431
2-3 stars	60
2-3 stars	8

SQL code used for analysis:

```
SELECT B.name, C.category, B.city, B.postal_code as zipcode, hours ,
CASE
    WHEN stars BETWEEN 2 AND 3 THEN '2-3 stars'
    WHEN stars BETWEEN 4 AND 5 THEN '4-5 stars'
END AS rating, B.review_count as reviews
From business B Inner join hours H  on B.id = H.business_id
Inner join category C on C.business_id = B.id
Where city = 'Phoenix' and category = 'Restaurants' and rating in ('2-3
stars','4-5 stars')
Group By name
Order By stars desc
```

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.

i. Difference 1:

The business that are still open have higher rating.

ii. Difference 2:

The business that are still open have more reviews and have longer working hours.

SQL code used for analysis:

SELECT b.name, c.category, b.is_open, h.hours, b.stars, b.review_count, b.postal_code

FROM business AS b INNER JOIN category AS c

ON b.id = c.business_id

INNER JOIN hours AS h

ON h.business_id = c.business_id

WHERE b.city = 'Toronto' AND b.state = 'ON'

GROUP BY b.is_open

ORDER BY b.stars

name	category	is_open	hours	stars	review_count	postal_code
99 Cent Sushi	Restaurants	0	Saturday|11:00-23:00	2.0	5	M5B 2E5
Toronto Acupuncture Studio	Acupuncture	1	Saturday|10:00-14:00	4.5	16	M6H 1B6

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.

Ideas for analysis include: Parsing out keywords and business attributes for sentiment analysis, clustering businesses to find commonalities or anomalies between them, predicting the overall star rating for a business, predicting the number of fans a user will have, and so on. These are just a few examples to get you started, so feel free to be creative and come up with your own problem you want to solve. Provide answers, in-line, to all of the following:

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

I chose to study the preference among different types of food Like Chinese, Japanese, Indian and etc on yelp database.

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

I will select various categories of cuisine such as "Chinese", "Mexican", "Korean", "French", "Italian", "Japanese", and "Indian". Afterward, I will examine the star ratings and review counts to gain insights into the popularity of these food types on Yelp.

iii. Output of your finished dataset:

category	Number_Of_Resturants	Total_Review	Star	city
Korean	2	31.5	4.25	Cuyahoga Falls
French	2	128.5	4.0	Las Vegas
Japanese	5	30.4	3.8	Las Vegas
Indian	5	12.6	3.6	Edinburgh
Italian	2	74.0	3.5	Montréal
Mexican	7	46.714285714285715	3.5	Tolleson
Chinese	4	199.0	3.125	Edinburgh

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

```
SELECT A.category, COUNT(B.name) AS
'Number_Of_Resturants', AVG(review_count) AS 'Total_Review',
AVG(stars) AS 'Star', B.city
FROM business AS B INNER JOIN category AS A
ON B.id = A.business_id
WHERE A.category IN
("Korean","Mexican","French","Italian","Chinese","Indian","Japanese")
GROUP BY A.category
ORDER BY AVG(stars) DESC
```