databricks SQL Challenge

(https://databricks.com)

```
# File location and type
file_location = "/FileStore/tables/tweets_new.csv"
file_type = "csv"

# CSV options
infer_schema = "true"
first_row_is_header = "true"
delimiter = ","

# The applied options are for CSV files. For other file types, these will be ignored.
df = spark.read.format(file_type) \
  .option("inferSchema", infer_schema) \
  .option("header", first_row_is_header) \
  .option("sep", delimiter) \
  .load(file_location)

display(df)
```

Table

	id	timestamp	text
1	0	2009-04-06T22:19:45.000+0000	@switchfoot http://twitpic.com/2y1zl - awww, that's a bummer. you shoulda got david carr of third
2	303	2009-04-06T22:19:49.000+0000	is upset that he can't update his facebook by texting it... and might cry as a result school today also
3	548	2009-04-06T22:19:53.000+0000	@kenichan i dived many times for the ball. managed to save 50% the rest go out of bounds
4	815	2009-04-06T22:19:57.000+0000	my whole body feels itchy and like its on fire
5	824	2009-04-06T22:19:57.000+0000	@nationwideclass no, it's not behaving at all. i'm mad. why am i here? because i can't see you all ov
6	1003	2009-04-06T22:20:00.000+0000	@kwesidei not the whole crew
7	1223	2009-04-06T22:20:03.000+0000	need a hug

10,000 rows | Truncated data

```
# File location and type
file_location = "/FileStore/tables/users.csv"
file_type = "csv"

# CSV options
infer_schema = "true"
first_row_is_header = "true"
delimiter = ","

# The applied options are for CSV files. For other file types, these will be ignored.
users_df = spark.read.format(file_type) \
  .option("inferSchema", infer_schema) \
  .option("header", first_row_is_header) \
  .option("sep", delimiter) \
  .load(file_location)

display(users_df)
```

Table

	id	user	age
1	0	_TheSpecialOne_	unknown
2	303	scotthamilton	unknown
3	548	mattycus	unknown
4	815	ElleCTF	unknown
5	824	Karoli	unknown
6	1003	joy_wolf	unknown
7	1223	mybirch	unknown

10,000 rows | Truncated data

```
# File location and type
file_location = "/FileStore/tables/followers_new.csv"
file_type = "csv"

# CSV options
infer_schema = "true"
first_row_is_header = "true"
delimiter = ","

# The applied options are for CSV files. For other file types, these will be ignored.
followers_df = spark.read.format(file_type) \
  .option("inferSchema", infer_schema) \
  .option("header", first_row_is_header) \
  .option("sep", delimiter) \
  .load(file_location)

display(followers_df)
```

Table

	id	following
1	210499023	587419676
2	576194305	585452007
3	603493582	591848745
4	763712420	521784660
5	209141897	85858045
6	595342589	595353768
7	784146077	326516445

10,000 rows | Truncated data

```
# Create a view or table

temp_table_name = "tweets"

df.createOrReplaceTempView(temp_table_name)

temp_table_name = "users"

users_df.createOrReplaceTempView(temp_table_name)

temp_table_name = "followers"

followers_df.createOrReplaceTempView(temp_table_name)


%sql

/* 1. Select All Usernames */

select id
from users
```

Table

	id
1	0
2	303
3	548
4	815
5	824
6	1003

7	1223

10,000 rows | Truncated data

Tweet Count by Age Group

```
%sql
SELECT u.age as user_age, COUNT(t.id) as tweet_count
FROM users as u
JOIN tweets as t ON u.id = t.id
WHERE u.age IN ('young', 'old')
GROUP BY u.age;
```

Table

	user_age	tweet_count
1	old	13214
2	young	18003

2 rows

Average Number of Followers by Age Group

```
%sql
SELECT
  u.age,
  AVG(follower_count) AS avg_followers
FROM
  users u
LEFT JOIN (
  SELECT
    following,
    COUNT(*) AS follower_count
  FROM
    followers
  GROUP BY
    following
) AS follower_counts ON u.id = follower_counts.following
GROUP BY
  u.age;
```

Table

	age	avg_followers
1	unknown	3.500079763644369
2	old	3.4829962887222603
3	young	3.5051162273384495

3 rows

Top 10 Hashtags Used by Younger Users

```
%sql
SELECT SUBSTRING_INDEX(SUBSTRING_INDEX(text, '#', -1), ' ', 1) as hashtag,
       COUNT(*) as hashtag_count
FROM tweets
WHERE id IN (SELECT id FROM users WHERE age = 'young')
   AND LOCATE('#', text) > 0
GROUP BY hashtag
ORDER BY hashtag_count DESC
LIMIT 10;
```

Table

	hashtag	hashtag_count
1	musicmonday	383
2	music	45
3	followfriday	25
4	iranelection	22
5	fb	12
6	myweakness	11
7	music4good	11

10 rows

Whether Young or Older Users Generally Follow Users in their Same Age Group

```
%sql
SELECT
  'young' AS follower_age,
  'young' AS following_age,
  COUNT(*) AS count
FROM
  Users u1
JOIN
  Followers f ON u1.id = f.id
JOIN
  Users u2 ON f.id = u2.id
WHERE
  u1.age = 'young'
  AND u2.age = 'young'
GROUP BY
  follower_age, following_age

UNION ALL

SELECT
  'old' AS follower_age,
  'old' AS following_age,
  COUNT(*) AS count
FROM
  Users u1
JOIN
  Followers f ON u1.id = f.id
JOIN
  Users u2 ON f.id = u2.id
WHERE
  u1.age = 'old'
  AND u2.age = 'old'
GROUP BY
  follower_age, following_age;
```

Table

	follower_age	following_age	count
1	young	young	63315
2	old	old	46333

2 rows

Top 5 Users with The Most Followers

```
%sql
SELECT u.id, u.user, COUNT(f.id) as follower_count
FROM users as u
LEFT JOIN followers as f ON u.id = f.id
GROUP BY u.id, u.user
ORDER BY follower_count DESC
LIMIT 5;
```

Table

	id	user	follower_count
1	787578650	stupiddie	14
2	89310999	tyefighter	14
3	722412147	aminhamenina	14
4	511719570	Jojo_x_Mojo	13
5	719161224	RPDOfficer	13

5 rows

Top 5 Users with Most Tweet Counts

```
%sql
SELECT u.id, u.user, COUNT(t.id) as tweet_count
FROM users as u
JOIN tweets as t ON u.id = t.id
GROUP BY u.id, u.user
ORDER BY tweet_count DESC
LIMIT 5;
```

Table

	id	user	tweet_count
1	285239913	droidgeek	10
2	226589861	unwritten_99	8
3	88456333	Camera_shy89	8
4	53315	DjGundam	2
5	321379	steveslee	2

5 rows

Tweet Frequency by Day of Week

```
%sql
SELECT DAYOFWEEK(t.timestamp) as day_of_week, COUNT(t.id) as tweet_count
FROM tweets as t
GROUP BY day_of_week
ORDER BY day_of_week;
```

Table

	day_of_week	tweet_count
1	1	344555
2	2	310225
3	3	185850
4	4	96806
5	5	106035

6	6	225594
7	7	330955

7 rows

Hour with The Most Tweets

Table

	tweet_hour	tweet_count
1	23	84750
2	7	83654
3	0	80865
4	6	80852
5	5	78623
6	22	78348
7	4	76995

24 rows