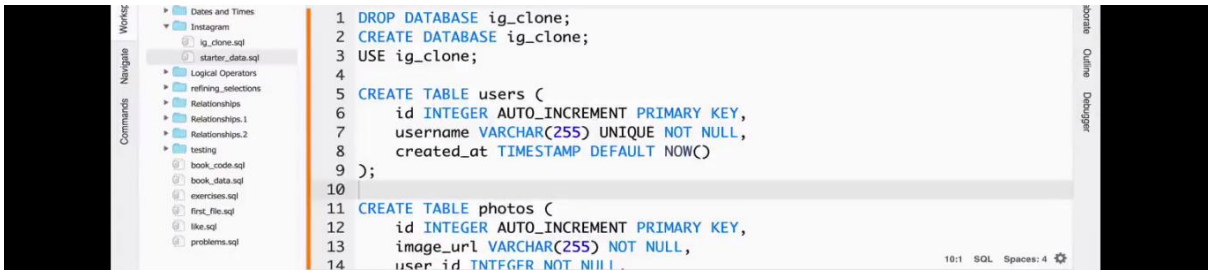
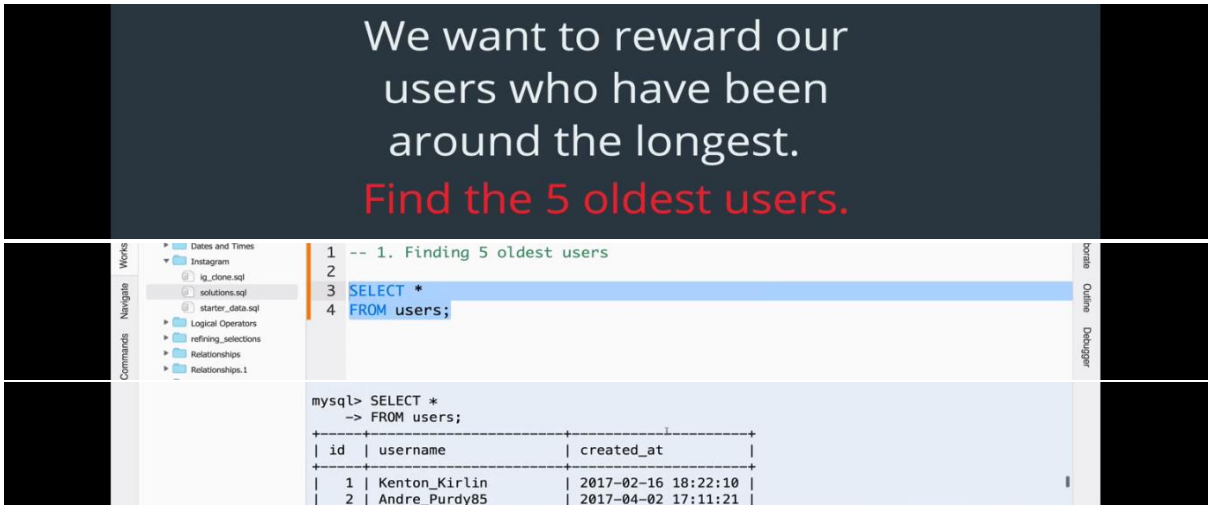
Lesson 15- working with lots of Instagram data

1- Loading the Instagram jumbo dataset
2- Instagram clone challenge 1
3- Instagram clone challenge 1 solution
4- Instagram clone challenge 2
5- Instagram clone challenge 2 solution
6- Instagram clone challenge 3
7- Instagram clone challenge 3 solution
8- Instagram clone challenge 4
9- Instagram clone challenge 4 solution
10- Instagram clone challenge 5
11- Instagram clone challenge 5 solution
12- Instagram clone challenge 6
13- Instagram clone challenge 6 solution
14- Instagram clone challenge 7
15- Instagram clone challenge 7 solution

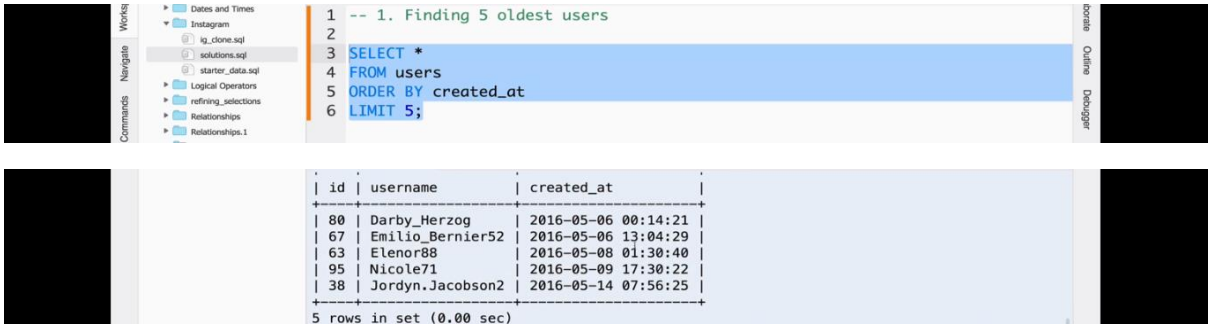
Create database



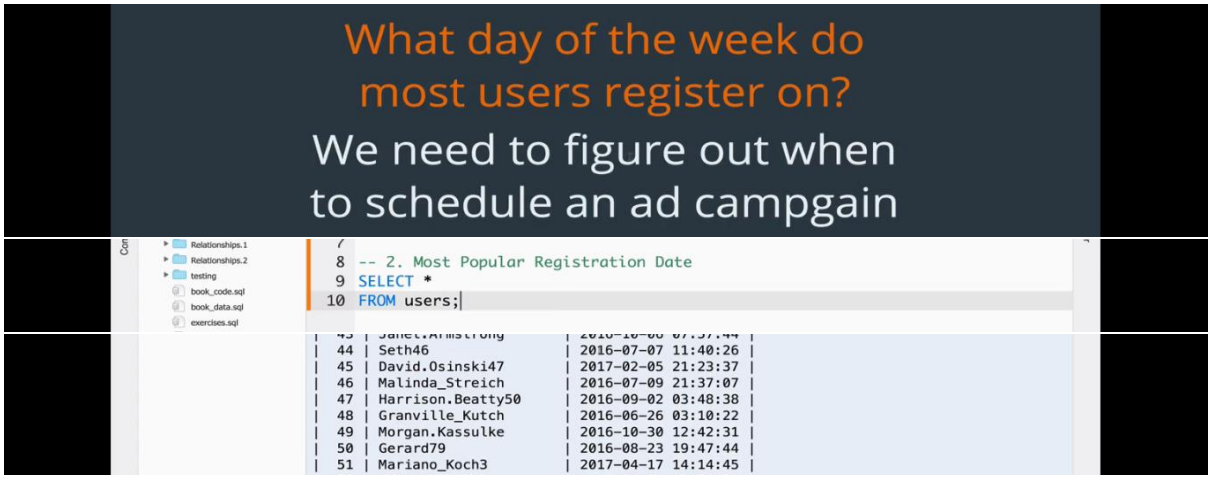
Challenges 1


We want to reward our users who have been around the longest.
Find the 5 oldest users.
-- 1. Finding 5 oldest users
SELECT *
FROM users;
mysql> SELECT *
-> FROM users;
id	username	created_at
1	Kenton_Kirlin	2017-02-16 18:22:10
2	Andre_Purdy85	2017-04-02 17:11:21


-- 1. Finding 5 oldest users
SELECT *
FROM users
ORDER BY created_at;
id	username	created_at
80	Darby_Herzog	2016-05-06 00:14:21
67	Emilio_Bernier52	2016-05-06 13:04:29
63	Elenor88	2016-05-08 01:30:40
95	Nicole71	2016-05-09 17:30:22
38	Jordyn.Jacobson2	2016-05-14 07:56:25
71	Nia_Haag	2016-05-14 15:38:50


-- 1. Finding 5 oldest users
SELECT *
FROM users
ORDER BY created_at
LIMIT 5;
id	username	created_at
80	Darby_Herzog	2016-05-06 00:14:21
67	Emilio_Bernier52	2016-05-06 13:04:29
63	Elenor88	2016-05-08 01:30:40
95	Nicole71	2016-05-09 17:30:22
38	Jordyn.Jacobson2	2016-05-14 07:56:25
5 rows in set (0.00 sec)

Challenge 2


What day of the week do most users register on?
We need to figure out when to schedule an ad campgain
-- 2. Most Popular Registration Date
SELECT *
FROM users;
44	Seth46	2016-07-07 11:40:26
45	David.Osinski47	2017-02-05 21:23:37
46	Malinda_Streich	2016-07-09 21:37:07
47	Harrison.Beatty50	2016-09-02 03:48:38
48	Granville_Kutch	2016-06-26 03:10:22
49	Morgan.Kassulke	2016-10-30 12:42:31
50	Gerard79	2016-08-23 19:47:44
51	Mariano_Koch3	2017-04-17 14:14:45

```
 8  -- 2. Most Popular Registration Date
 9  SELECT
10      username,
11      DAYNAME(created_at)
12  FROM users;
```

```
| Justina.Gaylord27     | Thursday  |
| Dereck65              | Thursday  |
| Alexandro35           | Wednesday |
| Jaclyn81              | Monday    |
| Billy52               | Wednesday |
| Annalise.McKenzie16   | Tuesday   |
| Norbert_Carroll35     | Monday    |
| Odessa2               | Friday    |
```

```
 8  -- 2. Most Popular Registration Date
 9  SELECT
10      username,
11      DAYNAME(created_at) AS day
12  FROM users
13  GROUP BY day;
```

```
+--------------------+-----------+
| Gus93              | Friday    |
| Kasandra_Homenick  | Monday    |
| Arely_Bogan63      | Saturday  |
| Andre_Purdy85      | Sunday    |
| Kenton_Kirlin      | Thursday  |
| Harley_Lind18      | Tuesday   |
| Aniya_Hackett      | Wednesday |
+--------------------+-----------+
```

```
 8  -- 2. Most Popular Registration Date
 9  SELECT
10      DAYNAME(created_at) AS day,
11      COUNT(*)
12  FROM users
13  GROUP BY day;
```

```
+-----------+----------+
| Friday    |       15 |
| Monday    |       14 |
| Saturday  |       12 |
| Sunday    |       16 |
| Thursday  |       16 |
| Tuesday   |       14 |
| Wednesday |       13 |
+-----------+----------+
```

```
 8  -- 2. Most Popular Registration Date
 9  SELECT
10      DAYNAME(created_at) AS day,
11      COUNT(*) AS total
12  FROM users
13  GROUP BY day
14  ORDER BY total;
```

```
| day       | total |
+-----------+-------+
| Saturday  |    12 |
| Wednesday |    13 |
| Monday    |    14 |
| Tuesday   |    14 |
| Friday    |    15 |
| Thursday  |    16 |
| Sunday    |    16 |
+-----------+-------+
```

```
 8  -- 2. Most Popular Registration Date
 9  SELECT
10      DAYNAME(created_at) AS day,
11      COUNT(*) AS total
12  FROM users
13  GROUP BY day
14  ORDER BY total DESC;
```

```
| day       | total |
+-----------+-------+
| Thursday  |    16 |
| Sunday    |    16 |
| Friday    |    15 |
| Monday    |    14 |
| Tuesday   |    14 |
| Wednesday |    13 |
| Saturday  |    12 |
```

```
 8  -- 2. Most Popular Registration Date
 9  SELECT
10      DAYNAME(created_at) AS day,
11      COUNT(*) AS total
12  FROM users
13  GROUP BY day
14  ORDER BY total DESC
15  LIMIT 2;
```

```
+----------+-------+
| day      | total |
+----------+-------+
| Thursday |    16 |
| Sunday   |    16 |
+----------+-------+
2 rows in set (0.01 sec)
```

Challenge 3

We want to target our inactive users with an email campaign.

Find the users who have never posted a photo

```
17  -- 3. Identify Inactive Users (users with no photos)
18  SELECT * FROM users;
19
```

```
|  62 | Ressie_Stanton46      | 2016-12-20 15:09:08 |
|  63 | Elenor88              | 2016-05-08 01:30:40 |
|  64 | Florence99            | 2016-10-06 23:08:30 |
|  65 | Adelle96              | 2016-10-01 00:37:57 |
|  66 | Mike.Auer39           | 2016-07-01 17:36:14 |
|  67 | Emilio_Bernier52      | 2016-05-06 13:04:29 |
|  68 | Franco_Keebler64      | 2016-11-13 20:09:26 |
|  69 | Karley_Bosco          | 2016-06-24 23:38:52 |
```

```
17  -- 3. Identify Inactive Users (users with no photos)
18  SELECT * FROM users;
19  SELECT * FROM photos;
20
```

```
| 231 | https://mariano.net    |      88 | 2017-05-06 18:30:41 |
| 232 | https://brittany.org   |      88 | 2017-05-06 18:30:41 |
| 233 | https://madaline.name  |      88 | 2017-05-06 18:30:41 |
| 234 | https://devonte.name   |      88 | 2017-05-06 18:30:41 |
| 235 | https://blanche.net    |      88 | 2017-05-06 18:30:41 |
| 236 | http://cali.net        |      88 | 2017-05-06 18:30:41 |
| 237 | http://mekhi.name      |      88 | 2017-05-06 18:30:41 |
| 238 | http://adela.com       |      88 | 2017-05-06 18:30:41 |
| 239 | https://devan.com      |      92 | 2017-05-06 18:30:41 |
```

```
17  -- 3. Identify Inactive Users (users with no photos)
18  SELECT *
19  FROM users
20  INNER JOIN photos
21      ON users.id = photos.user_id;
22
```

```
95 | 2017-05-06 18:30:41 |
|  95 | Nicole71              | 2016-05-09 17:30:22 | 246 | https://kathleen.biz   |
95 | 2017-05-06 18:30:41 |
|  96 | Keenan.Schamberger60  | 2016-08-28 14:57:28 | 247 | https://helmer.org     |
96 | 2017-05-06 18:30:41 |
|  96 | Keenan.Schamberger60  | 2016-08-28 14:57:28 | 248 | https://maggie.info    |
96 | 2017-05-06 18:30:41 |
|  96 | Keenan.Schamberger60  | 2016-08-28 14:57:28 | 249 | https://cecilia.net    |
```

```
17  -- 3. Identify Inactive Users (users with no photos)
18  SELECT username, image_url
19  FROM users
20  INNER JOIN photos
21      ON users.id = photos.user_id;
22
```

```
| Justina.Gaylord27     | https://henderson.com   |
| Justina.Gaylord27     | http://bonnie.info      |
| Justina.Gaylord27     | http://kennith.net      |
| Justina.Gaylord27     | http://camille.name     |
| Kaley9                | https://jensen.name     |
| Kaley9                | http://virginia.org     |
| Karley_Bosco          | http://reva.com         |
| Katarina.Dibbert      | https://ford.biz        |
```

```
-- 3. Identify Inactive Users (users with no photos)
SELECT username, image_url
FROM users
LEFT JOIN photos
    ON users.id = photos.user_id;
```

```
| Rick29               | https://celestino.name |
| Rick29               | http://violet.info     |
| Rick29               | http://nigel.biz       |
| Rocio33              | NULL                   |
| Sam52                | http://blanca.org      |
| Sam52                | http://delpha.com      |
| Seth46               | https://olga.org       |
| Seth46               | https://donavon.org    |
| Seth46               | http://moses.biz       |
```

```
-- 3. Identify Inactive Users (users with no photos)
SELECT username, image_url
FROM users
LEFT JOIN photos
    ON users.id = photos.user_id
WHERE photos.id IS NULL;
```

```
+---------------------+-----------+
| username            | image_url |
+---------------------+-----------+
| Aniya_Hackett       | NULL      |
| Bartholome.Bernhard | NULL      |
| Bethany20           | NULL      |
| Darby_Herzog        | NULL      |
| David.Osinski47     | NULL      |
| Duane60             | NULL      |
```

```
-- 3. Identify Inactive Users (users with no photos)
SELECT username
FROM photos
RIGHT JOIN users
    ON users.id = photos.user_id
WHERE photos.id IS NULL;
```

```
| Bethany20           |
| Darby_Herzog        |
| David.Osinski47     |
| Duane60             |
| Esmeralda.Mraz57    |
| Esther.Zulauf61     |
| Franco_Keebler64    |
| Hulda.Macejkovic    |
```

Challenge 4

We're running a new contest to see who can get the most likes on a single photo.

WHO WON??!!

```
-- 4. Identify most popular photo (and user who created it)
SELECT *
FROM photos;
```

```
| 244 | http://freeda.biz    |      94 | 2017-05-06 18:30:41 |
| 245 | http://ayden.name    |      95 | 2017-05-06 18:30:41 |
| 246 | https://kathleen.biz |      95 | 2017-05-06 18:30:41 |
| 247 | https://helmer.org   |      96 | 2017-05-06 18:30:41 |
| 248 | https://maggie.info  |      96 | 2017-05-06 18:30:41 |
| 249 | https://cecilia.net  |      96 | 2017-05-06 18:30:41 |
| 250 | http://ayla.org      |      97 | 2017-05-06 18:30:41 |
| 251 | https://elyssa.biz   |      97 | 2017-05-06 18:30:41 |
| 252 | http://jennie.com    |      98 | 2017-05-06 18:30:41 |
```

```
SELECT * FROM likes;
```

```
|     100 |      246 | 2017-05-06 18:30:42 |
|     100 |      248 | 2017-05-06 18:30:42 |
|     100 |      249 | 2017-05-06 18:30:42 |
|     100 |      255 | 2017-05-06 18:30:42 |
+---------+----------+---------------------+
8782 rows in set (0.00 sec)
```

```
-- 4. Identify most popular photo (and user who created it)
SELECT *
FROM photos
INNER JOIN likes
    ON likes.photo_id = photos.id;
```

```
| 257 | http://dedrick.info    |     100 | 2017-05-06 18:30:41 |      95 |      257 | 2017
-05-06 18:30:42 |
| 257 | http://dedrick.info    |     100 | 2017-05-06 18:30:41 |      97 |      257 | 2017
-05-06 18:30:42 |
+-----+------------------------+---------+---------------------+---------+----------+-----
----------------+
8782 rows in set (0.12 sec)
```

```
-- 4. Identify most popular photo (and user who created it)
SELECT photos.id, photos.image_url, likes.user_id
FROM photos
INNER JOIN likes
    ON likes.photo_id = photos.id;
```

```
| 257 | http://dedrick.info    |      48 |
| 257 | http://dedrick.info    |      54 |
| 257 | http://dedrick.info    |      57 |
| 257 | http://dedrick.info    |      60 |
| 257 | http://dedrick.info    |      61 |
| 257 | http://dedrick.info    |      65 |
| 257 | http://dedrick.info    |      66 |
| 257 | http://dedrick.info    |      70 |
| 257 | http://dedrick.info    |      71 |
```

```
-- 4. Identify most popular photo (and user who created it)
SELECT
    photos.id,
    photos.image_url,
    likes.user_id
FROM photos
INNER JOIN likes
    ON likes.photo_id = photos.id
GROUP BY photos.id;
```

```
| 253 | http://ryleigh.info    |       3 |
| 254 | https://darien.name    |       5 |
| 255 | https://xzavier.org    |       5 |
| 256 | https://kaela.name     |       3 |
| 257 | http://dedrick.info    |       3 |
+-----+------------------------+---------+
257 rows in set (0.01 sec)
```

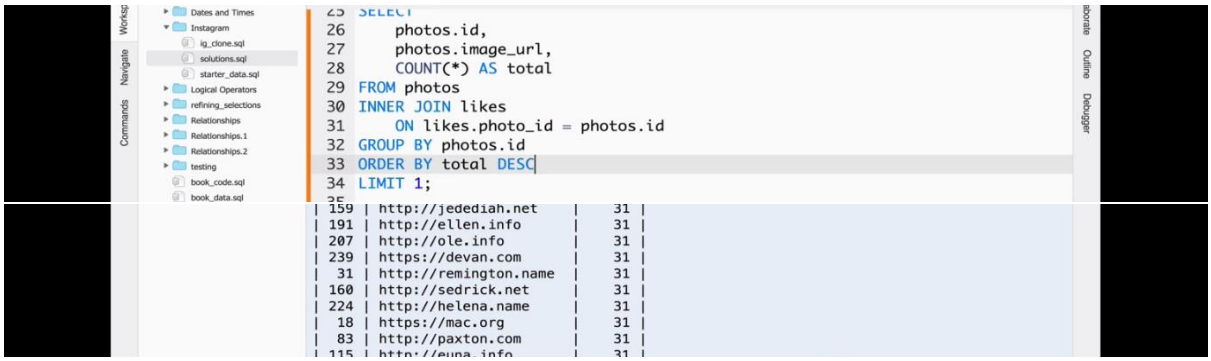
```
-- 4. Identify most popular photo (and user who created it)
SELECT
    photos.id,
    photos.image_url,
    COUNT(*)
FROM photos
INNER JOIN likes
    ON likes.photo_id = photos.id
GROUP BY photos.id;
```

```
| 237 | http://mekhi.name      |      32 |
| 238 | http://adela.com       |      27 |
| 239 | https://devan.com      |      31 |
| 240 | https://jarrett.name   |      30 |
| 241 | https://sid.biz        |      37 |
| 242 | https://jadyn.name     |      33 |
| 243 | https://erik.com       |      35 |
| 244 | http://freeda.biz      |      40 |
```

```
-- 4. Identify most popular photo (and user who created it)
SELECT
    photos.id,
    photos.image_url,
    COUNT(*) AS total
FROM photos
INNER JOIN likes
    ON likes.photo_id = photos.id
GROUP BY photos.id
ORDER BY total;
```

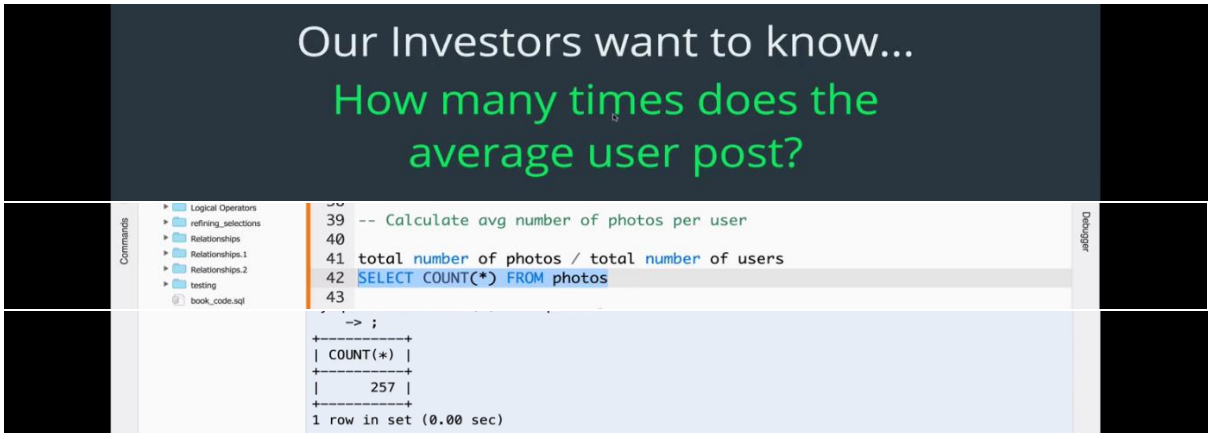
```
| 177 | https://nolan.name     |    36 |
| 257 | http://dedrick.info    |    36 |
|  17 | http://annamae.name    |    36 |
|   2 | https://shanon.org     |    36 |
| 211 | https://dimitri.info   |    36 |
| 116 | https://maymie.net     |    36 |
|  53 | http://collin.com      |    36 |
|  85 | http://fabiola.org     |    36 |
```

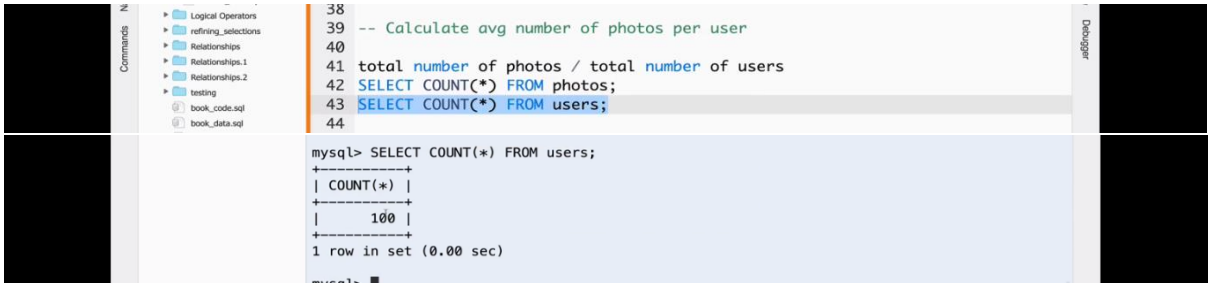

-- 4. Identify most popular photo (and user who created it)
SELECT
photos.id,
photos.image_url,
COUNT(*) AS total
FROM photos
INNER JOIN likes
ON likes.photo_id = photos.id
GROUP BY photos.id
ORDER BY total DESC;
193	https://javonte.org	30
179	http://ora.net	30
132	http://kailee.org	30
184	http://rico.biz	30
216	http://deontae.org	30
92	http://lacy.biz	29
108	http://boris.biz	29
12	http://felicity.name	29

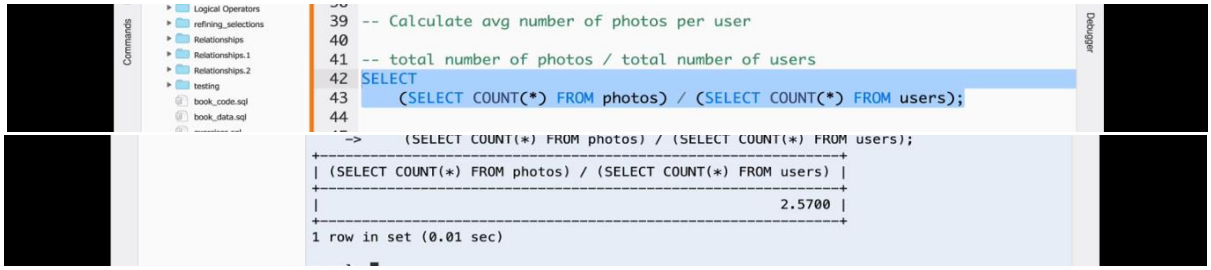

photos.id,
photos.image_url,
COUNT(*) AS total
FROM photos
INNER JOIN likes
ON likes.photo_id = photos.id
GROUP BY photos.id
ORDER BY total DESC
LIMIT 1;
159	http://jedediah.net	31
191	http://ellen.info	31
207	http://ole.info	31
239	https://devan.com	31
31	http://remington.name	31
160	http://sedrick.net	31
224	http://helena.name	31
18	https://mac.org	31
83	http://paxton.com	31


-- 4. Identify most popular photo (and user who created it)
SELECT
username,
photos.id,
photos.image_url,
COUNT(*) AS total
FROM photos
INNER JOIN likes
ON likes.photo_id = photos.id
INNER JOIN users
ON photos.user_id = users.id
GROUP BY photos.id
ORDER BY total DESC
(242 Bytes) 25:1 SQL Spaces: 4
-> LIMIT 1;
| username | id | image_url | total |
| Zack_Kemmer93 | 145 | https://jarret.name | 48 |
1 row in set (0.01 sec)

Challenge 5

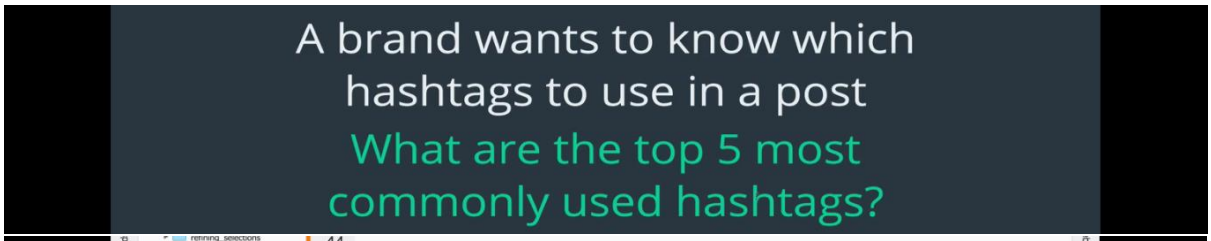

Our Investors want to know...
How many times does the average user post?
-- Calculate avg number of photos per user
total number of photos / total number of users
SELECT COUNT(*) FROM photos
-> ;
| COUNT(*) |
| 257 |
1 row in set (0.00 sec)

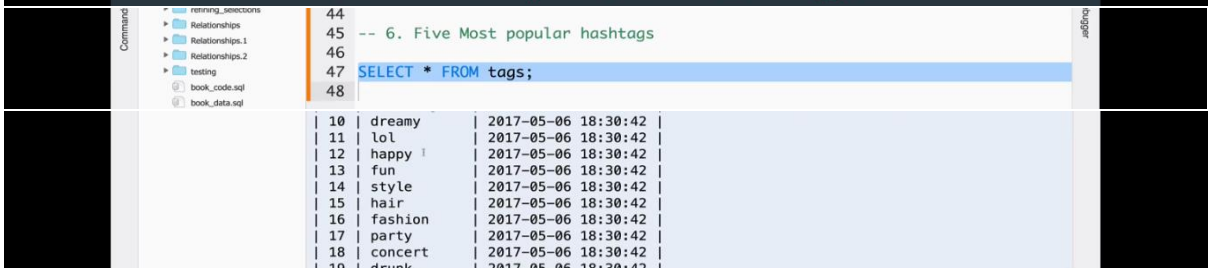

-- Calculate avg number of photos per user
total number of photos / total number of users
SELECT COUNT(*) FROM photos;
SELECT COUNT(*) FROM users;
mysql> SELECT COUNT(*) FROM users;
| COUNT(*) |
| 100 |
1 row in set (0.00 sec)

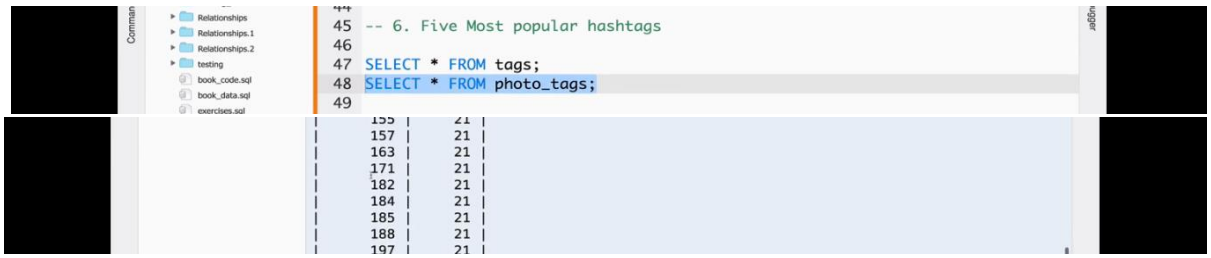

-- Calculate avg number of photos per user
-- total number of photos / total number of users
SELECT
(SELECT COUNT(*) FROM photos) / (SELECT COUNT(*) FROM users);
| (SELECT COUNT(*) FROM photos) / (SELECT COUNT(*) FROM users) |
| 2.5700 |
1 row in set (0.01 sec)


-- Calculate avg number of photos per user
-- total number of photos / total number of users
SELECT
(SELECT COUNT(*) FROM photos) / (SELECT COUNT(*) FROM users) AS avg;
| avg |
| 2.5700 |
1 row in set (0.00 sec)

Challenge 6


A brand wants to know which hashtags to use in a post
What are the top 5 most commonly used hashtags?


-- 6. Five Most popular hashtags
SELECT * FROM tags;
10	dreamy	2017-05-06 18:30:42
11	lol	2017-05-06 18:30:42
12	happy	2017-05-06 18:30:42
13	fun	2017-05-06 18:30:42
14	style	2017-05-06 18:30:42
15	hair	2017-05-06 18:30:42
16	fashion	2017-05-06 18:30:42
17	party	2017-05-06 18:30:42
18	concert	2017-05-06 18:30:42


-- 6. Five Most popular hashtags
SELECT * FROM tags;
SELECT * FROM photo_tags;
157	21
163	21
171	21
182	21
184	21
185	21
188	21

```sql
-- 6. Five Most popular hashtags

SELECT *
FROM photo_tags
JOIN tags
    ON photo_tags.tag_id = tags.id;
```

```sql
-- 6. Five Most popular hashtags

SELECT *
FROM photo_tags
JOIN tags
    ON photo_tags.tag_id = tags.id
GROUP BY tags.id;
```

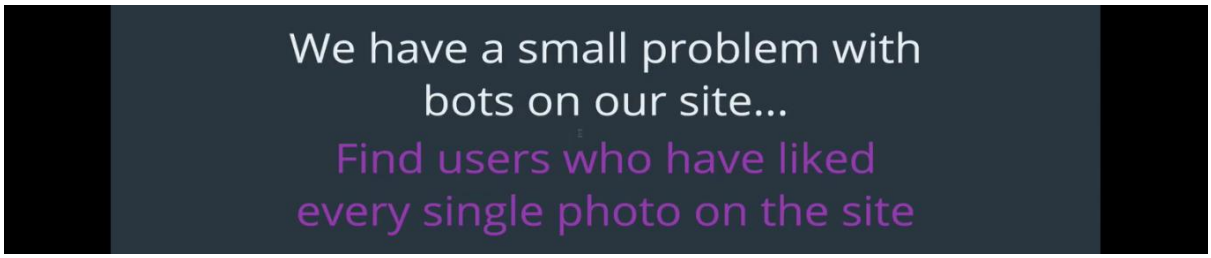
```sql
-- 6. Five Most popular hashtags

SELECT
    tags.tag_name,
    COUNT(*)
FROM photo_tags
JOIN tags
    ON photo_tags.tag_id = tags.id
GROUP BY tags.id;
```

```sql
-- 6. Five Most popular hashtags

SELECT
    tags.tag_name,
    COUNT(*) as total
FROM photo_tags
JOIN tags
    ON photo_tags.tag_id = tags.id
GROUP BY tags.id;
```

```sql
-- 6. Five Most popular hashtags

SELECT
    tags.tag_name,
    COUNT(*) as total
FROM photo_tags
JOIN tags
    ON photo_tags.tag_id = tags.id
GROUP BY tags.id
ORDER BY total;
```

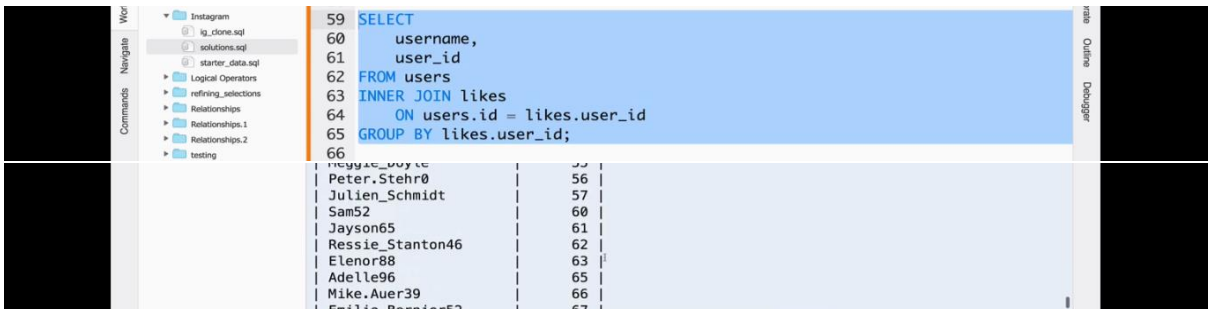
```sql
-- 6. Five Most popular hashtags

SELECT
    tags.tag_name,
    COUNT(*) as total
FROM photo_tags
JOIN tags
    ON photo_tags.tag_id = tags.id
GROUP BY tags.id
ORDER BY total DESC
LIMIT 5;
```

Challenge 7

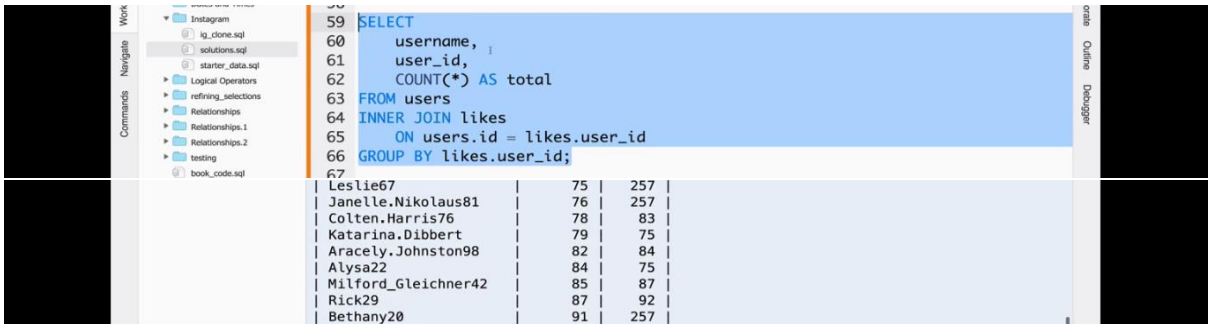


```
refining_selections
Relationships
Relationships.1
Relationships.2
testing
book_code.sql
book_data.sql
exercises.sql
first_file.sql

57 -- 7. Finding Bots - users who have liked every single photo
58
59 SELECT *
60 FROM users
61 INNER JOIN likes
62     ON users.id = likes.user_id;
63

|  99 | Alek_Watsica          | 2016-12-10 07:43:58 |      99 |      160 | 2017-05-06 18:3
0:42 |
|  99 | Alek_Watsica          | 2016-12-10 07:43:58 |      99 |      161 | 2017-05-06 18:3
0:42 |
|  99 | Alek_Watsica          | 2016-12-10 07:43:58 |      99 |      175 | 2017-05-06 18:3
0:42 |
|  99 | Alek_Watsica          | 2016-12-10 07:43:58 |      99 |      178 | 2017-05-06 18:3
0:42 |
```

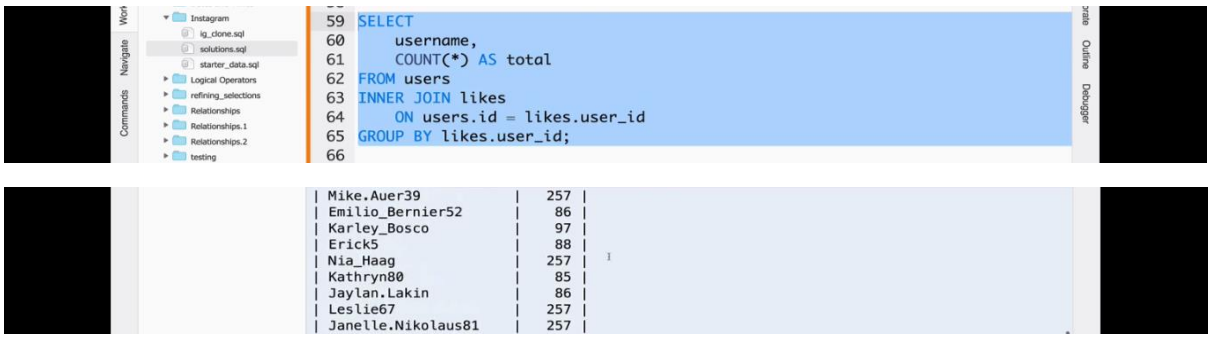
```
refining_selections
Relationships
Relationships.1
Relationships.2
testing
book_code.sql
book_data.sql
exercises.sql
first_file.sql
like.sql

57 -- 7. Finding Bots - users who have liked every single photo
58
59 SELECT *
60 FROM users
61 INNER JOIN likes
62     ON users.id = likes.user_id
63 GROUP BY likes.user_id;
64

|  99 | Alek_Watsica          | 2016-12-10 07:43:58 |      99 |        3 | 2017-05-06 18:3
0:42 |
| 100 | Javonte83             | 2017-03-27 22:06:37 |     100 |        3 | 2017-05-06 18:3
0:42 |
+-----+-----------------------+---------------------+---------+----------+----------------
------+
77 rows in set (0.01 sec)
```



```
Instagram
ig_clone.sql
solutions.sql
starter_data.sql
Logical Operators
refining_selections
Relationships
Relationships.1
Relationships.2
testing

59 SELECT
60     username,
61     user_id
62 FROM users
63 INNER JOIN likes
64     ON users.id = likes.user_id
65 GROUP BY likes.user_id;
66

| Peter.Stehr0          |      56 |
| Julien_Schmidt        |      57 |
| Sam52                 |      60 |
| Jayson65              |      61 |
| Ressie_Stanton46      |      62 |
| Elenor88              |      63 |
| Adelle96              |      65 |
| Mike.Auer39           |      66 |
```

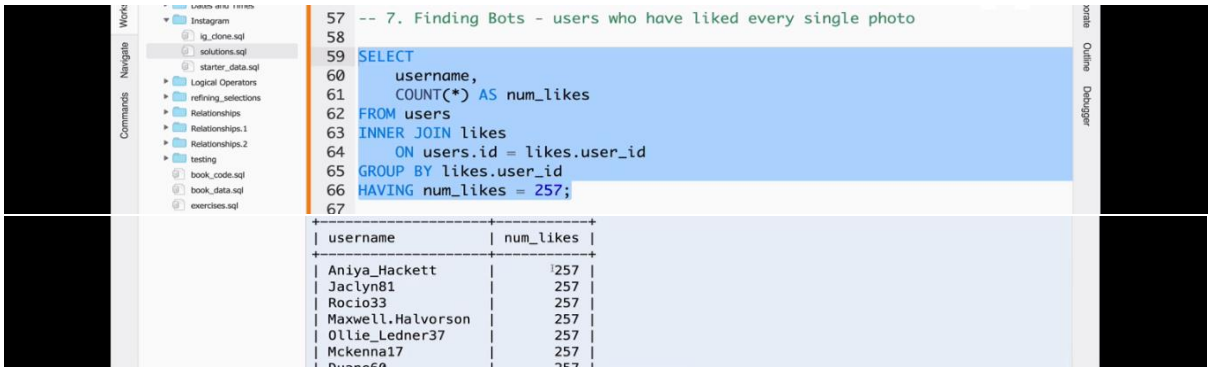


```
Instagram
ig_clone.sql
solutions.sql
starter_data.sql
Logical Operators
refining_selections
Relationships
Relationships.1
Relationships.2
testing
book_code.sql

59 SELECT
60     username,
61     user_id,
62     COUNT(*) AS total
63 FROM users
64 INNER JOIN likes
65     ON users.id = likes.user_id
66 GROUP BY likes.user_id;
67

| Leslie67              |      75 |   257 |
| Janelle.Nikolaus81    |      76 |   257 |
| Colten.Harris76       |      78 |    83 |
| Katarina.Dibbert      |      79 |    75 |
| Aracely.Johnston98    |      82 |    84 |
| Alysa22               |      84 |    75 |
| Milford_Gleichner42   |      85 |    87 |
| Rick29                |      87 |    92 |
| Bethany20             |      91 |   257 |
```



```
Instagram
ig_clone.sql
solutions.sql
starter_data.sql
Logical Operators
refining_selections
Relationships
Relationships.1
Relationships.2
testing

59 SELECT
60     username,
61     COUNT(*) AS total
62 FROM users
63 INNER JOIN likes
64     ON users.id = likes.user_id
65 GROUP BY likes.user_id;
66

| Mike.Auer39           |   257 |
| Emilio_Bernier52      |    86 |
| Karley_Bosco          |    97 |
| Erick5                |    88 |
| Nia_Haag              |   257 |
| Kathryn80             |    85 |
| Jaylan.Lakin          |    86 |
| Leslie67              |   257 |
| Janelle.Nikolaus81    |   257 |
```


Instagram
ig_clone.sql
solutions.sql
starter_data.sql
Logical Operators
refining_selections
Relationships
Relationships.1
Relationships.2
testing
book_code.sql
59 SELECT
60 username,
61 COUNT(*) AS num_likes
62 FROM users
63 INNER JOIN likes
64 ON users.id = likes.user_id
65 GROUP BY likes.user_id
66 WHERE num_likes = 257;
-> COUNT(*) AS num_likes
-> FROM users
-> INNER JOIN likes
-> ON users.id = likes.user_id
-> GROUP BY likes.user_id
-> WHERE num_likes = 257;
ERROR 1064 (42000): You have an error in your SQL syntax; check the manual that correspond
s to your MySQL server version for the right syntax to use near 'WHERE num_likes = 257' at
line 8


Instagram
ig_clone.sql
solutions.sql
starter_data.sql
Logical Operators
refining_selections
Relationships
Relationships.1
Relationships.2
testing
book_code.sql
book_data.sql
exercises.sql
57 -- 7. Finding Bots - users who have liked every single photo
59 SELECT
60 username,
61 COUNT(*) AS num_likes
62 FROM users
63 INNER JOIN likes
64 ON users.id = likes.user_id
65 GROUP BY likes.user_id
66 HAVING num_likes = 257;
username	num_likes
Aniya_Hackett	257
Jaclyn81	257
Rocio33	257
Maxwell.Halvorson	257
Ollie_Ledner37	257
Mckenna17	257


Instagram
ig_clone.sql
solutions.sql
starter_data.sql
Logical Operators
refining_selections
Relationships
Relationships.1
Relationships.2
testing
book_code.sql
book_data.sql
exercises.sql
first_file.sql
57 -- 7. Finding Bots - users who have liked every single photo
59 SELECT
60 username,
61 COUNT(*) AS num_likes
62 FROM users
63 INNER JOIN likes
64 ON users.id = likes.user_id
65 GROUP BY likes.user_id
66 HAVING num_likes = (SELECT COUNT(*) FROM photos);
username	num_likes
Aniya_Hackett	257
Jaclyn81	257
Rocio33	257