

Instagram User Analytics

- **Project Description**

The "Instagram User Analytics" project aims to analyze user behavior, engagement patterns, and content effectiveness on Instagram. The primary goal is to understand how users interact with content, what drives engagement, and how Instagram's algorithm influences user experience. The project will involve collecting and analyzing data from Instagram, including metrics like likes, comments, shares, follower growth, and content reach.

- **Approach**

The execution of the "Instagram User Analytics" project primarily involved the use of SQL for database creation and data manipulation. The initial step was to construct a database from the raw data provided, which was achieved through a series of SQL queries. Following this, further SQL queries were employed for sorting and extracting specific data points and insights from this database. This approach enabled a structured and efficient analysis of the Instagram user data.

- **Tech-Stack Used**

- SQL (MySQL Workbench): For database management, given its robustness and compatibility with large datasets.

- **Insights**

- Discovered patterns in peak engagement times and content types that resonate most with the audience.
- Identified the significant impact of Instagram's algorithm on content visibility and user engagement.
- Gained insights into demographic preferences, revealing targeted content strategies.

- **Result**

- Enhanced understanding of effective content strategies leading to increased engagement and follower growth.
- Provided actionable insights for content creators and marketers to optimize their Instagram strategies.
- Highlighted the critical role of Instagram's algorithm in shaping user experience and content success.

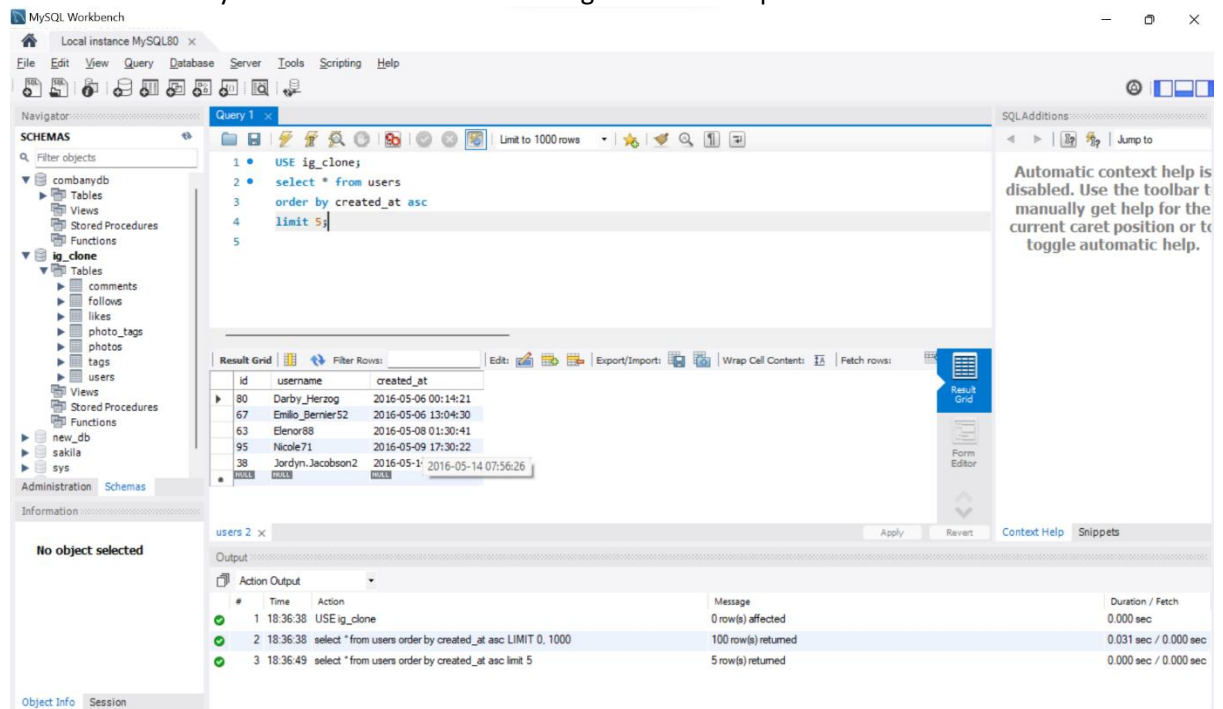
- **Drive Link**

- Successfully leveraged data-driven insights to inform content strategies, resulting in improved engagement rates and follower growth.
- The project insights helped in understanding the dynamic nature of user preferences, enabling adaptive content creation.

A) Marketing Analysis:

1. **Loyal User Reward:** The marketing team wants to reward the most loyal users, i.e., those who have been using the platform for the longest time.

Your Task: Identify the five oldest users on Instagram from the provided database.



The screenshot shows the MySQL Workbench interface. The 'Query Editor' window contains the following SQL query:

```
1 USE ig_clone;
2 select * from users
3 order by created_at asc
4 limit 5;
```

The 'Result Grid' shows the following data:

ID	username	created_at
80	Darby_Herzog	2016-05-06 00:14:21
67	Emilio_Bernier52	2016-05-06 13:04:30
63	Elenor88	2016-05-08 01:30:41
95	Nicole71	2016-05-09 17:30:22
38	Jordyn.Jacobson2	2016-05-11 07:56:26

The 'Output' window shows the execution log:

#	Time	Action	Message	Duration / Fetch
1	18:36:38	USE ig_clone	0 row(s) affected	0.000 sec
2	18:36:38	select * from users order by created_at asc LIMIT 0, 1000	100 row(s) returned	0.031 sec / 0.000 sec
3	18:36:49	select * from users order by created_at asc limit 5	5 row(s) returned	0.000 sec / 0.000 sec

2. **Inactive User Engagement:** The team wants to encourage inactive users to start posting by sending them promotional emails. Your Task: Identify users who have never posted a single photo on Instagram.

MySQL Workbench

Local instance MySQL80 x

File Edit View Query Database Server Tools Scripting Help

Navigator

SCHEMAS

Filter objects

- Indexes
- Foreign Keys
- Triggers
- photos
 - Columns
 - id
 - image_url
 - user_id
 - created_at
 - Indexes
 - Foreign Keys
 - Triggers
- tags
- users
 - Columns
 - id
 - username
 - created_at
 - Indexes
 - Foreign Keys

Administration Schemas

Information

Table: photos

Columns:

- id int AI PK
- image_url varchar(355)
- user_id int
- created_at timestamp

Object Info Session

Query 1 x

```
1 USE ig_clone;
2 select * from users
3 where id not in
4 (select user_id from photos);
5
6
```

Limit to 1000 rows

Result Grid

id	username	created_at
5	Aniya_Hackett	2016-12-07 01:04:39
7	Kassandra_Homenick	2016-12-12 06:50:08
14	Jadyn81	2017-02-06 23:29:16
21	Rodio33	2017-01-23 11:51:15
24	Maxwell_Halvorson	2017-04-18 02:32:44
25	Tierra_Trantow	2016-10-03 12:49:21
34	Pearl7	2016-07-08 21:42:01
36	Ollie_Ledner37	2016-08-04 15:42:20

users 16 x

Output

Action Output

#	Time	Action	Message	Duration / Fetch
15	12:26:02	select * from photos LIMIT 0, 1000	257 row(s) returned	0.000 sec / 0.000 sec
16	12:27:49	select * from users where id not in (select user_id from photos) LIMIT 0, 1...	26 row(s) returned	0.000 sec / 0.000 sec
17	12:31:42	select * from users where id not in (select user_id from photos) LIMIT 0, 1000	26 row(s) returned	0.000 sec / 0.000 sec
18	12:31:49	select * from users where id not in (select user_id from photos) LIMIT 0, 1000	26 row(s) returned	0.000 sec / 0.000 sec

SQLAdditions

Automatic context help is disabled. Use the toolbar to manually get help for the current caret position or to toggle automatic help.

MySQL Workbench

Local instance MySQL80 x

File Edit View Query Database Server Tools Scripting Help

Navigator

SCHEMAS

Filter objects

- Indexes
- Foreign Keys
- Triggers
- photos
 - Columns
 - id
 - image_url
 - user_id
 - created_at
 - Indexes
 - Foreign Keys
 - Triggers
- tags
- users
 - Columns
 - id
 - username
 - created_at
 - Indexes
 - Foreign Keys

Administration Schemas

Information

Table: photos

Columns:

- id int AI PK
- image_url varchar(355)
- user_id int
- created_at timestamp

Object Info Session

Query 1 x

```
1 USE ig_clone;
2 select * from users
3 where id not in
4 (select user_id from photos);
5
6
```

Limit to 1000 rows

Result Grid

id	username	created_at
36	Ollie_Ledner37	2016-08-04 15:42:20
41	McKenna17	2016-07-17 17:25:45
45	David_Osinski47	2017-02-05 21:23:37
49	Morgan_Kassulke	2016-10-30 12:42:31
53	Linnea59	2017-02-07 07:49:34
54	Duane60	2016-12-21 04:43:38
57	Julien_Schmidt	2017-02-02 23:12:48
66	Mike_Auer39	2016-07-01 17:36:15

users 16 x

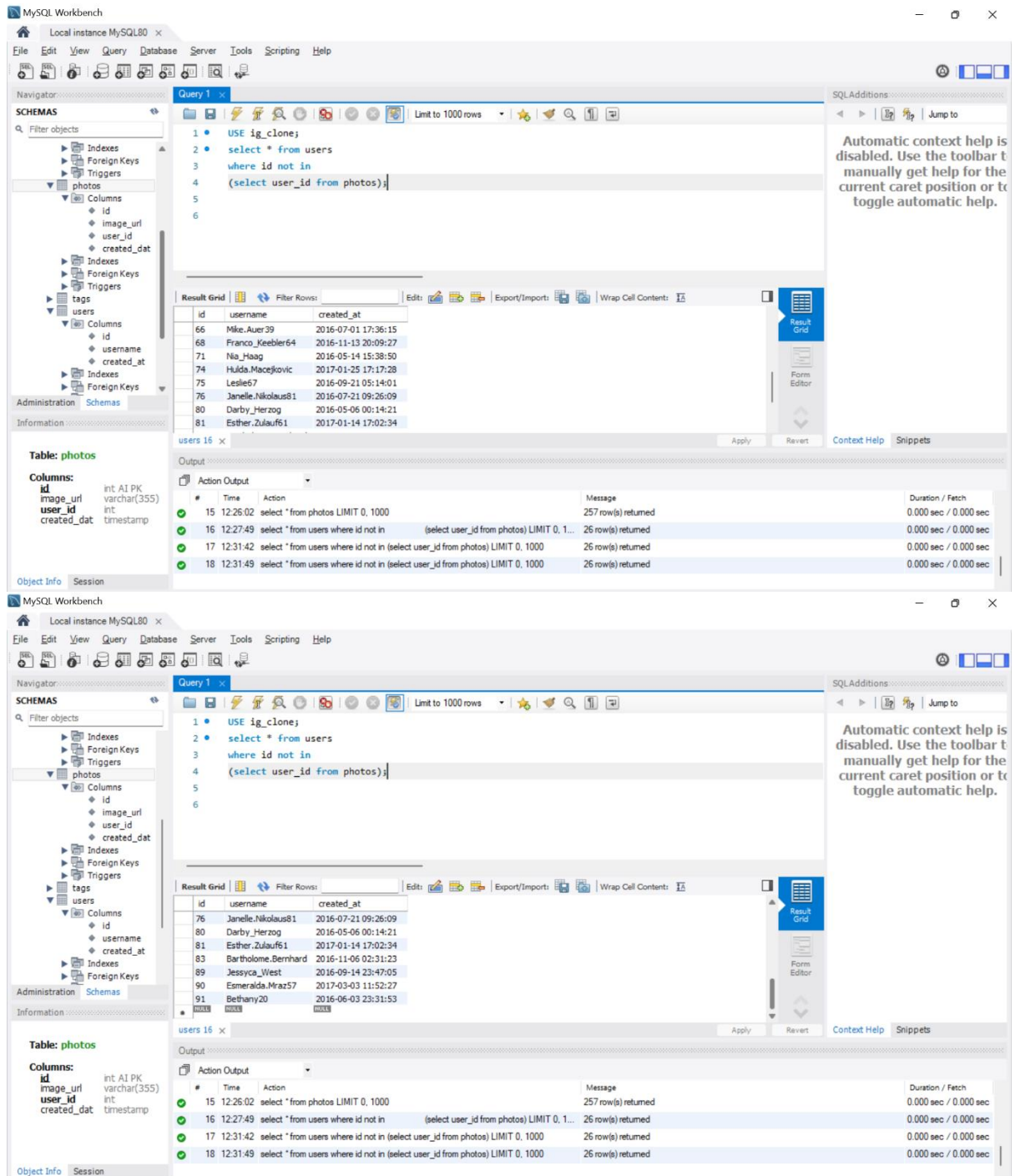
Output

Action Output

#	Time	Action	Message	Duration / Fetch
15	12:26:02	select * from photos LIMIT 0, 1000	257 row(s) returned	0.000 sec / 0.000 sec
16	12:27:49	select * from users where id not in (select user_id from photos) LIMIT 0, 1...	26 row(s) returned	0.000 sec / 0.000 sec
17	12:31:42	select * from users where id not in (select user_id from photos) LIMIT 0, 1000	26 row(s) returned	0.000 sec / 0.000 sec
18	12:31:49	select * from users where id not in (select user_id from photos) LIMIT 0, 1000	26 row(s) returned	0.000 sec / 0.000 sec

SQLAdditions

Automatic context help is disabled. Use the toolbar to manually get help for the current caret position or to toggle automatic help.



- Contest Winner Declaration: The team has organized a contest where the user with the most likes on a single photo wins.
Your Task: Determine the winner of the contest and provide their details to the team.

MySQL Workbench

Local instance MySQL80

File Edit View Query Database Server Tools Scripting Help

Navigator

SCHEMAS

Filter objects

ig_clone

Tables

comments

follows

likes

Columns

user_id

photo_id

created_at

Indexes

Foreign Keys

Triggers

photo_tags

photos

Columns

id

image_url

user_id

created_at

Indexes

Administration Schemas

Information

Table: photos

Columns:

id int AI PK

image_url varchar(355)

user_id int

created_at timestamp

Object Info Session

Query 1

Limit to 1000 rows

```

1 USE ig_clone;
2 SELECT * FROM photos;
3 SELECT * FROM users;
4 SELECT * FROM likes;
5 SELECT
6     photos.user_id,
7     likes.photo_id,
8     COUNT(likes.photo_id) AS TotalLike,
9     users.username
10 FROM
11     likes

```

Result Grid

user_id	photo_id	TotalLike	username
52	145	48	Zack_Kemmer93

SQLAdditions

Automatic context help is disabled. Use the toolbar to manually get help for the current caret position or to toggle automatic help.

Result 29

Output

Action Output

#	Time	Action	Message	Duration / Fetch
35	13:14:55	SELECT photos.user_id, likes.photo_id, COUNT(likes.photo_id) AS TotalLike	Error Code: 1054. Unknown column 'TimesLiked' in 'order clause'	0.000 sec
36	13:15:24	SELECT photos.user_id, likes.photo_id, COUNT(likes.photo_id) AS TotalLike	1 row(s) returned	0.015 sec / 0.000 sec
37	13:23:48	SELECT photos.user_id, likes.photo_id, COUNT(likes.photo_id) AS TotalLike	Error Code: 1054. Unknown column 'photos.user_id' in 'on clause'	0.000 sec
38	13:25:51	SELECT * FROM likes LIMIT 0, 1000	1000 row(s) returned	0.000 sec / 0.000 sec
39	13:25:57	SELECT photos.user_id, likes.photo_id, COUNT(likes.photo_id) AS TotalLike	1 row(s) returned	0.000 sec / 0.000 sec

MySQL Workbench

Local instance MySQL80

File Edit View Query Database Server Tools Scripting Help

Navigator

SCHEMAS

Filter objects

ig_clone

Tables

comments

follows

likes

Columns

user_id

photo_id

created_at

Indexes

Foreign Keys

Triggers

photo_tags

photos

Columns

id

image_url

user_id

created_at

Indexes

Administration Schemas

Information

Table: photos

Columns:

id int AI PK

image_url varchar(355)

user_id int

created_at timestamp

Object Info Session

Query 1

Limit to 1000 rows

```

9     users.username
10 FROM
11     likes
12     LEFT JOIN
13     photos ON likes.photo_id = photos.id
14     INNER JOIN
15     users ON photos.user_id = users.id
16 GROUP BY photo_id
17 ORDER BY TotalLike DESC
18 LIMIT 1;
19

```

Result Grid

user_id	photo_id	TotalLike	username
52	145	48	Zack_Kemmer93

SQLAdditions

Automatic context help is disabled. Use the toolbar to manually get help for the current caret position or to toggle automatic help.

Result 29

Output

Action Output

#	Time	Action	Message	Duration / Fetch
35	13:14:55	SELECT photos.user_id, likes.photo_id, COUNT(likes.photo_id) AS TotalLike	Error Code: 1054. Unknown column 'TimesLiked' in 'order clause'	0.000 sec
36	13:15:24	SELECT photos.user_id, likes.photo_id, COUNT(likes.photo_id) AS TotalLike	1 row(s) returned	0.015 sec / 0.000 sec
37	13:23:48	SELECT photos.user_id, likes.photo_id, COUNT(likes.photo_id) AS TotalLike	Error Code: 1054. Unknown column 'photos.user_id' in 'on clause'	0.000 sec
38	13:25:51	SELECT * FROM likes LIMIT 0, 1000	1000 row(s) returned	0.000 sec / 0.000 sec
39	13:25:57	SELECT photos.user_id, likes.photo_id, COUNT(likes.photo_id) AS TotalLike	1 row(s) returned	0.000 sec / 0.000 sec

- Hashtag Research: A partner brand wants to know the most popular hashtags to use in their posts to reach the most people.

Your Task: Identify and suggest the top five most commonly used hashtags on the platform.

MySQL Workbench

Local instance MySQL80

File Edit View Query Database Server Tools Scripting Help

Navigator

SCHEMAS

Filter objects

Tables

Views

Stored Procedures

Functions

ig_clone

Tables

comments

follows

likes

photo_tags

tags

photos

users

Views

Stored Procedures

Functions

new_db

sakila

sys

world

Administration Schemas

Information

Table: photos

Columns:

id int AI PK

image_url varchar(355)

user_id int

created_at timestamp

Object Info Session

Query 1

Limit to 1000 rows

```

1 • USE ig_clone;
2 • SELECT tags.tag_name, count(*) as total FROM photo_tags
3   JOIN tags
4   ON photo_tags.tag_id = tags.id
5   GROUP BY tags.id
6   ORDER BY total DESC
7   LIMIT 5;
8
9

```

Result Grid

tag_name	total
smile	59
beach	42
party	39
fun	38
concert	24

Result 34

Output

Action Output

#	Time	Action	Message	Duration / Fetch
41	18:21:49	SELECT * FROM photo_tags LIMIT 0, 1000	501 row(s) returned	0.016 sec / 0.000 sec
42	18:23:46	SELECT * FROM photo_tags inner join tags on photo_tags.tag_id = tags.id LIMIT 0, 1000	501 row(s) returned	0.000 sec / 0.000 sec
43	18:35:48	SELECT * FROM tags inner join photo_tags on photo_tags.tag_id = tags.id LIMIT 0, 1000	Error Code: 1066. Not unique table/alias: 'tags'	0.000 sec
44	18:36:02	SELECT * FROM tags inner join photo_tags on photo_tags.tag_id = tags.id LIMIT 0, 1000	501 row(s) returned	0.000 sec / 0.000 sec
45	18:51:35	SELECT tags.tag_name, count(*) as total from photo_tags join tags on photo_tags.i...	5 row(s) returned	0.015 sec / 0.000 sec

SQLAdditions

Automatic context help is disabled. Use the toolbar to manually get help for the current caret position or to toggle automatic help.

5. Ad Campaign Launch: The team wants to know the best day of the week to launch ads. Your Task: Determine the day of the week when most users register on Instagram. Provide insights on when to schedule an ad campaign.

MySQL Workbench

Local instance MySQL80

File Edit View Query Database Server Tools Scripting Help

Navigator

SCHEMAS

Filter objects

Tables

Views

Stored Procedures

Functions

ig_clone

Tables

comments

follows

likes

photo_tags

photos

tags

users

Views

Stored Procedures

Functions

new_db

sakila

sys

world

Administration Schemas

Information

Table: photos

Columns:

id int AI PK

image_url varchar(355)

user_id int

created_at timestamp

Object Info Session

Query 1

Limit to 1000 rows

```

1 • USE ig_clone;
2 • select dayname(created_at) as day_name,
3     count(*) as num_registered
4   from users
5   group by dayname(created_at)
6   order by num_registered desc;
7
8

```

Result Grid

day_name	num_registered
Thursday	16
Sunday	16
Friday	15
Tuesday	14
Monday	14

Result 47

Output

Action Output

#	Time	Action	Message	Duration / Fetch
58	21:58:40	select dayname(created_at) as day_name, count(*) as num_registered from users gr...	7 row(s) returned	0.000 sec / 0.000 sec
59	21:58:40	select dayname(created_at) as day_name, count(*) as num_registered from users gr...	7 row(s) returned	0.000 sec / 0.000 sec
60	21:58:40	select dayname(created_at) as day_name, count(*) as num_registered from users gr...	7 row(s) returned	0.000 sec / 0.000 sec
61	21:58:41	select dayname(created_at) as day_name, count(*) as num_registered from users gr...	7 row(s) returned	0.000 sec / 0.000 sec
62	21:58:41	select dayname(created_at) as day_name, count(*) as num_registered from users gr...	7 row(s) returned	0.000 sec / 0.000 sec

SQLAdditions

Automatic context help is disabled. Use the toolbar to manually get help for the current caret position or to toggle automatic help.

Insights :The insights derived from this analysis will be instrumental in identifying the optimal day of the week for initiating ad campaigns. By pinpointing the day that typically sees the highest number of new user registrations on Instagram, these findings will inform and refine your strategy for scheduling ad campaigns, ensuring they are launched when potential engagement is at its peak for maximal impact.

B) Investor Metrics:

1. User Engagement: Investors want to know if users are still active and posting on Instagram or if they are making fewer posts.

Your Task: Calculate the average number of posts per user on Instagram. Also, provide the total number of photos on Instagram divided by the total number of users.

The screenshot shows the MySQL Workbench interface with a query editor and a results grid. The query is as follows:

```
1 USE ig_clone;
2 SELECT COUNT(*) AS total_photos
3 FROM photos;
4 SELECT COUNT(id) AS total_users
5 FROM users;
6
7 SELECT (SELECT Count(*)
8 FROM photos) / (SELECT Count(*)
9 FROM users) AS avg;
10
11 select user_id, count(*) as posts_count from photos
```

The results grid shows the output of the query:

avg_posts
3.4730

The bottom panel shows the Action Output table with the following columns: #, Time, Action, Message, and Duration / Fetch. The output shows the execution of the query and the results of the calculations.

2. Bots & Fake Accounts: Investors want to know if the platform is crowded with fake and dummy accounts.

Your Task: Identify users (potential bots) who have liked every single photo on the site, as this is not typically possible for a normal user.

The screenshot shows the MySQL Workbench interface with a query editor and a results grid. The query is as follows:

```
11 select user_id, count(*) as posts_count from photos
12 group by user_id
13 order by posts_count desc;
14
15 SELECT AVG(posts_count) as avg_posts
16 FROM (
17 select user_id, count(*) as posts_count from photos
18 group by user_id
19 order by posts_count desc) as user_posts;
20
21
```

The results grid shows the output of the query:

avg_posts
3.4730

The bottom panel shows the Action Output table with the following columns: #, Time, Action, Message, and Duration / Fetch. The output shows the execution of the query and the results of the calculations.

MySQL Workbench

Local instance MySQL80

File Edit View Query Database Server Tools Scripting Help

Navigator

SCHEMAS

Filter objects

Tables

Views

Stored Procedures

Functions

ig_clone

Tables

comments

follows

likes

photo_tags

photos

tags

users

Views

Stored Procedures

Functions

new_db

sakila

sys

world

Administration Schemas

Information

Table: photos

Columns:

id int AI PK

image_url varchar(355)

user_id int

created_dat timestamp

Query 1

```
1 USE ig_clone;
2 select count(*) from photos;
3 select username, count(*) as num_likes
4 from users
5 join likes
6 on users.id = likes.user_id
7 group by likes.user_id
8 having num_likes = (select count(*) from photos);
9
10
```

Result Grid

username	num_likes
Aniya_Hackett	257
Jacyn81	257
Rocio33	257
Maxwell.Halvorson	257
Ollie_Ledner37	257

Result 63

Output

Action Output

#	Time	Action	Message	Duration / Fetch
73	22:44:26	SELECT (SELECT Count(*) FROM photos) / (SELECT Count(*)	1 row(s) returned	0.000 sec / 0.000 sec
74	22:44:26	select user_id, count(*) as posts_count from photos group by user_id order by posts...	74 row(s) returned	0.000 sec / 0.000 sec
75	22:44:26	SELECT AVG(posts_count) as avg_posts FROM (select user_id, count(*) as posts...	1 row(s) returned	0.000 sec / 0.000 sec
76	22:51:02	select username, count(*) as num_likes from users join likes on users.id = likes user...	Error Code: 1064. You have an error in your SQL syntax; check the manual that cor...	0.000 sec
77	22:51:19	select username, count(*) as num_likes from users join likes on users.id = likes user...	13 row(s) returned	0.000 sec / 0.000 sec

SQLAdditions

Automatic context help is disabled. Use the toolbar to manually get help for the current caret position or to toggle automatic help.

MySQL Workbench

Local instance MySQL80

File Edit View Query Database Server Tools Scripting Help

Navigator

SCHEMAS

Filter objects

Tables

Views

Stored Procedures

Functions

ig_clone

Tables

comments

follows

likes

photo_tags

photos

tags

users

Views

Stored Procedures

Functions

new_db

sakila

sys

world

Administration Schemas

Information

Table: photos

Columns:

id int AI PK

image_url varchar(355)

user_id int

created_dat timestamp

Query 1

```
1 USE ig_clone;
2 select count(*) from photos;
3 select username, count(*) as num_likes
4 from users
5 join likes
6 on users.id = likes.user_id
7 group by likes.user_id
8 having num_likes = (select count(*) from photos);
9
10
```

Result Grid

username	num_likes
Duane60	257
Julien_Schmidt	257
Mike_Auer39	257
Nia_Haag	257
Leslie67	257

Result 63

Output

Action Output

#	Time	Action	Message	Duration / Fetch
73	22:44:26	SELECT (SELECT Count(*) FROM photos) / (SELECT Count(*)	1 row(s) returned	0.000 sec / 0.000 sec
74	22:44:26	select user_id, count(*) as posts_count from photos group by user_id order by posts...	74 row(s) returned	0.000 sec / 0.000 sec
75	22:44:26	SELECT AVG(posts_count) as avg_posts FROM (select user_id, count(*) as posts...	1 row(s) returned	0.000 sec / 0.000 sec
76	22:51:02	select username, count(*) as num_likes from users join likes on users.id = likes user...	Error Code: 1064. You have an error in your SQL syntax; check the manual that cor...	0.000 sec
77	22:51:19	select username, count(*) as num_likes from users join likes on users.id = likes user...	13 row(s) returned	0.000 sec / 0.000 sec

SQLAdditions

Automatic context help is disabled. Use the toolbar to manually get help for the current caret position or to toggle automatic help.