

Q1) We want to reward our users who have been around the longest. Find the 5 oldest users.

➤ `SELECT * FROM users ORDER BY created_at LIMIT 5;`

id	username	created_at
80	Mukul	2016-05-06 00:14:21
67	Soujanya	2016-05-06 13:04:30
63	Shivam Sid	2016-05-08 01:30:41
95	Afreen	2016-05-09 17:30:22
38	Nandita	2016-05-14 07:56:26

Q2) What day of the week do most users register on? We need to figure out when to schedule and ad campaign.

➤ `SELECT DAYNAME(created_at) as day, COUNT(created_at) as day_count FROM users GROUP BY day ORDER BY day_count DESC LIMIT 1;`

day	day_count
sunday	17

Q3) We want to target our inactive users with an email campaign. Find the users who have never posted a photo.

➤ `SELECT * FROM users WHERE id NOT IN (SELECT user_id FROM photos);`

id	username	created_at
5	Ria Juneja	2016-12-07 01:04:39
7	Farhan Akhtar	2016-12-12 06:50:08
14	Anurag	2017-02-06 23:29:16
21	Vikas	2017-01-23 11:51:15
24	Marvin	2017-04-18 02:32:44
25	Sandra	2016-10-03 12:49:21
34	Kritika	2016-07-08 21:42:01
36	Lusee	2016-08-04 15:42:20
41	Nikita Sharma	2016-07-17 17:25:45
45	Zakir Khan	2017-02-05 21:23:37
49	Divya	2016-10-30 12:42:31
53	Palak	2017-02-07 07:49:34
54	Durga	2016-12-21 04:43:38
57	Rahul Thakur	2017-02-02 23:12:48
66	Sonal	2016-07-01 17:36:15
68	Rupam Kalita	2016-11-13 20:09:27
71	Anusha	2016-05-14 15:38:50
74	Mariya	2017-01-25 17:17:28
75	Saad	2016-09-21 05:14:01
76	Yasir	2016-07-21 09:26:09
80	Mukul	2016-05-06 00:14:21
81	Pooja Mundra	2017-01-14 17:02:34
83	Lakhan	2016-11-06 02:31:23
89	Seema	2016-09-14 23:47:05
90	Asha	2017-03-03 11:52:27
91	Manjeet	2016-06-03 23:31:53

Q4) We're running a new contest to see who can get the most likes on a single photo. WHO WON?

➤ `SELECT users.id as user_id, users.username as username, photos.id as photo_id, photos.image_url as url, COUNT(likes.user_id) as like_count FROM likes JOIN photos ON likes.photo_id=photos.id JOIN users ON photos.user_id=users.id GROUP BY likes.photo_id ORDER BY like_count DESC LIMIT 1;`

user_id	username	photo_id	url	like_count
52	Nisha	145	https://jarret.name	48

Q5) Our Investors want to know. How many times does the average user post? Total number of photos/total number of users.

- `SELECT ROUND((SELECT COUNT(id) FROM photos)/(SELECT COUNT(id) FROM users),2) as photo_posting_avg_time;`

photo_posting_avg_time
2.57

Q6) User ranking by postings higher to lower?

- `SELECT users.id, users.username, COUNT(photos.id) as photos_posted FROM users JOIN photos ON users.id=photos.user_id GROUP BY users.id ORDER BY photos_posted DESC;`

id	username	photos_posted
23	Sanjeet	12
88	Laxmi	11
59	Sandeep	10
86	Vaishnavi	9
58	Riyanshi Gupta	8
29	Mashad	8
77	Vaibhav	6
33	Priyanka	5
52	Nisha	5
47	Aveek	5
6	Gautam Dhawan	5
13	Amreen Ansari	5
51	Simran	5
78	Uday	5
11	Aditya	5
1	Sachin	5
72	Sunil	5
65	Soni	5
43	Natasha	5
64	Sireesha	5
26	Maryam	5
2	Sarah Khan	4
3	Shatreen Shams	4
9	Abhishek	4
8	Iqbal Ziya	4
46	Neeraj	4
12	Aman Vishen	4
44	Bobby	4
63	Shivam Sid	4
32	Jerin	4
28	Azhar	4
16	Ashutosh	4
15	Apoorv Sharma	4
87	Digvijay	4
4	Isha Negi	3
10	Aditi	3
50	Gungun	3
67	Soujanya	3
17	Yash Vardhan	3
42	Maya	3
92	Ranjeet	3
96	Sofia	3
99	Deepak	3
38	Nandita	2
100	Amal	2
82	Adeeba	2
84	Shivangi	2
85	Mitali	2
60	Sakshi Goel	2
62	Saurabh	2
19	Vinay	2
93	Nia	2
95	Afreen	2
30	Gulafsha	2
97	Ketika Sharma	2
35	Kuldeep	2
31	Inderjeet	1
27	Fajar	1
79	Ronak	1
40	Nikhil	1
61	Sandhya	1
29	Neha	1
22	Suresh	1
20	Vinayak	1
48	Binish Khan	1
18	Wilson	1
37	Mani	1
94	Amrita	1
69	Anandita Kalita	1
70	Prachi	1
55	Priya	1
98	Rohit	1
56	Preeti	1
73	Sunita	1

Q7) Total numbers of users who have posted at least one time.

- `SELECT COUNT(DISTINCT user_id) as user_posted_atleast_once FROM photos;`

user_posted_atleast_once
74

**Q8) A brand wants to know which hashtags to use in a post. What are the top 5 most commonly used hashtags?**

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

tag_id	tag_name	tag_count
21	smile	59
20	beach	42
17	party	39
13	fun	38
18	concert	24

**Q9) We have a small problem with bots on our site. Find users who have liked every single photo on the site and Commented on every single photo.**

- `SELECT users.id as user_id, users.username as username, COUNT(users.id) as total_likes_by_user FROM users JOIN likes ON users.id=likes.user_id GROUP BY users.id HAVING total_likes_by_user = (SELECT COUNT(*) FROM photos);`

user_id	username	total_likes_by_user
5	Ria Juneja	257
14	Anurag	257
21	Vikas	257
24	Marvin	257
36	Lusee	257
41	Nikita Sharma	257
54	Durga	257
57	Rahul Thakur	257
66	Sonal	257
71	Anusha	257
75	Saad	257
76	Yasir	257
91	Manjeet	257

- `SELECT users.id as user_id, users.username as username, COUNT(comments.user_id) as total_comment FROM users JOIN comments ON users.id=comments.user_id GROUP BY users.id HAVING total_comment = (SELECT COUNT(DISTINCT comments.photo_id) FROM comments);`

user_id	username	total_comment
5	Ria Juneja	257
14	Anurag	257
21	Vikas	257
24	Marvin	257
36	Lusee	257
41	Nikita Sharma	257
54	Durga	257
57	Rahul Thakur	257
66	Sonal	257
71	Anusha	257
75	Saad	257
76	Yasir	257
91	Manjeet	257

**Q10) We also have a problem with celebrities. Find users who have never commented on a photo.**

- `SELECT users.id as user_id, users.username as username, comments.comment_text FROM users LEFT JOIN comments ON users.id=comments.user_id WHERE comments.comment_text IS NULL;`

user_id	username	comment_text
1	Sachin	NULL
7	Farhan Akhtar	NULL
23	Sanjeet	NULL
25	Sandra	NULL
29	Mashad	NULL
34	Kritika	NULL
45	Zakir Khan	NULL
49	Divya	NULL
51	Simran	NULL
53	Palak	NULL
58	Riyanshi Gupta	NULL
59	Sandeep	NULL
64	Sireesha	NULL
68	Rupam Kalita	NULL
74	Mariya	NULL
77	Vaibhav	NULL
80	Mukul	NULL
81	Pooja Mundra	NULL
83	Lakhan	NULL
86	Vaishnavi	NULL
88	Laxmi	NULL
89	Seema	NULL
90	Asha	NULL