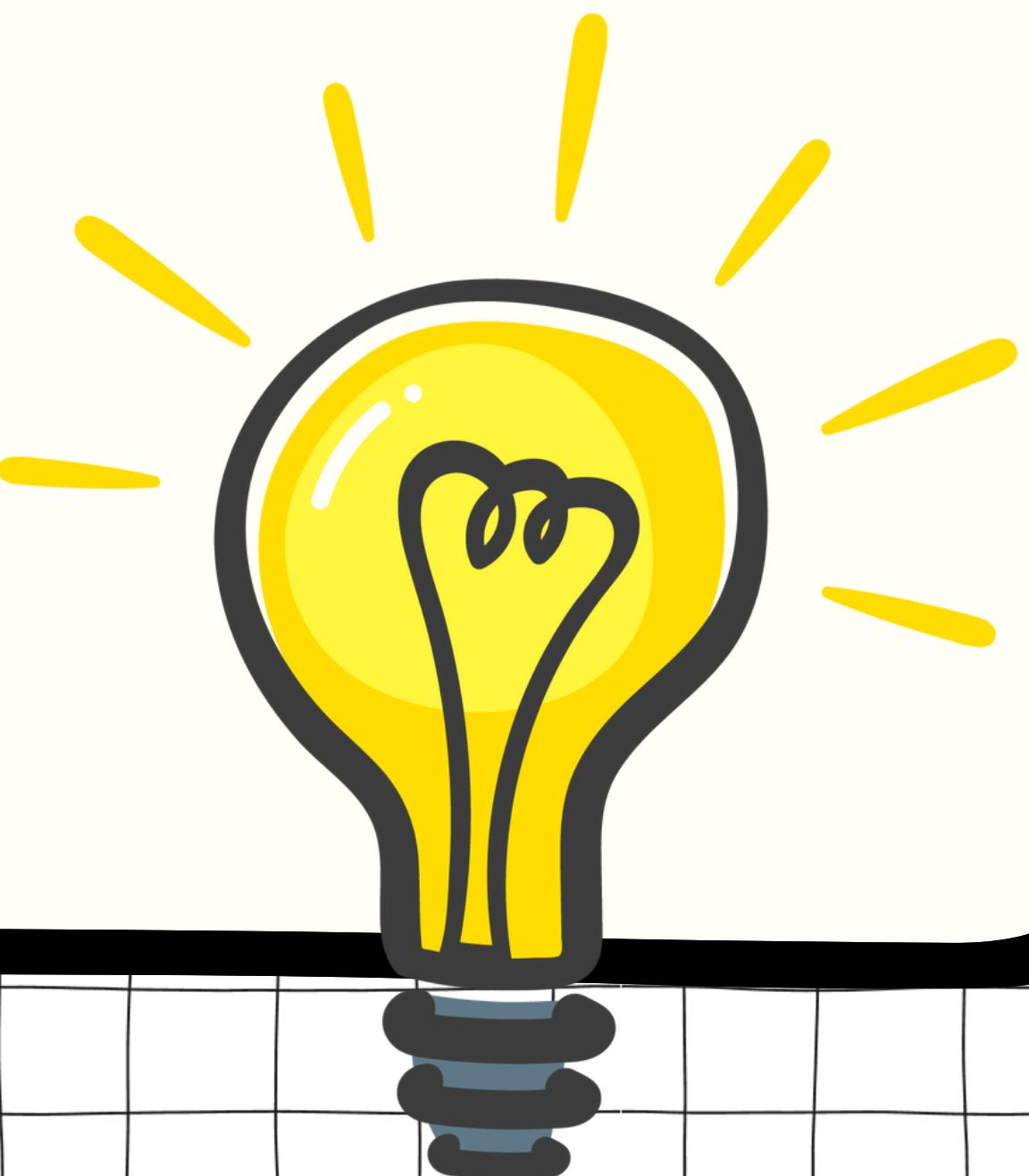


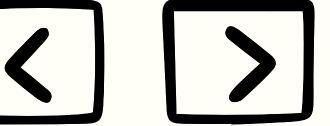
SQL Project



-by Satish Singh



Contents



01

User Table

02

Logins Table

03

Query-1

04

Query-2

05

Query-3

06

Query-4

07

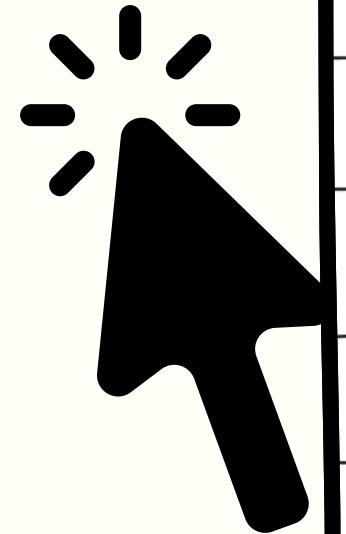
Query-5

08

Query-6

09

Query-7



User Table

select * from users

	USER_ID	USER_NAME	USER_STATUS
1	1	Alice	Active
2	2	Bob	Inactive
3	3	Charlie	Active
4	4	David	Active
5	5	Eve	Inactive
6	6	Frank	Active
7	7	Grace	Inactive
8	8	Heidi	Active
9	9	Ivan	Inactive
10	10	Judy	Active

Logins Table

`select * from logins`

	USER_ID	LOGIN_TIMESTAMP	SESSION_ID	SESSION_SCORE
1	1	2023-07-15 09:30:00.000	1001	85
2	2	2023-07-22 10:00:00.000	1002	90
3	3	2023-08-10 11:15:00.000	1003	75
4	4	2023-08-20 14:00:00.000	1004	88
5	5	2023-09-05 16:45:00.000	1005	82
6	6	2023-10-12 08:30:00.000	1006	77
7	7	2023-11-18 09:00:00.000	1007	81
8	8	2023-12-01 10:30:00.000	1008	84
9	9	2023-12-15 13:15:00.000	1009	79
10	10	2024-06-25 15:00:00.000	1010	92
11	1	2024-01-10 07:45:00.000	1011	86
12	2	2024-01-25 09:30:00.000	1012	89
13	3	2024-02-05 11:00:00.000	1013	78
14	4	2024-03-01 14:30:00.000	1014	91
15	5	2024-03-15 16:00:00.000	1015	83
16	6	2024-04-12 08:00:00.000	1016	80
17	7	2024-05-18 09:15:00.000	1017	82
18	8	2024-05-28 10:45:00.000	1018	87
19	9	2024-06-15 13:30:00.000	1019	76
20	10	2024-06-26 15:45:00.000	1020	93
21	10	2024-06-27 15:00:00.000	1021	92
22	10	2024-06-28 15:45:00.000	1022	93
23	1	2024-01-10 07:45:00.000	1101	86
24	3	2024-01-25 09:30:00.000	1102	89
25	5	2024-01-15 11:00:00.000	1103	78
26	2	2023-11-10 07:45:00.000	1201	82
27	4	2023-11-25 09:30:00.000	1202	84
28	6	2023-11-15 11:00:00.000	1203	80

Query 1

Which users did not log in during the past 5 months?

```
with cte as  
(select user_id,  
month(max(login_timestamp)) as last_login,  
month(getdate()) as present_date  
from logins  
group by user_id)
```

```
select s.user_name from users s join cte c  
on s.user_id = c.user_id  
where (c.present_date-c.last_login >= 5);
```

Query 2

How many users and sessions were there in each quarter, ordered from newest to oldest?

```
select DATEPART(quarter, LOGIN_TIMESTAMP) as  
quarter_number,COUNT(*) as session_cnt,  
COUNT(distinct USER_ID) as user_cnt  
,DATETRUNC(quarter,MIN(LOGIN_TIMESTAMP))  
as first_quarter_date  
from logins  
group by DATEPART(quarter, LOGIN_TIMESTAMP)
```

Query 3

Which users logged in during January 2024 but did not log in during November 2023?

```
select distinct user_id from logins  
where LOGIN_TIMESTAMP between '2024-01-01' and  
'2024-01-31' AND user_id NOT IN  
(select user_id from logins where LOGIN_TIMESTAMP  
between '2023-11-01' and '2023-11-30')
```

Query 4

What is the percentage change in sessions from the last quarter?

```
with cte as
(select COUNT(*) as session_cnt,
COUNT(distinct USER_ID) as user_cnt
,DATETRUNC(quarter,MIN(LOGIN_TIMESTAMP))as first_quarter_date
from logins
group by DATEPART(quarter, LOGIN_TIMESTAMP)
)
select*, LAG(session_cnt) over(order by first_quarter_date) as prev_session_cnt
,(session_cnt-(LAG(session_cnt)over(order by
first_quarter_date)))*100.0/(LAG(session_cnt) over(order by
first_quarter_date)) as percentage_change
from cte
```

Query 5

Which user had the highest session score each day?

```
with cte as(  
select user_id, CAST(LOGIN_TIMESTAMP as date) as  
login_date  
, SUM(session_score) as score  
from logins  
group by user_id, CAST(LOGIN_TIMESTAMP as date)  
--order by CAST(LOGIN_TIMESTAMP as date),score  
)  
select * from (  
select * , ROW_NUMBER() over(partition by login_date  
order by score desc)as rn  
from cte) a  
where rn=1
```

Query 6

• **Which users have had a session every single day since their first login?**

```
with cte as (
select user_id, CAST(LOGIN_TIMESTAMP as date) as date
from logins
group by user_id, CAST(LOGIN_TIMESTAMP as date))

select user_id, min(CAST(LOGIN_TIMESTAMP as date)) as
first_login
,DATEDIFF(day, min(CAST(LOGIN_TIMESTAMP as
date)),GETDATE())-38 as no_of_login_days_required
,COUNT(distinct CAST(LOGIN_TIMESTAMP as date)) as
no_of_login_days
from logins
group by user_id
having (DATEDIFF(day, min(CAST(LOGIN_TIMESTAMP as
date)),GETDATE())-38)=(COUNT(distinct
CAST(LOGIN_TIMESTAMP as date)))
order by user_id
```

Query 7

On what dates were there no logins at all?

```
select cal_date  
from calendar_dim c  
INNER JOIN (select  
cast(min(LOGIN_TIMESTAMP) as date) as  
first_date, cast(max(LOGIN_TIMESTAMP) as  
date) as last_date  
from logins) a on c.cal_date between first_date  
and last_date  
where cal_date not in  
(select distinct cast(LOGIN_TIMESTAMP as  
date) from logins)
```

Thank you



Satish Singh

