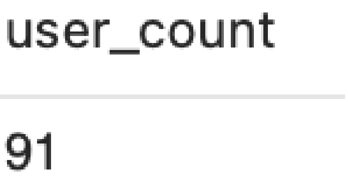
1) Find the number of users who organized a meeting and who were invited to another meeting (regardless of the timestamp of the meeting)

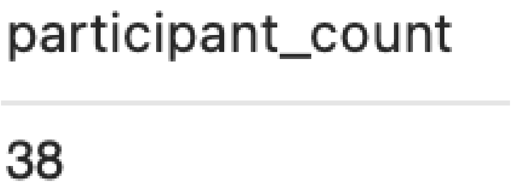
```
select count(distinct organizer_id) as user_count
from dim_meetings
where organizer_id in
(select participant_id
from fact_participations);
```



user_count
91

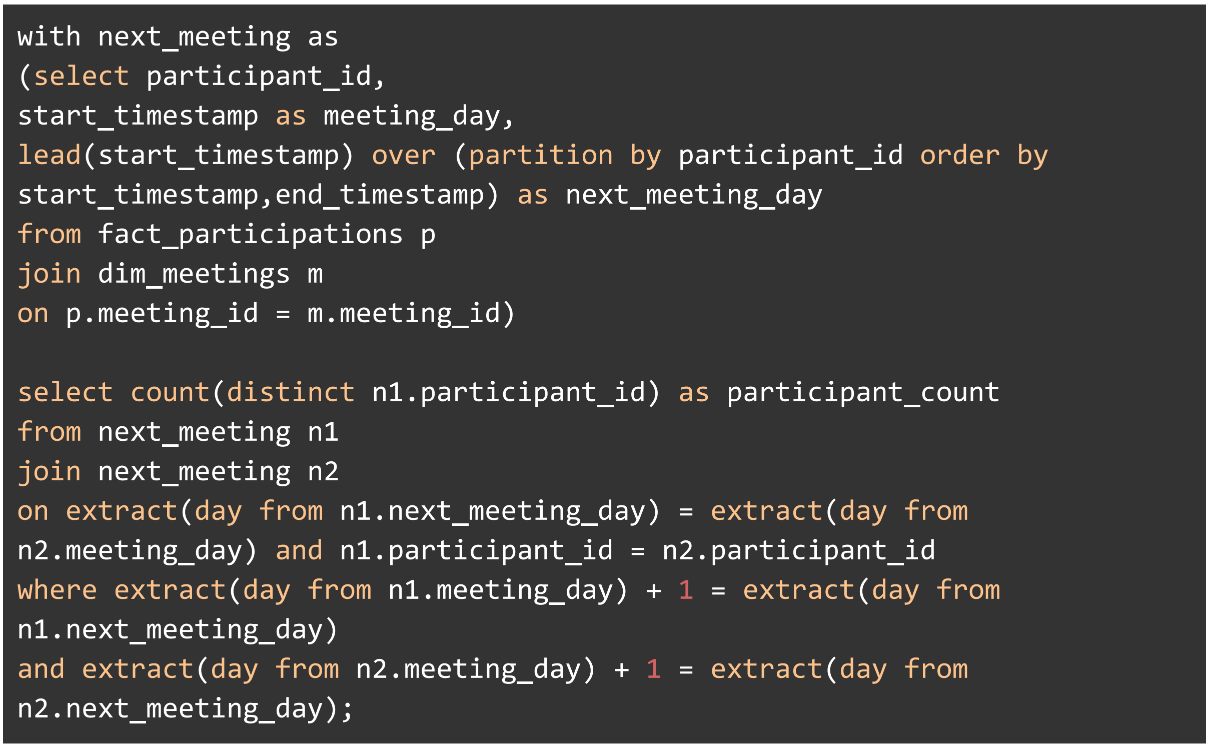
2) Find the number of users who organized a meeting and who were invited to another meeting whose timestamp overlaps with the organized meeting timestamp

```
with conflicting_meetings as (
select distinct fact_participations.participant_id
from fact_participations
join dim_meetings organized
on fact_participations.meeting_id = organized.meeting_id

join dim_meetings invited
on fact_participations.participant_id = invited.organizer_id

where organized.start_timestamp >= invited.start_timestamp
and organized.start_timestamp < invited.end_timestamp
or organized.end_timestamp > invited.start_timestamp
and organized.end_timestamp <= invited.end_timestamp)

select count(participant_id) as participant_count
from conflicting_meetings;
```

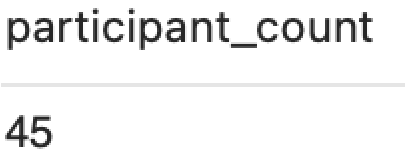


participant_count
38

3) Find the number of users invited to meetings on three consecutive days

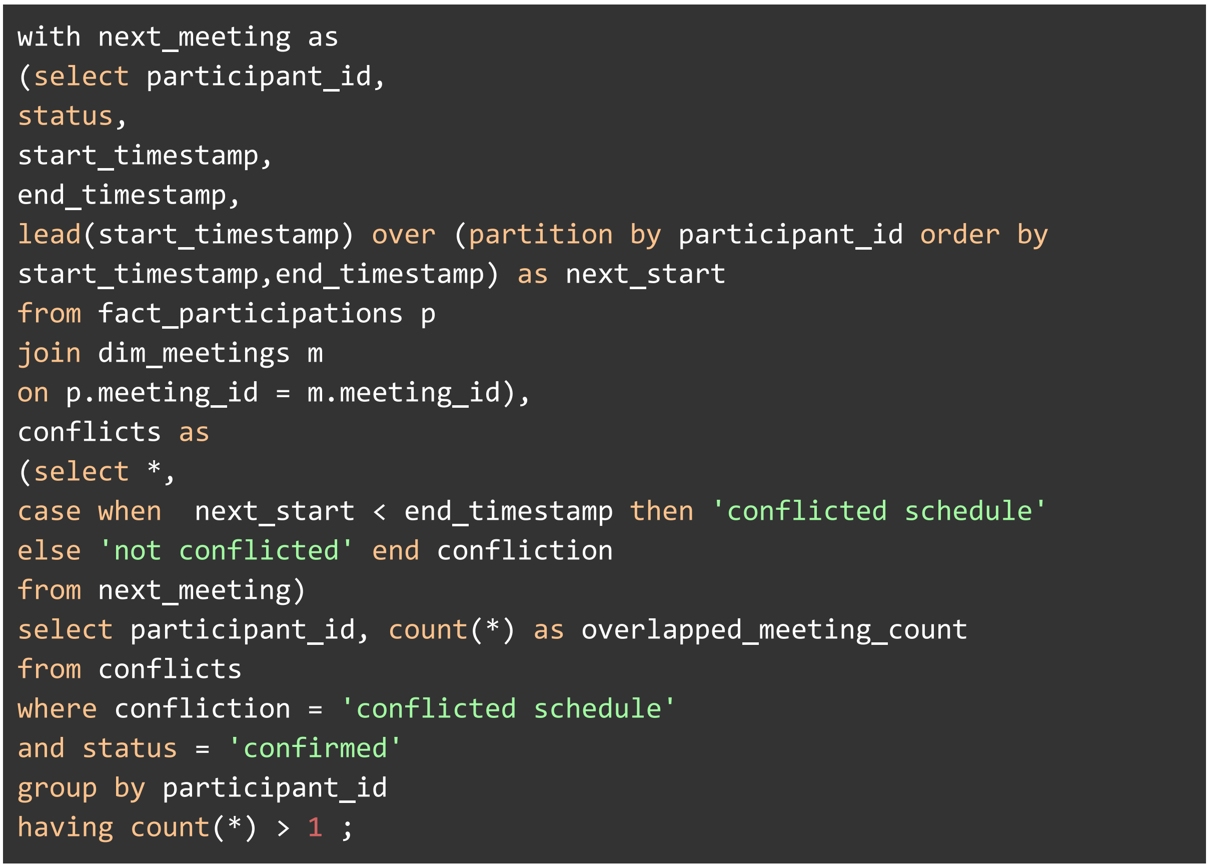


```
with next_meeting as
(select participant_id,
start_timestamp as meeting_day,
lead(start_timestamp) over (partition by participant_id order by
start_timestamp,end_timestamp) as next_meeting_day
from fact_participations p
join dim_meetings m
on p.meeting_id = m.meeting_id)

select count(distinct n1.participant_id) as participant_count
from next_meeting n1
join next_meeting n2
on extract(day from n1.next_meeting_day) = extract(day from
n2.meeting_day) and n1.participant_id = n2.participant_id
where extract(day from n1.meeting_day) + 1 = extract(day from
n1.next_meeting_day)
and extract(day from n2.meeting_day) + 1 = extract(day from
n2.next_meeting_day);
```

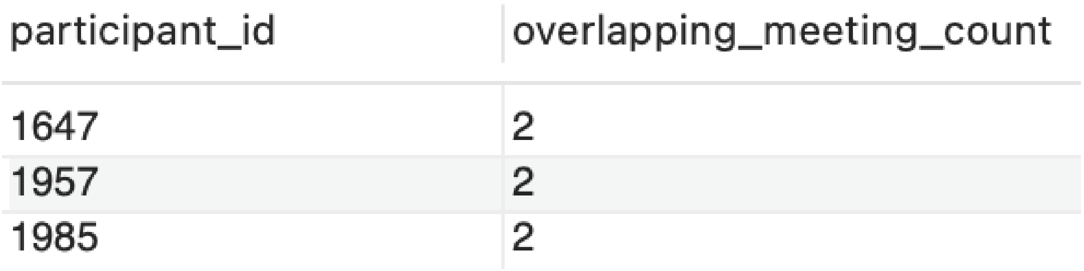


participant_count
45

4) Find the participant_id of the users invited to overlapping meetings and who have confirmed their participation in at least two disjoint meetings.



```
with next_meeting as
(select participant_id,
status,
start_timestamp,
end_timestamp,
lead(start_timestamp) over (partition by participant_id order by
start_timestamp,end_timestamp) as next_start
from fact_participations p
join dim_meetings m
on p.meeting_id = m.meeting_id),
conflicts as
(select *,
case when  next_start < end_timestamp then 'conflicted schedule'
else 'not conflicted' end confliction
from next_meeting)
select participant_id, count(*) as overlapped_meeting_count
from conflicts
where confliction = 'conflicted schedule'
and status = 'confirmed'
group by participant_id
having count(*) > 1 ;
```



participant_id	overlapping_meeting_count
1647	2
1957	2
1985	2