Case Study : Mobile App User Behavior Analysis

The dataset consists of two tables: User1 and User2. Below is a brief description of the schema.

User1
- user_id: Unique Id of a user.
- country: Country where the users installed the app.
- Platform: Mobile platform the users use to operate the app.
- user_first_seen_date: Date of installing the app.

User2
- user_id: Unique Id of a user.
- Date: Date on which the user was using the app.
- sessionID: Id of a session (30 min. of activity counts as one session for a user).

Questions to Answer

1. Find the 3-day, 7-day, and overall running sum of the number of unique sessions every day for every "country" and "platform."**
2. Calculate the average number of days users use the app.**
3. Calculate the retention percentage for Day-1, Day-3, and Day-7.**

Q1. Find the 3-day, 7-day and overall running sum of the number of unique sessions every day for every “country” and “platform”.

Query-

```
with t1 as
 (select
    u1.country,
    u1.platform,
    u2.date,
    count(distinct u2.sessionid) as distinct_session
 from
    statfinity_sql_case.user_1 u1
 join
    statfinity_sql_case.user_2 u2
 on
    u1.user_id = u2.user_id
 group by
    u1.country,
    u1.platform,
    u2.date
 order by
    u1.country,
    u1.platform,
    u2.date)
select
    country,
    platform,
    date,
 sum(distinct_session ) over(partition by country, platform order by date rows between 2 preceding
and current row) as three_day_running_sum,
 sum(distinct_session ) over(partition by country, platform order by date rows between 6 preceding
and current row) as seven_day_running_sum,
 sum(distinct_session ) over(partition by country, platform order by date) as overall_running_sum
from t1;
```

Q2. Average number of days users use the app.

Query-

```
select
  round(avg(activedays),2) as avgdays,
  round(avg(activedays)) as roundedavgdays
from
   (select
      user_id,
      count(distinct date) as activedays
   from
      statfinity_sql_case.user_2
   group by
      user_id);
```

Q3. Retention percentage for Day-1, Day-3 and Day-7.

Query-

```
with t1 as
   (select
      u1.user_id,
      u1.user_first_seen_date,
      u2.date,
      date_diff(u2.date, u1.user_first_seen_date, DAY) AS day_difference, u2.sessionID
   from
       statfinity_sql_case.user_1 u1
   left join
       statfinity_sql_case.user_2 u2
   on
      u1.user_id = u2.user_id),

day1retention as
    (select
        count(distinct user_id) as day_1_retention
    from t1
    where
        day_difference = 1),
day3retention as
    (select
        count(distinct user_id) as day_3_retention
    from t1
    where
        day_difference = 3),
day7retention as
    (select
        count(distinct user_id) as day_7_retention
    from t1
    where
        day_difference = 7)
```

```
select
   round((day_1_retention/(select count(distinct user_id) from t1)),4) * 100 as
day_1_retention_percent,
   round((day_3_retention/(select count(distinct user_id) from t1)),4) * 100 as
day_3_retention_percent,
   round((day_7_retention/(select count(distinct user_id) from t1)),4) * 100 as
day_7_retention_percent
from day1retention, day3retention, day7retention;
```