Case: Socialgram

Qn	Query

1	SELECT
	   name
	FROM
	   `user`
	WHERE
	   `UserID` IN (SELECT
	       UserID
	     FROM
	       `user-group`
	     WHERE
	      isModerator = 'Y');

2	SELECT name FROM `user` LEFT JOIN `user-group` ON `user`.UserID = `user-group`.UserID WHERE `user-group`.isBanned = 'Y' GROUP BY `user`.UserID HAVING COUNT(DISTINCT `user-group`.GroupID) > 1;
3	SELECT user.name, SUM(photo.filesize) AS `Total_Photo_Filesize` FROM user LEFT JOIN `user-photo` ON user.UserID = `user-photo`.UserID LEFT JOIN photo ON `user-photo`.PhotoID = photo.PhotoID GROUP BY user.UserID HAVING `Total_Photo_Filesize` > 1000;
4	SELECT AlbumName, COUNT(DISTINCT UserID) AS `num_users` FROM album GROUP BY albumID , PhotoID HAVING `num_users` >= 2;
5	SELECT mutual_following.UserID, user_a.name, mutual_following.FollowingUserID, user_b.name FROM (SELECT following_a.UserID, following_a.FollowingUserID FROM `User-Following` following_a LEFT JOIN `User-Following` following_b ON following_a.FollowingUserID =

```
                   following_b.UserID
                        AND following_a.UserID = following_b.FollowingUserID
                     WHERE
                       following_b.UserID IS NOT NULL) mutual_following
                        LEFT JOIN
                     user user_a ON mutual_following.UserID = user_a.UserID
                        LEFT JOIN
                     user user_b ON mutual_following.FollowingUserID = user_b.UserID
                   ORDER BY mutual_following.UserID ASC;
```

6	SELECT

```
SELECT
  user_a.name, user_b.name, user_c.name
FROM
  (SELECT
     following_a.UserID AS a_ID,
        following_b.UserID AS b_ID,
        following_c.UserID AS c_ID
   FROM
     `User-Following` following_a
   LEFT JOIN `User-Following` following_b ON following_a.FollowingUserID =
following_b.UserID
   LEFT JOIN `User-Following` following_c ON following_b.FollowingUserID =
following_c.UserID
        AND following_c.FollowingUserID = following_a.UserID
   WHERE
     following_c.UserId IS NOT NULL) relational_triads
        LEFT JOIN
  user user_a ON relational_triads.a_ID = user_a.UserID
        LEFT JOIN
  user user_b ON relational_triads.b_ID = user_b.UserID
        LEFT JOIN
  user user_c ON relational_triads.c_ID = user_c.UserID
ORDER BY relational_triads.a_ID ASC;
```

Case: The Steps Challenge

Qn	Query
| 1 | ```
SELECT
   Name
FROM
   participant;
``` |
| 2 | ```
SELECT
   participant.Name
FROM
   `participant-weeklychallenge`
      LEFT JOIN
   participant ON `participant-weeklychallenge`.ParticipantID = participant.ParticipantID
GROUP BY `participant-weeklychallenge`.ParticipantID
HAVING COUNT(DISTINCT WeeklyChallengeID) > 2;
``` |
| 3 | ```
SELECT
   COUNT(DISTINCT ParticipantID) AS `PartAmt`
FROM
   redemption
WHERE
   RewardID IN (SELECT
         RewardID
      FROM
         reward
      WHERE
         RewardName = 'Car Wash at Caltex');
``` |
| 4 | ```
SELECT
   Name
FROM
   participant
WHERE
   ParticipantID IN (SELECT
         participantID
      FROM
         SCPoint
      GROUP BY participantID
      HAVING SUM(BasicPointsEarned) > 10000)
      AND participantID IN (SELECT DISTINCT
         ParticipantID
      FROM
         redemption
      WHERE
         RewardID IN (SELECT
               RewardID
            FROM
               reward
            WHERE
               RewardName = 'Car Wash at Caltex'));
``` |