Medium

Table: Activity

```
+--------------+---------+
| Column Name  | Type    |
+--------------+---------+
| player_id    | int     |
| device_id    | int     |
| event_date   | date    |
| games_played | int     |
+--------------+---------+
```

(player_id, event_date) is the primary key (combination of columns with unique values) of this table.
This table shows the activity of players of some games.
Each row is a record of a player who logged in and played a number of games (possibly 0) before logging out on someday using some device.

Write a solution to report the **fraction** of players that logged in again on the day after the day they first logged in, **rounded to 2 decimal places**. In other words, you need to count the number of players that logged in for at least two consecutive days starting from their first login date, then divide that number by the total number of players.

The result format is in the following example.

Example 1:

Input:
Activity table:
```
+-----------+-----------+------------+--------------+
| player_id | device_id | event_date | games_played |
+-----------+-----------+------------+--------------+
| 1         | 2         | 2016-03-01 | 5            |
| 1         | 2         | 2016-03-02 | 6            |
| 2         | 3         | 2017-06-25 | 1            |
| 3         | 1         | 2016-03-02 | 0            |
| 3         | 4         | 2018-07-03 | 5            |
+-----------+-----------+------------+--------------+
```
Output:
```
+-----------+
| fraction  |
+-----------+
| 0.33      |
+-----------+
```
Explanation:
Only the player with id 1 logged back in after the first day he had logged in so the answer is 1/3 = 0.33


```sql
# Write your MySQL query statement below
SELECT
  ROUND(COUNT(DISTINCT player_id) / (SELECT COUNT(DISTINCT player_id) FROM Activity), 2)
AS fraction
FROM Activity
WHERE (player_id, DATE_SUB(event_date, INTERVAL 1 DAY))
  IN (SELECT player_id, MIN(event_date) AS first_login FROM Activity GROUP BY player_id);
```