180. Consecutive Numbers

Table: Logs

```
+-------------+---------+
| Column Name | Type    |
+-------------+---------+
| id          | int     |
| num         | varchar |
+-------------+---------+
```

In SQL, id is the primary key for this table.
id is an autoincrement column.

Find all numbers that appear at least three times consecutively.
Return the result table in **any order**.
The result format is in the following example.

Example 1:
Input:
Logs table:

```
+----+-----+
| id | num |
+----+-----+
| 1  | 1   |
| 2  | 1   |
| 3  | 1   |
| 4  | 2   |
| 5  | 1   |
| 6  | 2   |
| 7  | 2   |
+----+-----+
```

Output:

```
+-----------------+
| ConsecutiveNums |
+-----------------+
| 1               |
+-----------------+
```

Explanation: 1 is the only number that appears consecutively for at least three times.

```
# Write your MySQL query statement below
select l1.num as ConsecutiveNums from Logs l1 join Logs l2 on l1.id+1 = l2.id and l1.num =       l2.num
 join Logs l3 on l2.id+1=l3.id and l2.num = l3.num
 group by l1.num
```

 # Using window functions:
Window functions allow you to perform calculations across a set of table rows that are somehow related to the current row. In this case, you can use the **LEAD** or **LAG** window functions to look at the subsequent or preceding rows without using self-join

```
WITH CTE AS (

SELECT
Num,
LEAD(Num,1) OVER (order by Id) as l1,
LEAD(Num, 2) OVER (order by Id) as l2
FROM
Logs )

SELECT distinct Num as ConsecutiveNums
FROM CTE
WHERE num = l1 and num=l2 ;
```