Q1 Table: Person

+-------------+---------+

| Column Name | Type |

+-------------+---------+

| personId | int |

| lastName | varchar |

| firstName | varchar |

+-------------+---------+

personId is the primary key column for this table.

This table contains information about the ID of some persons and their first and last names.

Table: Address

+-------------+---------+

| Column Name | Type |

+-------------+---------+

| addressId | int |

| personId | int |

| city | varchar |

| state | varchar |

+-------------+---------+

addressId is the primary key column for this table.

Each row of this table contains information about the city and state of one person with ID = PersonId.

Write an SQL query to report the first name, last name, city, and state of each person in the Person table. If the address of a personId is not present in the Address table, report null instead.

Return the result table in any order.

Q2 Table: Employee

+-------------+------+

| Column Name | Type |

+-------------+------+

| id | int |

| salary | int |

+-------------+------+

id is the primary key column for this table.

Each row of this table contains information about the salary of an employee.

Write an SQL query to report the second highest salary from the Employee table. If there is no second highest salary, the query should report null.

The query result format is in the following example.

Q3 Table: Employee

+-------------+------+

| Column Name | Type |

+-------------+------+

| id | int |

| salary | int |

+-------------+------+

id is the primary key column for this table.

Each row of this table contains information about the salary of an employee.

Write an SQL query to report the nth highest salary from the Employee table. If there is no nth highest salary, the query should report null.

The query result format is in the following example.

Q4 SQL Schema

Table: Scores

+-------------+---------+

| Column Name | Type |

+-------------+---------+

| id | int |

| score | decimal |

+-------------+---------+

id is the primary key for this table.

Each row of this table contains the score of a game. Score is a floating point value with two decimal places.

Write an SQL query to rank the scores. The ranking should be calculated according to the following rules:

* The scores should be ranked from the highest to the lowest.
* If there is a tie between two scores, both should have the same ranking.
* After a tie, the next ranking number should be the next consecutive integer value. In other words, there should be no holes between ranks.

Return the result table ordered by score in descending order.

The query result format is in the following example.

Q5 SQL Schema

Table: Logs

+-------------+---------+

| Column Name | Type |

+-------------+---------+

| id | int |

| num | varchar |

+-------------+---------+

id is the primary key for this table.

id is an autoincrement column.

Write an SQL query to find all numbers that appear at least three times consecutively.

Return the result table in **any order**.

The query 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.