2408. Design SQL

Premium

Medium
Topics
Companies
Hint

You are given two string arrays, `names` and `columns`, both of size `n`. The `i`th table is represented by the name `names[i]` and contains `columns[i]` number of columns.

You need to implement a class that supports the following **operations**:

- **Insert** a row in a specific table with an id assigned using an *auto-increment* method, where the id of the first inserted row is 1, and the id of each *new* row inserted into the same table is **one greater** than the id of the **last inserte**
- **Remove** a row from a specific table. Removing a row **does not** affect the id of the next inserted row.
- **Select** a specific cell from any table and return its value.
- **Export** all rows from any table in csv format.

Implement the `SQL` class:

- `SQL(String[] names, int[] columns)`
  - Creates the `n` tables.
- `bool ins(String name, String[] row)`
  - Inserts `row` into the table `name` and returns `true`.
  - If `row.length` **does not** match the expected number of columns, or `name` is **not** a valid table, returns `false` without any insertion.
- `void rmv(String name, int rowId)`
  - Removes the row `rowId` from the table `name`.
  - If `name` is **not** a valid table or there is no row with id `rowId`, no removal is performed.
- `String sel(String name, int rowId, int columnId)`
  - Returns the value of the cell at the specified `rowId` and `columnId` in the table `name`.
  - If `name` is **not** a valid table, or the cell `(rowId, columnId)` is **invalid**, returns `"<null>"`.
- `String[] exp(String name)`
  - Returns the rows present in the table `name`.
  - If name is **not** a valid table, returns an empty array. Each row is represented as a string, with each cell value (**including** the row's id) separated by a `","`.

Example 1:

Input:

```
["SQL","ins","sel","ins","exp","rmv","sel","exp"]
[[["one","two","three"],[2,3,1]],["two",["first","second","third"]],["two",1,3],["two",["fourth","fifth","sixth"]],["two"],["two",1],["two",2,2],["two"]]
```

Output: