Table: Department

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

| Column Name | Type |

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

| id | int |

| revenue | int |

| month | varchar |

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

(id, month) is the primary key of this table.

The table has information about the revenue of each department per month.

The month has values in ["Jan","Feb","Mar","Apr","May","Jun","Jul","Aug","Sep","Oct","Nov","Dec"].

Write an SQL query to reformat the table such that there is a department id column and a revenue column **for each month**.

The query result format is in the following example:

Department table:

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

| id | revenue | month |

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

| 1 | 8000 | Jan |

| 2 | 9000 | Jan |

| 3 | 10000 | Feb |

| 1 | 7000 | Feb |

| 1 | 6000 | Mar |

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

Result table:

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

| id | Jan\_Revenue | Feb\_Revenue | Mar\_Revenue | ... | Dec\_Revenue |

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

| 1 | 8000 | 7000 | 6000 | ... | null |

| 2 | 9000 | null | null | ... | null |

| 3 | null | 10000 | null | ... | null |

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

Note that the result table has 13 columns (1 for the department id + 12 for the months).

# Write your MySQL query statement below

select id,

sum(case when month = 'Jan' then revenue else null end) as Jan\_Revenue,

sum(case when month = 'Feb' then revenue else null end) as Feb\_Revenue,

sum(case when month = 'Mar' then revenue else null end) as Mar\_Revenue,

sum(case when month = 'Apr' then revenue else null end) as Apr\_Revenue,

sum(case when month = 'May' then revenue else null end) as May\_Revenue,

sum(case when month = 'Jun' then revenue else null end) as Jun\_Revenue,

sum(case when month = 'Jul' then revenue else null end) as Jul\_Revenue,

sum(case when month = 'Aug' then revenue else null end) as Aug\_Revenue,

sum(case when month = 'Sep' then revenue else null end) as Sep\_Revenue,

sum(case when month = 'Oct' then revenue else null end) as Oct\_Revenue,

sum(case when month = 'Nov' then revenue else null end) as Nov\_Revenue,

sum(case when month = 'Dec' then revenue else null end) as Dec\_Revenue

from department

group by id

order by id;

select id,

max(case when month = 'Jan' then revenue else null end) as Jan\_Revenue,

max(case when month = 'Feb' then revenue else null end) as Feb\_Revenue,

max(case when month = 'Mar' then revenue else null end) as Mar\_Revenue,

max(case when month = 'Apr' then revenue else null end) as Apr\_Revenue,

max(case when month = 'May' then revenue else null end) as May\_Revenue,

max(case when month = 'Jun' then revenue else null end) as Jun\_Revenue,

max(case when month = 'Jul' then revenue else null end) as Jul\_Revenue,

max(case when month = 'Aug' then revenue else null end) as Aug\_Revenue,

max(case when month = 'Sep' then revenue else null end) as Sep\_Revenue,

max(case when month = 'Oct' then revenue else null end) as Oct\_Revenue,

max(case when month = 'Nov' then revenue else null end) as Nov\_Revenue,

max(case when month = 'Dec' then revenue else null end) as Dec\_Revenue

from department

group by id

order by id;

SELECT

id,

MAX(IF(month = 'Jan', revenue, null)) AS Jan\_Revenue,

MAX(IF(month = 'Feb', revenue, null)) AS Feb\_Revenue,

MAX(IF(month = 'Mar', revenue, null)) AS Mar\_Revenue,

MAX(IF(month = 'Apr', revenue, null)) AS Apr\_Revenue,

MAX(IF(month = 'May', revenue, null)) AS May\_Revenue,

MAX(IF(month = 'Jun', revenue, null)) AS Jun\_Revenue,

MAX(IF(month = 'Jul', revenue, null)) AS Jul\_Revenue,

MAX(IF(month = 'Aug', revenue, null)) AS Aug\_Revenue,

MAX(IF(month = 'Sep', revenue, null)) AS Sep\_Revenue,

MAX(IF(month = 'Oct', revenue, null)) AS Oct\_Revenue,

MAX(IF(month = 'Nov', revenue, null)) AS Nov\_Revenue,

MAX(IF(month = 'Dec', revenue, null)) AS Dec\_Revenue

FROM Department

GROUP BY id

SELECT

id,

sum(IF(month = 'Jan', revenue, null)) AS Jan\_Revenue,

sum(IF(month = 'Feb', revenue, null)) AS Feb\_Revenue,

sum(IF(month = 'Mar', revenue, null)) AS Mar\_Revenue,

sum(IF(month = 'Apr', revenue, null)) AS Apr\_Revenue,

sum(IF(month = 'May', revenue, null)) AS May\_Revenue,

sum(IF(month = 'Jun', revenue, null)) AS Jun\_Revenue,

sum(IF(month = 'Jul', revenue, null)) AS Jul\_Revenue,

sum(IF(month = 'Aug', revenue, null)) AS Aug\_Revenue,

sum(IF(month = 'Sep', revenue, null)) AS Sep\_Revenue,

sum(IF(month = 'Oct', revenue, null)) AS Oct\_Revenue,

sum(IF(month = 'Nov', revenue, null)) AS Nov\_Revenue,

sum(IF(month = 'Dec', revenue, null)) AS Dec\_Revenue

FROM Department

GROUP BY id

MAX vs SUM

It seems the test data does not have duplicate data, so you can either use MAX or SUM and the solution will work.

For a specific id, since the input data does not have duplicates, we only have 1 Jan, 1 Feb, 1 Mar, etc., so whether we use MAX or SUM, the column will just display that revenue.

The MAX lets us combine each revenue with null and selects that 1 revenue for the cell.