# Reference: <https://www.datageekinme.com/projects/leetcode-sql-solutions/>

# All the codes are run in Zepl notebook

# 579. Find Cumulative Salary of an Employee

The Employee table holds the salary information in a year.

Write a SQL to get the cumulative sum of an employee’s salary over a period of 3 months but exclude the most recent month.

The result should be displayed by ‘Id’ ascending, and then by ‘Month’ descending.

Example  
Input

| **Id** | **Month** | **Salary** |
| --- | --- | --- |
| 1 | 1 | 20 |
| 2 | 1 | 20 |
| 1 | 2 | 30 |
| 2 | 2 | 30 |
| 3 | 2 | 40 |
| 1 | 3 | 40 |
| 3 | 3 | 60 |
| 1 | 4 | 60 |
| 3 | 4 | 70 |

Output

| **Id** | **Month** | **Salary** |
| --- | --- | --- |
| 1 | 3 | 90 |
| 1 | 2 | 50 |
| 1 | 1 | 20 |
| 2 | 1 | 20 |
| 3 | 3 | 100 |
| 3 | 2 | 40 |

Explanation  
Employee ‘1’ has 3 salary records for the following 3 months except the most recent month ‘4’: salary 40 for month ‘3’, 30 for month ‘2’ and 20 for month ‘1’  
So the cumulative sum of salary of this employee over 3 months is 90(40+30+20), 50(30+20) and 20 respectively.

| **Id** | **Month** | **Salary** |
| --- | --- | --- |
| 1 | 3 | 90 |
| 1 | 2 | 50 |
| 1 | 1 | 20 |

Employee ‘2’ only has one salary record (month ‘1’) except its most recent month ‘2’.

| **Id** | **Month** | **Salary** |
| --- | --- | --- |
| 2 | 1 | 20 |

Employ ‘3’ has two salary records except its most recent pay month ‘4’: month ‘3’ with 60 and month ‘2’ with 40. So the cumulative salary is as following.

| **Id** | **Month** | **Salary** |
| --- | --- | --- |
| 3 | 3 | 100 |
| 3 | 2 | 40 |

FINISHED

Took 0 millisec. Updated by anonymous on June 16 2018, 5:48:34 AM (MDT) (outdated)

# 601. Human Traffic Of Stadium

X city built a new stadium, each day many people visit it and the stats are saved as these columns: id, date, people

Please write a query to display the records which have 3 or more consecutive rows and the amount of people more than 100(inclusive).

For example, the table stadium:

| **id** | **date** | **people** |
| --- | --- | --- |
| 1 | 2017-01-01 | 10 |
| 2 | 2017-01-02 | 109 |
| 3 | 2017-01-03 | 150 |
| 4 | 2017-01-04 | 99 |
| 5 | 2017-01-05 | 145 |
| 6 | 2017-01-06 | 1455 |
| 7 | 2017-01-07 | 199 |
| 8 | 2017-01-08 | 188 |

For the sample data above, the output is:

| **id** | **date** | **people** |
| --- | --- | --- |
| 5 | 2017-01-05 | 145 |
| 6 | 2017-01-06 | 1455 |
| 7 | 2017-01-07 | 199 |
| 8 | 2017-01-08 | 188 |

Note:  
Each day only have one row record, and the dates are increasing with id increasing.

FINISHED

Took 0 millisec. Updated by anonymous on June 16 2018, 10:49:45 AM (MDT) (outdated)

# 615. Average Salary: Departments vs. Company

Given two tables as below, write a query to display the comparison result (higher/lower/same) of the average salary of employees in a department to the company’s average salary.  
Table: salary

| **id** | **employee\_id** | **amount** | **pay\_date** |
| --- | --- | --- | --- |
| 1 | 1 | 9000 | 2017-03-31 |
| 2 | 2 | 6000 | 2017-03-31 |
| 3 | 3 | 10000 | 2017-03-31 |
| 4 | 1 | 7000 | 2017-02-28 |
| 5 | 2 | 6000 | 2017-02-28 |
| 6 | 3 | 8000 | 2017-02-28 |

The employee\_id column refers to the employee\_id in the following table employee.

| **employee\_id** | **department\_id** |
| --- | --- |
| 1 | 1 |
| 2 | 2 |
| 3 | 2 |

So for the sample data above, the result is:

| **pay\_month** | **department\_id** | **comparison** |
| --- | --- | --- |
| 2017-03 | 1 | higher |
| 2017-03 | 2 | lower |
| 2017-02 | 1 | same |
| 2017-02 | 2 | same |

Explanation

In March, the company’s average salary is (9000+6000+10000)/3 = 8333.33…

The average salary for department ‘1’ is 9000, which is the salary of employee\_id ‘1’ since there is only one employee in this department. So the comparison result is ‘higher’ since 9000 > 8333.33 obviously.

The average salary of department ‘2’ is (6000 + 10000)/2 = 8000, which is the average of employee\_id ‘2’ and ‘3’. So the comparison result is ‘lower’ since 8000 < 8333.33.

With he same formula for the average salary comparison in February, the result is ‘same’ since both the department ‘1’ and ‘2’ have the same average salary with the company, which is 7000.

FINISHED

Took 0 millisec. Updated by anonymous on June 16 2018, 7:05:54 AM (MDT) (outdated)

# 618. Students Report by Geography

A U.S graduate school has students from Asia, Europe and America. The students’ location information are stored in table student as below.

| **name** | **continent** |
| --- | --- |
| Jack | America |
| Pascal | Europe |
| Xi | Asia |
| Jane | America |

Pivot the continent column in this table so that each name is sorted alphabetically and displayed underneath its corresponding continent. The output headers should be America, Asia and Europe respectively. It is guaranteed that the student number from America is no less than either Asia or Europe.

For the sample input, the output is:

| **America** | **Asia** | **Europe** |
| --- | --- | --- |
| Jack | Xi | Pascal |
| Jane |  |  |

Follow-up: If it is unknown which continent has the most students, can you write a query to generate the student report?

FINISHED

Took 0 millisec. Updated by anonymous on June 16 2018, 7:41:34 AM (MDT) (outdated)

# 569. Median Empployee Salary

The Employee table holds all employees. The employee table has three columns: Employee Id, Company Name, and Salary.

| **Id** | **Company** | **Salary** |
| --- | --- | --- |
| 1 | A | 2341 |
| 2 | A | 341 |
| 3 | A | 15 |
| 4 | A | 15314 |
| 5 | A | 451 |
| 6 | A | 513 |
| 7 | B | 15 |
| 8 | B | 13 |
| 9 | B | 1154 |
| 10 | B | 1345 |
| 11 | B | 1221 |
| 12 | B | 234 |
| 13 | C | 2345 |
| 14 | C | 2645 |
| 15 | C | 2645 |
| 16 | C | 2652 |
| 17 | C | 65 |

Write a SQL query to find the median salary of each company. Bonus points if you can solve it without using any built-in SQL functions.

| **Id** | **Company** | **Salary** |
| --- | --- | --- |
| 5 | A | 451 |
| 6 | A | 513 |
| 12 | B | 234 |
| 9 | B | 1154 |
| 14 | C | 2645 |

FINISHED

Took 0 millisec. Updated by anonymous on June 16 2018, 8:01:58 AM (MDT) (outdated)

# 571. Find Median given frequency of Numbers

The Numbers table keeps the value of number and its frequency.

| **Number** | **Frequency** |
| --- | --- |
| 0 | 7 |
| 1 | 1 |
| 2 | 3 |
| 3 | 1 |

In this table, the numbers are 0, 0, 0, 0, 0, 0, 0, 1, 2, 2, 2, 3, so the median is (0 + 0) / 2 = 0.

| **median** |
| --- |
| 0.0000 |

Write a query to find the median of all numbers and name the result as median.

FINISHED

Took 0 millisec. Updated by anonymous on June 16 2018, 8:40:23 AM (MDT) (outdated)