

```

-- create a table
CREATE TABLE poll_investments (
    user_id VARCHAR(10),
    poll_id VARCHAR(10),
    poll_option_id CHAR(1),
    amount INT,
    created_dt DATE
);

-- insert some values
INSERT INTO poll_investments (user_id, poll_id, poll_option_id, amount, created_dt)
VALUES
('id1', 'p1', 'A', 200, '2021-12-01'),
('id2', 'p1', 'C', 250, '2021-12-01'),
('id3', 'p1', 'A', 200, '2021-12-01'),
('id4', 'p1', 'B', 500, '2021-12-01'),
('id5', 'p1', 'C', 50, '2021-12-01'),
('id6', 'p1', 'D', 500, '2021-12-01'),
('id7', 'p1', 'C', 200, '2021-12-01'),
('id8', 'p1', 'A', 100, '2021-12-01');

-- query/read
SELECT * FROM poll_investments;

WITH loser AS(
    -- lost_amt=>1500 by A, B, D
    SELECT SUM(amount) AS lost_amt
    FROM poll_investments
    WHERE poll_option_id IN ('A', 'B', 'D')),

winner AS(
    -- gain who poll on C=>user_id: id2, id5, id7
    SELECT user_id, amount, amount/ -- finding out the ratios of winners
        (SELECT SUM(amount) FROM poll_investments WHERE poll_option_id='C') AS ratio
    FROM poll_investments
    WHERE poll_option_id = 'C')

SELECT user_id, ROUND((ratio*lost_amt + amount)) AS Returns
FROM winner, loser

```

<https://www.mycompiler.io/view/C34fl3zmtiI>

```

-- create a table
CREATE TABLE sales_data (
    city VARCHAR(50),
    year INT,
    month INT,
    sales INT
);

-- insert some values
INSERT INTO sales_data (city, year, month, sales) VALUES
('Delhi', 2020, 5, 4300), ('Delhi', 2020, 6, 2000), ('Delhi', 2020, 7, 2100), ('Delhi',
2020, 8, 2200),
('Delhi', 2020, 9, 1900), ('Delhi', 2020, 10, 200),
('Mumbai', 2020, 5, 4400), ('Mumbai', 2020, 6, 2800), ('Mumbai', 2020, 7,
6000), ('Mumbai', 2020, 8, 9300),
('Mumbai', 2020, 9, 4200), ('Mumbai', 2020, 10, 9700),
('Bangalore', 2020, 5, 1000), ('Bangalore', 2020, 6, 2300), ('Bangalore', 2020, 7,
6800),
('Bangalore', 2020, 8, 7000), ('Bangalore', 2020, 9, 2300), ('Bangalore', 2020, 10,
8400);

-- query/read
SELECT *
FROM sales_data;

WITH serial AS(
-- serial order of city Delhi-1, Mumbai-2, Bangalore-3
SELECT *, CASE city WHEN "Delhi" THEN 1
                WHEN "Mumbai" THEN 2
                WHEN "Bangalore" THEN 3
                END AS city_order
FROM sales_data)

-- Col: city, year, month, sales, [city_order]
SELECT city, year, month, sales,
LAG(sales) OVER(PARTITION BY city_order ORDER BY month) AS previousMonthSales,
LEAD(sales) OVER(PARTITION BY city_order ORDER BY month) AS nextMonthSales,
SUM(sales) OVER(PARTITION BY city_order ORDER BY month) AS YTDSales
FROM serial

```

<https://onecompiler.com/mysql/42mvztjcu>