

1)

You have below SQL tables: authors and books.

The authors dataset has 1M+ rows; here's the first six rows:

author_name	book_name
author_1	book_1
author_1	book_2
author_2	book_3
author_2	book_4
author_2	book_5
author_3	book_6

... ..

The books dataset also has 1M+ rows and here's the first six:

book_name	sold_copies
book_1	1000
book_2	1500
book_3	34000
book_4	29000
book_5	40000
book_6	4400

... ..

Create an SQL query that shows the TOP 3 authors who sold the most books in total!

Answer:

```
SELECT DISTINCT author_name FROM authors WHERE book_name IN (SELECT TOP 3 FROM books);
```

2)

You work for a startup that makes an online presentation software. Table has an event log that records every time a user inserted an image into a presentation. (One user can insert multiple images.) The event_log SQL table looks like this:

user_id	event_date_time
7494212	1535308430

7494212 1535308433
1475185 1535308444
6946725 1535308475
6946725 1535308476
6946725 1535308477

... ..

...and it has over one billion rows.

Note: If the event_date_time column's format doesn't look familiar, google "epoch timestamp"!

Write an SQL query to find out how many users inserted more than 1000 but less than 2000 images in their presentations!

Answer:

```
SELECT COUNT(DISTINCT user_id) FROM Table WHERE COUNT(user_id) BETWEEN 1000 and 2000;
```

3)

You have two SQL tables! The first one is called employees and it contains the employee names, the unique employee ids and the department names of a company. Sample:

department_name	employee_id	employee_name
Sales	123	John Doe
Sales	211	Jane Smith
HR	556	Billy Bob
Sales	711	Robert Hayek
Marketing	235	Edward Jorgson
Marketing	236	Christine Packard

... ..

The second one is named salaries. It holds the same employee names and the same employee ids – and the salaries for each employee. Sample:

salary	employee_id	employee_name
500	123	John Doe
600	211	Jane Smith
1000	556	Billy Bob

400 711 Robert Hayek
1200 235 Edward Jorgson
200 236 Christine Packard

.....

The company has 546 employees, so both tables have 546 rows.

Print every department where the average salary per employee is lower than \$500!

Answer:

```
SELECT DISTINCT department_name WHERE employee_name IN (SELECT employee_name  
FROM salaries GROUP BY salary HAVING AVG(salary) < 500);
```

4) results as shown in Results table below, please write SQL for that (SQL to produce Results):

Pricing

pricedt	symbol	price
1/1/2019	AAPL	100.5
1/7/2019	AAPL	120.25
1/11/2019	MSFT	200
1/20/2019	AAPL	300
1/24/2019	MSFT	306.6
1/25/2019	GOOG	999
4/1/2019	AAPL	400
4/11/2019	AAPL	700
12/25/2019	MSFT	400
12/31/2019	GOOG	1000

Ratings

effectivedt	symbol	rating
1/1/2019	AAPL	A+
1/11/2019	MSFT	B+
1/20/2019	AAPL	A-
1/25/2019	GOOG	A+
4/1/2019	AAPL	A++
12/25/2019	MSFT	B-

Results

pricedt	symbol	price	rating
1/1/2019	AAPL	100.5	A+
1/7/2019	AAPL	120.25	A+
1/11/2019	MSFT	200	B+
1/20/2019	AAPL	300	A-
1/24/2019	MSFT	306.6	B+

1/25/2019	GOOG	999	A+
4/1/2019	AAPL	400	A++
4/11/2019	AAPL	700	A++
12/25/2019	MSFT	400	B-
12/31/2019	GOOG	1000	A+

Answer:

SELECT pricedt, symbol, price, rating FROM Pricing INNER JOIN Ratings ON symbol = symbol;