# Тестовые данные

Таблица 1 Таблица 2

|  |  |  |  |
| --- | --- | --- | --- |
| **city** | **type** | **cost** | **reportdate** |
| null | Peach | 632 | 31.03.2020 |
| Paris | Peach | 268 | 01.04.2020 |
| Paris | Peach | 115 | 02.04.2020 |
| Paris | Apple | 303 | 31.03.2020 |
| Paris | Apple | 4 | 01.04.2020 |
| Paris | Apple | 734 | 02.04.2020 |
| Paris | Pinapple | null | 31.03.2020 |
| Paris | Pinapple | 600 | 02.04.2020 |
| Paris | Pinapple | 160 | 03.04.2020 |
| Lyon | Peach | 337 | 31.03.2020 |
| Lyon | Peach | 740 | null |
| Lyon | Peach | 62 | 02.04.2020 |
| Lyon | Apple | 983 | 31.03.2020 |
| Lyon | Apple | 509 | 01.04.2020 |
| Lyon | Apple | 609 | 02.04.2020 |
| Marseille | Watermelon | 869 | 01.04.2020 |
| Marseille | Watermelon | 129 | 04.04.2020 |
| Marseille | Watermelon | 761 | 05.04.2020 |

|  |  |  |  |
| --- | --- | --- | --- |
| **city** | **type** | **cost** | **reportdate** |
| Paris | Peach | 690 | 31.03.2020 |
| Paris | Peach | 95 | 01.04.2020 |
| Paris | Peach | 50 | 02.04.2020 |
| Paris | Apple | 538 | 31.03.2020 |
| Paris | Apple | 303 | 01.04.2020 |
| Paris | Apple | 836 | 02.04.2020 |
| Paris | Pinapple | 137 | 31.03.2020 |
| Paris | Pinapple | 849 | 02.04.2020 |
| Paris | Pinapple | 900 | 03.04.2020 |
| Lyon | Peach | 20 | 31.03.2020 |
| Lyon | Peach | 255 | 01.04.2020 |
| Lyon | Peach | 706 | 02.04.2020 |
| Lyon | Apple | 284 | 31.03.2020 |
| Lyon | Apple | 958 | 01.04.2020 |
| Lyon | Melon | 790 | 02.04.2020 |
| Lyon | Pinapple | 535 | 01.04.2020 |
| Lyon | Pinapple | 29 | 04.04.2020 |
| Lyon | Pinapple | 610 | 06.04.2020 |
| null | Apple | 438 | 06.04.2020 |

# Задания

Напишите SQL-запросы, которые выполнят следующее:

1. Выведите всю информацию по городу Paris из таблицы 1

SELECT \* FROM table\_1 WHERE city='Paris';

1. Выведите все уникальные даты из таблицы #city\_from\_base1, где есть информация по городу Lyon для ананасов (Pinapple)

SELECT DISTINCT \* FROM table\_1 WHERE city='Lyon' AND type='Pinapple';

1. Выведите сумму всех стоимостей в одном запросе по датам и по городам и по типам товара из таблицы 2

SELECT reportdate AS 'Group', SUM(cost)

FROM table\_2

GROUP BY reportdate

UNION ALL

SELECT city, SUM(cost)

FROM table\_2

GROUP BY city

UNION ALL

SELECT type, SUM(cost)

FROM table\_2

GROUP BY type;

1. Покажите все строчки где есть null из таблицы 2

SELECT \*

FROM table\_2

WHERE city ISNULL

OR type ISNULL

OR cost ISNULL

OR reportdate ISNULL;

1. Выведите уникальный список городов и товаров из обеих таблиц

SELECT DISTINCT city, type

FROM table\_1

UNION

SELECT DISTINCT city, type

FROM table\_2;

1. Найдите такие типы товаров для Lyon, которые отсутствуют в таблице 2, но которые есть в таблице 1

SELECT type

FROM table\_1

WHERE city='Lyon'

EXCEPT

SELECT type

FROM table\_2

WHERE city='Lyon';

1. Найдите все даты из таблицы 1, которых нет в таблице 2

SELECT reportdate

FROM table\_1

EXCEPT

SELECT reportdate

FROM table\_2;

1. Просуммируйте стоимость в таблицах для города Paris, по типам товара

SELECT type, SUM(cost)

FROM (SELECT \* FROM table\_1 UNION ALL SELECT \* FROM table\_2)

WHERE city='Paris'

GROUP BY type;

1. Объедините данные из двух таблиц, найдите максимальную и минимальную стоимость по каждому типу товаров и в каждом городе на каждую дату. Выведите в виде одной таблицы формата – город, дата, тип товара, макс. стоимость, мин. стоимость

SELECT type AS 'GROUP', MAX(cost), MIN(cost)

FROM (SELECT \* FROM table\_1 UNION ALL SELECT \* FROM table\_2)

GROUP BY type

UNION ALL

SELECT city, MAX(cost), MIN(cost)

FROM (SELECT \* FROM table\_1 UNION ALL SELECT \* FROM table\_2)

GROUP BY city

UNION ALL

SELECT reportdate, MAX(cost), MIN(cost)

FROM (SELECT \* FROM table\_1 UNION ALL SELECT \* FROM table\_2)

GROUP BY reportdate;

SELECT city, reportdate, type, MAX(cost), MIN(cost)

FROM (SELECT \* FROM table\_1 UNION ALL SELECT \* FROM table\_2)

GROUP BY city, reportdate, type

1. На основе предыдущего задания, добавьте к каждой строке ближайшую предыдущую дату, (если имеется) и добавьте в вывод предыдущую дату, макс.стоимость и мин.стоимость на эту дату. (можно использовать cte, временные таблицы и любые альтернативные подходы)

Если у вас есть несколько способов решения, можете присылать все возможные варианты.

Сайт где можно выполнить запросы - <https://sqliteonline.com/>

INSERT INTO table\_1 (city, type, cost, reportdate)

VALUES ('Paris', 'Peach', 690, '31.03.2020'),

('Paris', 'Peach', 95, '01.04.2020'),

('Paris', 'Peach', 50, '02.04.2020'),

('Paris', 'Apple', 538, '31.03.2020'),

('Paris', 'Apple', 303, '01.04.2020'),

('Paris', 'Apple', 836, '02.04.2020'),

('Paris', 'Pinapple', 137, '31.03.2020'),

('Paris', 'Pinapple', 849, '02.04.2020'),

('Paris', 'Pinapple', 900, '03.04.2020'),

('Lyon', 'Peach', 20, '31.03.2020'),

('Lyon', 'Peach', 255, '01.04.2020'),

('Lyon', 'Peach', 706, '02.04.2020'),

('Lyon', 'Apple', 284, '31.03.2020'),

('Lyon', 'Apple', 958, '01.04.2020'),

('Lyon', 'Melon', 790, '02.04.2020'),

('Lyon', 'Pinapple', 535, '01.04.2020'),

('Lyon', 'Pinapple', 29, '04.04.2020'),

('Lyon', 'Pinapple', 610, '06.04.2020'),

('null', 'Apple', 438, '06.04.2020')

;

INSERT INTO table\_2 (city, type, cost, reportdate)

VALUES (null, 'Peach', 632, '31.03.2020'),

('Paris', 'Peach', 115, '02.04.2020'),

('Paris', 'Apple', 303, '31.03.2020'),

('Paris', 'Apple', 4, '01.04.2020'),

('Paris', 'Apple', 734, '02.04.2020'),

('Paris', 'Pinapple', null, '31.03.2020'),

('Paris', 'Pinapple', 600, '02.04.2020'),

('Paris', 'Pinapple', 160, '03.04.2020'),

('Lyon', 'Peach', 337, '31.03.2020'),

('Lyon', 'Peach', 740, null),

('Lyon', 'Peach', 62, '02.04.2020'),

('Lyon', 'Apple', 983, '31.03.2020'),

('Lyon', 'Apple', 509, '01.04.2020'),

('Lyon', 'Apple', 609, '02.04.2020'),

('Marseille', 'Watermelon', 869, '01.04.2020'),

('Marseille', 'Watermelon', 129, '04.04.2020'),

('Marseille', 'Watermelon', 761, '05.04.2020');