**Финальное задание**

Даны две исходные таблицы:

* *FLIGHTS*, содержащая в себе информацию о рейсах;
* *FLIGHT\_TIKETS*, содержащая в себе информацию о билетах на различные рейсы.

Исходные поля таблицы *FLIGHTS*:

flight\_id: int (id рейса)

town\_from: varchar (название города вылета)

town\_to: varchar (название города прилёта)

date: datetime (дата вылета)

Исходные поля таблицы *FLIGHT\_TIKETS*:

id: int (id билета)

flight\_id: int (id рейса)

passenger: varchar (ФИО пассажира)

amount: int (стоимость билета)

Исходная таблица *FLIGHTS*:

|  |  |  |  |
| --- | --- | --- | --- |
| **flight\_id** | **user\_id\_sender** | **user\_id\_receiver** | **date** |
| 1 | Москва | Челябинск | 2023-01-01 12:35:00 |
| 2 | Париж | Берлин | 2023-01-03 14:42:00 |
| 3 | Милан | Москва | 2023-01-07 22:50:00 |
| 4 | Абакан | Краснодар | 2023-01-08 23:30:00 |
| 5 | Москва | Челябинск | 2023-01-05 04:00:00 |
| 6 | Барнаул | Владивосток | 2023-01-25 01:00:00 |
| 7 | Сочи | Ростов | 2023-01-01 12:00:00 |
| 8 | Париж | Берлин | 2023-01-16 03:00:00 |
| 9 | Лондон | Сингапур | 2023-01-10 14:00:00 |
| 10 | Москва | Челябинск | 2023-01-28 09:35:00 |
| 11 | Лондон | Сингапур | 2023-01-23 08:12:00 |
| 12 | Париж | Лондон | 2023-01-05 16:55:00 |

Исходная таблица *FLIGHT\_TIKETS*:

|  |  |  |  |
| --- | --- | --- | --- |
| **id** | **flight\_id** | **user\_id\_receiver** | **amount** |
| 1 | 2 | Калинина Т. Д. | 1000 |
| 2 | 5 | Васильева В. М. | 750 |
| 3 | 10 | Егоров Е. А. | 970 |
| 4 | 3 | Лапин А. М. | 1012 |
| 5 | 2 | Смирнова Е. С. | 841 |
| 6 | 1 | Софронов И. А. | 1400 |
| 7 | 4 | Степанов М. П. | 925 |
| 8 | 5 | Крючков М. М. | 760 |
| 9 | 11 | Иванов А. Н. | 1352 |
| 10 | 12 | Михайлов И. Ф. | 638 |
| 11 | 9 | Островская А. М. | 1107 |
| 12 | 8 | Егоров Р. М. | 900 |
| 13 | 12 | Корчагин Ф. М. | 765 |
| 14 | 6 | Никифоров М. А. | 1700 |
| 15 | 7 | Фролова М. А. | 692 |

***Задача***: Определить минимальные и максимальные цены за билеты для всех рейсов, дата вылета которых попадает в интервал 10 дней, начиная с текущего момента (сделать так, чтобы всегда бралась дата выполнения запроса). В результирующую выборку должны попасть только строки, в которых разница между максимальной и минимальной ценой за билет не больше 333. Оператор SELECT должен возвращать ROUTE (соединённые через тире town\_from и town\_to), min\_amount, max\_amount. Результат отсортировать по первому столбцу.

Пример выходных данных на момент 2023-01-01:

|  |  |  |
| --- | --- | --- |
| **route** | **min\_amount** | **max\_amount** |
| Абакан-Краснодар | 925 | 925 |
| Лондон-Сингапур | 1107 | 1107 |
| Милан-Москва | 1012 | 1012 |
| Париж-Берлин | 841 | 1000 |
| Париж-Лондон | 638 | 765 |
| Сочи-Ростов | 692 | 692 |

**Примечание:** В качестве ответа на задание необходимо предоставить SQL запрос, удовлетворяющий всем поставленным условиям. Пример выходных данных предоставлять не нужно!!!

***Пример решения:***

select CONCAT(f.town\_from,'-', f.town\_to) as route, min(t.amount) as min\_amount, max(t.amount) as max\_amount

from FLIGHTS f

left join FLIGHT\_TIKETS t on f.flight\_id=t.flight\_id

where date between NOW() and ADDDATE(NOW(), INTERVAL 10 DAY)

group by f.town\_from,f.town\_to

having max(t.amount) - min(t.amount) <=333

order by 1

**Задание на оконные функции**

Дана исходная таблица *SLARIES*, о зарплатах сотрудников в отделах.

Исходные поля:

department: varchar (департамент сотрудника)

employee: varchar (ФИО сотрудника)

salary: varchar (зарплата сотрудника)

Исходная таблица *SALARIES*:

|  |  |  |
| --- | --- | --- |
| **department** | **employee** | **salary** |
| develop | Joann Brown | 1200 |
| develop | Angela Johnson | 1500 |
| develop | Mark Brady | 1350 |
| sales | Rene Johnson | 1600 |
| sales | Steven Davis | 1100 |
| sales | Jane Peters | 2050 |
| sales | James Silva | 1740 |
| marketing | Rebecca Berry | 1480 |
| marketing | Jesus Gray | 1670 |
| marketing | Maria Edwards | 1670 |

***Задача***: С использованием оконной функции определить сотрудников, которые имеют максимальную зарплату в своём отделе (учесть ситуацию, когда несколько сотрудников могут иметь одинаковую зарплату).

Выходные данные должны выглядеть следующим образом:

|  |  |  |
| --- | --- | --- |
| **department** | **employee** | **salary** |
| develop | Angela Johnson | 1500 |
| sales | Jane Peters | 2050 |
| marketing | Jesus Gray | 1670 |
| marketing | Maria Edwards | 1670 |

***Пример решения:***

WITH base AS (SELECT department, employee, salary,

RANK() OVER(partition by department ORDER BY salary DESC) rnk

FROM SALARIES)

SELECT department, employee, salary

FROM base

WHERE rnk = 1

**Финальное задание (интерпретация)**

Даны две исходные таблицы: *EVENTS* (содержит в себе информацию о действиях пользователей на сайте и том, с какого устройства было выполнено действие) и *USERS* (содержит в себе информацию о пользователях).

Исходные поля *EVENTS*:

user\_id: int (id пользователя, совершившего действие)

time: datetime (дата действия)

event\_type: varchar (тип действия)

device: varchar (название устройства)

Исходные поля *USERS*:

user\_id: int (id пользователя, совершившего действие)

created\_at: datetime (дата создания пользователя)

language: varchar (язык пользователя)

city: varchar (город пользователя)

Фрагмент исходной таблицы *EVENTS*:

|  |  |  |  |
| --- | --- | --- | --- |
| **user\_id** | **time** | **event\_type** | **device** |
| 6 | 2023-06-09 18:26:54 | login | hp |
| 2 | 2023-08-29 13:18:38 | like\_message | macbook |
| 5 | 2023-07-01 12:47:56 | search | nexus |
| 1 | 2023-05-23 10:44:16 | send\_message | iphone |
| 4 | 2023-07-31 17:15:14 | like\_message | ipad |
| 1 | 2023-07-24 18:43:19 | enter\_email | samsung |
| 3 | 2023-08-27 05:54:28 | login | ipad |
| 2 | 2023-05-10 10:03:34 | search | macbook |
| 4 | 2023-05-26 10:42:12 | search | lenovo |
| 1 | 2023-07-30 19:39:13 | enter\_email | dell |

Фрагмент исходной таблицы *USERS*:

|  |  |  |  |
| --- | --- | --- | --- |
| **user\_id** | **created\_at** | **language** | **city** |
| 1 | 2013-01-01 4:41:13 | russian | 1 |
| 2 | 2013-01-05 15:30:45 | italian | 1 |
| 3 | 2013-01-10 11:04:58 | russian | 1 |
| 4 | 2013-01-16 20:40:24 | english | 2 |
| 5 | 2013-01-16 11:22:22 | italian | 2 |
| 6 | 2013-01-24 13:41:38 | italian |  |

***Исходный запрос:***

SELECT

language,

COUNT(DISTINCT

CASE

WHEN e.device IN ('macbook','iphone','ipad') THEN e.user\_id ELSE NULL

END) AS n\_apple\_user,

COUNT(DISTINCT e.user\_id) AS n\_total\_users

FROM *EVENTS* e

LEFT JOIN *USERS* u on e.user\_id=u.user\_id

GROUP BY u.language

ORDER BY n\_total\_users DESC

***Задание:***

Сформулируйте содержательную интерпретацию представленного SQL-запроса. Постарайтесь максимально точно описать, что делает данный запрос. Необходимо также привести таблицу, содержащую результат запроса применительно к представленным выше данным.

***Примерный вариант ответа:***

Запрос находит количество пользователей продуктов Apple и общее количество пользователей различных устройств, группируя их по языковому признаку. Результат упорядочен по убыванию общего числа пользователей.

|  |  |  |
| --- | --- | --- |
| **nationality** | **n\_apple\_user** | **n\_total\_users** |
| Italian | 1 | 3 |
| Russian | 2 | 2 |
| English | 1 | 1 |