

Лабораторная работа 3

Выполнил: Немков Даниил, КМБО-05-23

Размер БД: Большой

```
-- 1. Вычислить, как реальное время полета отличается от
запланированного
-- в среднем (в секундах), максимально (в минутах)
SELECT AVG(ABS(EXTRACT(MINUTE FROM(actulal_diff - scheduled_diff)) *
60)) as average_seconds,
        MAX(ABS(EXTRACT(MINUTE FROM(actulal_diff -
scheduled_diff)))) as max_minutes
FROM(
    SELECT    scheduled_departure,
              scheduled_arrival,
              (scheduled_arrival- scheduled_departure) AS
scheduled_diff,
              actual_departure,
              actual_arrival,
              (actual_arrival - actual_departure) AS actulal_diff
    FROM flights
    WHERE scheduled_arrival- scheduled_departure IS NOT NULL --
Можно не ставить фильтр на ненулевые значения
        and actual_arrival - actual_departure IS NOT NULL --
Ответ не меняется, но пусть будет на всякий
)
```

	average_seconds numeric	max_minutes numeric
1	58.7139034873350432	22

```
-- 2. Вычислить, сколько человек входит в одно бронирование
-- в среднем (округлить до целых), минимальное значение,
максимальное значение
SELECT ROUND(AVG(pas_count)) as avg_pas_count,
```

```

        MIN(pas_count) as min_pas_count,
        MAX(pas_count) as max_pas_count
FROM (
    SELECT book_ref, COUNT(passenger_id) as pas_count
    FROM tickets
    GROUP BY book_ref
    ORDER BY pas_count DESC
)

```

	avg_pas_count numeric	min_pas_count bigint	max_pas_count bigint
1	1	1	5

-- 3. Всегда ли билеты на места одного класса на одном рейсе стоят одинаково? Вывести TRUE или FALSE

```


-- Пояснение к запросу:
-- если максимальное количество разных цен отлично
от единицы, то
-- существует рейс, в котором на билеты какого-то
класса обслуживания
-- есть более одной цены

```

```


SELECT CASE
    WHEN MAX(price_count) > 1 THEN FALSE
    ELSE TRUE
END
FROM(
    -- тут мы нашли количество РАЗНЫХ цен на каждый класс
каждого рейса
    SELECT flight_id, fare_conditions, COUNT(DISTINCT amount) as
price_count
    FROM ticket_flights
    GROUP BY flight_id, fare_conditions
)

```

	case boolean 
1	false

-- 4.Вычислить, сколько полетов запланировано в аэропорт SVO или из аэропорта SVO

```
SELECT COUNT(DISTINCT flight_id)
FROM flights
WHERE status = 'Scheduled'
      AND (departure_airport = 'SVO'
          OR arrival_airport = 'SVO')
```

	count bigint 
1	2800

-- 5.Вычислить, сколько полетов совершено в аэропорт SVO или из него



```
SELECT COUNT(DISTINCT flight_id)
FROM flights
WHERE status = 'Arrived'
      AND (departure_airport = 'SVO' OR arrival_airport = 'SVO')
```

	count bigint 
1	35729

-- 6.Вычислить количество перелетов для одного билета
-- максимальное
-- среднее

```
SELECT MAX(flight_count) as max_flight_count,
       AVG(flight_count) as avg_flight_count
FROM(
    SELECT DISTINCT ticket_no, COUNT(flight_id) as flight_count
    FROM ticket_flights
```

```
GROUP BY ticket_no
)
```

	max_flight_count bigint 	avg_flight_count numeric 
1	6	2.8448334953185866





```
-- 7.Вывести имена пассажиров, которые хотя бы раз летали классом  
"Бизнес"
```

```
SELECT DISTINCT passenger_name  
FROM tickets  
WHERE EXISTS  
(  
    SELECT passenger_id  
    FROM ticket_flights  
    WHERE tickets.ticket_no = ticket_flights.ticket_no  
        AND fare_conditions = 'Business'  
)
```



	passenger_name text 
1	ADELINA ABRAMOVA
2	ADELINA AFANASEVA
3	ADELINA ALEKSEEVA
4	ADELINA ANDREEVA
5	ADELINA ANISIMOVA
6	ADELINA ARKHIPOVA
7	ADELINA BARANOVA
8	ADELINA BELOVA
9	ADELINA BELYAEVA
10	ADELINA BOGDANOVA
11	ADELINA DANILOVA
12	ADELINA EFIMOVA
13	ADELINA EGOROVA
14	ADELINA ERMAKOVA
15	ADELINA FEDOROVA
16	ADELINA FILATOVA
17	ADELINA FILIPPOVA
18	ADELINA FOMINA
19	ADELINA FROLOVA
Total rows: 1000 of 26982 Query	

```
-- 8.Вывести количество бронирований, в которое входит пассажир
"ILYA PAVLOV",
```

```
-- а также среднюю стоимость таких бронирований, минимальную
-- стоимость таких бронирований,
-- максимальную стоимость таких бронирований (одной строкой)
SELECT COUNT(DISTINCT book_ref) as books_count,
        AVG(total_amount) as avg_amount,
        MIN(total_amount) as min_amount,
        MAX(total_amount) as max_amount
FROM bookings
WHERE EXISTS
    (
    SELECT *
    FROM tickets
    WHERE tickets.book_ref = bookings.book_ref
        AND passenger_name = 'ILYA PAVLOV'
    )
```

	books_count bigint 	avg_amount numeric 	min_amount numeric 	max_amount numeric 
1	281	91634.163701067616	6000.00	666000.00



```
-- 9.Выяснить, сколько всего мест во всех самолетах соответствует
-- каждому классу обслуживания
SELECT COUNT(seat_no) as seat_count, fare_conditions
FROM seats
GROUP BY fare_conditions
```

	seat_count bigint 	fare_conditions character varying (10) 
1	152	Business
2	48	Comfort
3	1139	Economy

```
-- 10.Вывести все даты, в которые в аэропорт DME прилетали самолеты
-- или
-- из аэропорта DME самолеты улетали. Формат вывода:
-- столбец с датами,
-- столбец с типом полета (прилет/вылет). Отсортировать
```

по дате (сначала свежие)

```
SELECT DISTINCT date, type
FROM
    (SELECT actual_departure::date as date, 'departure' as type
    FROM flights
    WHERE departure_airport = 'DME'
        AND actual_departure IS NOT NULL
        AND status = 'Arrived')
UNION
    (SELECT actual_arrival::date as date, 'arrival' as type
    FROM flights
    WHERE arrival_airport = 'DME'
        AND actual_arrival IS NOT NULL
        AND status = 'Departed')
ORDER BY DATE DESC
```

	date date		type text	
1	2017-08-15		departure	
2	2017-08-14		departure	
3	2017-08-13		departure	
4	2017-08-12		departure	
5	2017-08-11		departure	
6	2017-08-10		departure	
7	2017-08-09		departure	
8	2017-08-08		departure	
9	2017-08-07		departure	
10	2017-08-06		departure	
11	2017-08-05		departure	
12	2017-08-04		departure	
13	2017-08-03		departure	
14	2017-08-02		departure	
15	2017-08-01		departure	
Total rows: 366 of 366 Query				