|  |
| --- |
|  |
| МИНОБРНАУКИ РОССИИ  Федеральное государственное бюджетное образовательное учреждение  высшего образования  **«МИРЭА – Российский технологический университет»**  **РТУ МИРЭА** |

Институт комплексной безопасности и специального приборостроения

Кафедра КБ-14 «Интеллектуальные системы информационной безопасности»

### Политики безопасности баз данных

### Практическая работа № 1

### Работа со сторонними базами данными. Построение и оптимизация

# ОТЧЁТ

# Выполнил студент группы БСБО-07-20 Любовский С.В.

# Задание 1.

Скачать и запустить демонстрационную базу данных с сайта PostgresPRO. Выполните следующие запросы:

1. Вывести какие модели самолётов летают в Уфу
2. Среднее количество людей на рейсах из Санкт-Петербурга в Москву
3. Найти модель самолета с максимальным количеством сидений (учитывается, что таких моделей может быть несколько)
4. Вывести рейсы, число мест в которых больше, чем проданных на них билетов
5. Вывести общую сумму потраченные на билеты каждым пассажиром
6. На каких местах сидел пассажир, летающий чаще всего?
7. Выведите таблицу самолетов, отсортированных по убыванию количества мест с дополнительным атрибутом, в котором самолёты пронумерованы по частоте полётов.

# Выполнение задания 1.

1.

SELECT

    distinct(a.model)

FROM

    aircrafts a

    LEFT OUTER JOIN flights f ON a.aircraft\_code = f.aircraft\_code

    AND f.arrival\_airport = 'UFA';

2.

SELECT

    AVG(a.c)

FROM

    (

        SELECT

            COUNT(tf.ticket\_no) c

        FROM

            flights f

            LEFT OUTER JOIN ticket\_flights tf ON f.flight\_id = tf.flight\_id

        WHERE

            f.departure\_airport IN (

                SELECT

                    a.airport\_code

                FROM

                    airports a

                WHERE

                    a.city = 'Санкт-Петербург'

            )

            AND f.arrival\_airport IN (

                SELECT

                    a.airport\_code

                FROM

                    airports a

                WHERE

                    a.city = 'Москва'

            )

        GROUP BY

            f.flight\_id

    ) a;

3.

WITH model\_seatc AS (

    SELECT

        COUNT(\*) c,

        a.model m

    FROM

        aircrafts a

        LEFT OUTER JOIN seats s ON a.aircraft\_code = s.aircraft\_code

    GROUP BY

        a.model

)

SELECT

    model\_seatc.c "Количество мест",

    model\_seatc.m "Модель самолета"

FROM

    model\_seatc

WHERE

    model\_seatc.c = (

        SELECT

            MAX(model\_seatc.c)

        FROM

            model\_seatc

    );

4.

WITH cs AS (

    SELECT

        a.aircraft\_code ac,

        COUNT(\*) sc

    FROM

        aircrafts a

        LEFT OUTER JOIN seats s ON a.aircraft\_code = s.aircraft\_code

    GROUP BY

        a.aircraft\_code

),

fta AS (

    SELECT

        f.flight\_no fn,

        COUNT(\*) tc,

        f.aircraft\_code ac

    FROM

        flights f

        LEFT OUTER JOIN ticket\_flights tf ON f.flight\_id = tf.flight\_id

    GROUP BY

        f.flight\_no,

        f.aircraft\_code

)

SELECT

    fta.fn flight\_no,

    fta.tc tickets\_count,

    cs.sc seats\_count

FROM

    fta

    LEFT OUTER JOIN cs ON fta.ac = cs.ac

WHERE

    fta.tc < cs.sc;

5.

SELECT

    t.passenger\_name,

    SUM(b.total\_amount) AS total\_spent

FROM

    tickets t

    LEFT OUTER JOIN bookings b ON t.book\_ref = b.book\_ref

GROUP BY

    t.passenger\_name;

6.

SELECT

    t.passenger\_name,

    bp.seat\_no

FROM

    (

        SELECT

            t.passenger\_id pasid,

            COUNT(tf.flight\_id) fc

        FROM

            tickets t

            LEFT OUTER JOIN ticket\_flights tf ON t.ticket\_no = tf.ticket\_no

        GROUP BY

            t.passenger\_id

        ORDER BY

            fc DESC

        LIMIT

            1

    ) AS pasid\_fc

    LEFT OUTER JOIN tickets t ON pasid\_fc.pasid = t.passenger\_id

    LEFT OUTER JOIN boarding\_passes bp ON t.ticket\_no = bp.ticket\_no;

7.

SELECT

    a.\*,

    pf.fc flight\_count,

    ps.sc seats\_count

FROM

    aircrafts a

    LEFT OUTER JOIN (

        SELECT

            a1.aircraft\_code acode,

            count(f.flight\_id) fc

        FROM

            aircrafts a1

            LEFT OUTER JOIN flights f ON a1.aircraft\_code = f.aircraft\_code

        GROUP BY

            a1.aircraft\_code

    ) pf ON pf.acode = a.aircraft\_code

    LEFT OUTER JOIN (

        SELECT

            s.aircraft\_code acode,

            count(s.seat\_no) sc

        FROM

            seats s

        GROUP BY

            s.aircraft\_code

    ) ps ON ps.acode = a.aircraft\_code

ORDER BY

    ps.sc DESC;

# Задание 2.

1) Table1

|  |  |  |
| --- | --- | --- |
| ID | NAME | CLASS\_ITEM |
| 1 | Арфа | A |
| 2 | аккордеон | NULL |
| 3 | Барабан | B |
| 4 | рояль | NULL |
| 5 | труба | A |
| 6 | Пианино | C |

Что выведет следующий запрос и почему:

SELECT \* FROM Table1 WHERE CLASS\_ITEM <> ‘A’;

Ответ:

|  |  |  |
| --- | --- | --- |
| ID | NAME | CLASS\_ITEM |
| 3 | Барабан | B |
| 6 | Пианино | C |

В связи с тем, что любое\_значение <> null == null, а не True, то строки с id 2, 4 не будут в результирующей выборке, как и строки с id 1,5 т.к. в них значение class\_item = ‘A’

2) Какие значения выведут следующие операции и объясните почему:

1. NULL <> 1
   1. Ответ: null
2. NULL <> NULL
   1. Ответ: null
3. NULL = NULL
   1. Ответ: null
4. NULL IS NOT NULL
   1. Ответ:false
5. NULL IS NULL
   1. Ответ:true

3) Дана таблица:

Table1

|  |  |  |
| --- | --- | --- |
| ID | NAME | CLASS\_ITEM |
| 1 | Арфа | A |
| 2 | аккордеон | NULL |
| 3 | Барабан | B |
| 4 | рояль | NULL |
| 5 | труба | A |
| 6 | Пианино | C |

Какие строчки вернет следующий запрос:

SELECT \* FROM Table1 WHERE NAME LIKE ‘A%’;

Ответ: строка с id = 1

4) Дана таблица:

Table1

|  |  |  |
| --- | --- | --- |
| ID | NAME | CLASS\_ITEM |
| 1 | Арфа | A |
| 2 | аккордеон | NULL |
| 3 | Барабан | B |
| 4 | рояль | NULL |
| 5 | труба | A |
| 6 | Пианино | C |

Какие строчки вернет следующий запрос:

SELECT \* FROM Table1 WHERE UPPER(NAME) LIKE ‘A%’;

Ответ: будет являться таблица из 2 строк c id = 1 и 2

5) Имеется две таблицы Т1 и Т2. Известно, что в таблице Т1 - 3 строчки, а в таблице Т2 - 4 строчки. Не зная какие цифры лежат в графах NOM обеих таблиц, необходимо предположить какое минимальное и какое максимальное количество строчек может вернуть запрос ниже:

SELECT \* FROM T1 LEFT JOIN T2 ON T1.NOM = T2.NOM;

Ответ:

Минимальное количество строчек будет тогда, когда в кортежах первой таблицы не будет ни одного совпадающего значения столбца NOM, то есть 3 кортежа в итоге.

Максимальное количество строчек будет тогда, когда все значения набора кортежей 1ой таблицы столбца NOM будут идентичны любым кортежам столбца NOM 2ой таблицы, то есть 4 (Тут важно уточнить, что это возможно только при условии, что столбец NOM не уникален и может содержать одинаковые значения, в противном случае ответ 3).

6)Найдите ошибку в SQL запросе:

SELECT ID\_ITEM, NAME\_ITEM, EXTRACT(YEAR FROM DATE\_IMPORT) AS YEAR\_IMPORTANCE FROM Table1 WHERE YEAR\_IMPORTANCE > 2010;

Ответ: Alias YEAR\_IMPORTANCE нельзя использовать в WHERE, требуется JOIN, а также желательно уточнить, что внутри DATE\_IMPORT будет тип timestamp.