|  |
| --- |
| https://lh6.googleusercontent.com/QcftzNtI05T0Y6fjdSh1Rr2rt8oqZ1IvnLvbn1jLJ7CCyteVir3k-xBLv4SL1wAgWJsRhmmJSR0UW-RP63_GQenE4vVWv05BRoZTsmIcBccVTnfxwmsnNMvjg599x9SqZd8E3dkd |
| МИНОБРНАУКИ РОССИИ |
| Федеральное государственное бюджетное образовательное учреждение высшего образования  **«МИРЭА - Российский технологический университет»**  **РТУ МИРЭА** |

Институт перспективных технологий и индустриального программирования (ИПТИП)

|  |  |
| --- | --- |
| **ОТЧЕТ ПО ЛАБОРАТОРНЫМ РАБОТАМ** | |
| **по дисциплине** | |
| «Базы данных и анализ промышленных данных» | |
| **Лабораторная работа №2** | |
| Выполнил студент группы ЭФМО-02-23 | Мурадов Н.Н. |

Москва 2023

**СОДЕРЖАНИЕ**

[Задача 3](#_Toc146411761)

[Решение 5](#_Toc146411762)

[Результаты 6](#_Toc146411763)

Задача

**Создайте следующие запросы:**

1. Выбрать рейсы из города А в город Б

|  |  |
| --- | --- |
| **Вх. данные** | **Вых. данные** |
| Город А, город Б *Пример: Город=Москва*  *Город = Санкт-Петербург* | Дата и время отправления и прибытия |

1. Выбрать промежуточные станции для определенного рейса

|  |  |
| --- | --- |
| **Вх. данные** | **Вых. данные** |
| Номер поезда, дата и время отправления  *Пример:*  *Номер поезда=05*  *Дата и время отправления 03.09.2023 17:20* | Список промежуточных станций, отсортированных по пути следования |

1. Выбрать номера поездов, которые останавливаются на указанной промежуточной станции

|  |  |
| --- | --- |
| **Вх. данные** | **Вых. данные** |
| Название промежуточной станции, дата и время отправления | Список номеров поездов, пункт отправления |

1. Выбрать типы вагонов, которые есть на определенном рейсе

|  |  |
| --- | --- |
| **Вх. данные** | **Вых. данные** |
| Номер поезда, дата и время отправления | Список типов вагонов отсортированных в алфавитном порядке названия пункта  отправления |

Иногда возникает необходимость рассчитать количество строк результата, определить среднее, максимально или минимальное значение для поля. Для этих целей используются агрегатные функции (см. таблица 1).

Таблица 1 – Агрегатные функции SQL

|  |  |
| --- | --- |
| **Агрегатная функция** | **Назначение** |
| СOUNT(\*) | количество строк, возвращаемых запросом |
| MAX(имя\_поля) | максимальное значение для поля |
| MIN(имя\_поля) | минимальное значение для поля |
| AVG (имя\_поля) | среднее значение для поля |
| SUM (имя\_поля) | сумма значений всех строк поля |

1. Вывести количество свободных мест на определенный рейс

|  |  |
| --- | --- |
| **Вх. данные** | **Вых. данные** |
| Номер поезда, дата и время  отправления | Количество свободных мест |

1. Рассчитать среднюю стоимость места в вагоне-купе для поездов из города А в город Б

|  |  |
| --- | --- |
| **Вх. данные** | **Вых. данные** |
| Город А, город Б, тип вагона | Средняя стоимость |

1. Вывести стоимость самого дорогого билета для поезда отправляющегося в указанную дату

|  |  |
| --- | --- |
| **Вх. данные** | **Вых. данные** |
| Дата отправления | Стоимость |

1. Вывести количество свободных нижних мест в плацкарте на указанный рейс *(нижние места имеют нечетный номер)*

|  |  |
| --- | --- |
| **Вх. данные** | **Вых. данные** |
| Номер поезда, дата отправления,  тип вагона | Количество свободных нижних  мест |

1. Вывести минимальную стоимость билета в купе поезда, проезжающего указанную промежуточную станцию в определенную дату

|  |  |
| --- | --- |
| **Вх. данные** | **Вых. данные** |
| Тип вагона, название промежуточной станции, дата  отправления | Минимальная стоимость билета, номер поезда, пункт отправления,  пункт прибытия |

1. Вывести номера поездов, которые останавливаются на указанной промежуточной станции, в течение последующих 90 суток

|  |  |
| --- | --- |
| **Вх. данные** | **Вых. данные** |
| Название промежуточной станции | Номер поезда, дата отправления,  отсортированная в порядке возрастания |

Решение

Листинг кода:

SELECT JOURNEY.TIME\_OF\_DEPARTURE, JOURNEY.TIME\_OF\_ARRIVAL FROM JOURNEY JOIN ROUTE ON JOURNEY.ROUTE\_ID = ROUTE.ID WHERE (ROUTE.DEPARTURE\_POINT = 2 AND ROUTE.ARRIVAL\_POINT = 7);

SELECT w.\* FROM JOURNEY j INNER JOIN ROUTE r ON j.ROUTE\_ID = r.ID INNER JOIN WAY\_STATION w ON w.JOURNEY\_ID = j.ID WHERE (r.TRAIN\_NUMBER = '#5341' AND j.TIME\_OF\_DEPARTURE = '2016-07-22 19:10:25') ORDER BY w.CITY\_ID;

SELECT r.TRAIN\_NUMBER, r.ARRIVAL\_POINT FROM JOURNEY j INNER JOIN ROUTE r ON j.ROUTE\_ID = r.ID INNER JOIN WAY\_STATION w ON w.JOURNEY\_ID = j.ID WHERE (w.ID = 44 AND w.TIME\_OF\_DEPARTURE = '2016-07-24 20:10:25');

SELECT t.TYPE\_WAGON, c.NAME FROM JOURNEY j INNER JOIN ROUTE r ON j.ROUTE\_ID = r.ID INNER JOIN TICKET t ON t.JOURNEY\_ID = j.ID INNER JOIN CITY c ON r.DEPARTURE\_POINT = c.ID WHERE (r.TRAIN\_NUMBER = '#5343' AND j.TIME\_OF\_DEPARTURE = '2016-07-24 19:10:25') ORDER BY c.NAME;

SELECT COUNT(t.\*) FROM JOURNEY j INNER JOIN ROUTE r ON j.ROUTE\_ID = r.ID INNER JOIN TICKET t ON t.JOURNEY\_ID = j.ID WHERE (r.TRAIN\_NUMBER = '#5343' AND j.TIME\_OF\_DEPARTURE = '2016-07-24 19:10:25' AND t.STATE = 2);

SELECT AVG(t.PRICE) FROM JOURNEY j INNER JOIN ROUTE r ON j.ROUTE\_ID = r.ID INNER JOIN TICKET t ON t.JOURNEY\_ID = j.ID WHERE (r.DEPARTURE\_POINT = 2 AND r.ARRIVAL\_POINT = 3 AND t.TYPE\_WAGON = 2);

SELECT MAX(t.PRICE) FROM JOURNEY j JOIN TICKET t ON t.JOURNEY\_ID = j.ID WHERE (j.TIME\_OF\_DEPARTURE = '2016-07-22 19:10:25');

SELECT COUNT(t.\*) FROM JOURNEY j INNER JOIN ROUTE r ON j.ROUTE\_ID = r.ID INNER JOIN TICKET t ON t.JOURNEY\_ID = j.ID WHERE (( t.PLACE % 2 ) != 0 AND r.TRAIN\_NUMBER = '#5341' AND j.TIME\_OF\_DEPARTURE = '2016-07-22 19:10:25' AND t.TYPE\_WAGON = 1);

SELECT r.TRAIN\_NUMBER, r.DEPARTURE\_POINT, r.ARRIVAL\_POINT, MIN(t.PRICE) as MIN\_PRICE FROM JOURNEY j INNER JOIN ROUTE r ON j.ROUTE\_ID = r.ID INNER JOIN TICKET t ON t.JOURNEY\_ID = j.ID INNER JOIN WAY\_STATION w ON w.JOURNEY\_ID = j.ID WHERE (w.TIME\_OF\_DEPARTURE = '2016-07-24 20:10:25' AND t.TYPE\_WAGON = 2 AND w.CITY\_ID = 5) GROUP BY r.TRAIN\_NUMBER, r.DEPARTURE\_POINT, r.ARRIVAL\_POINT;

SELECT r.TRAIN\_NUMBER, w.TIME\_OF\_DEPARTURE FROM JOURNEY j INNER JOIN ROUTE r ON j.ROUTE\_ID = r.ID INNER JOIN TICKET t ON t.JOURNEY\_ID = j.ID INNER JOIN WAY\_STATION w ON w.JOURNEY\_ID = j.ID WHERE (w.TIME\_OF\_DEPARTURE > '2016-07-22 19:10:25' AND w.TIME\_OF\_DEPARTURE <= '2016-07-22 19:10:25'::timestamp + '90 day'::interval AND w.JOURNEY\_ID = 22) ORDER BY w.TIME\_OF\_DEPARTURE;

Результаты

