**Липецкий государственный технический университет**

Факультет автоматизации и информатики Кафедра автоматизированных систем управления

Задание №3

по дисциплине «Расширенный курс по базам данных»

Написание сложных запросов. Конструкция WITH. Агрегатные функции. Оконные функции.

Студент Закиров Р.Р.

Группа АС-20-1

Руководитель Мирошников А.И.

Липецк 2023 г.

**Часть 1**

Группировка и оконные функции

1. Добавить нумерацию строк для выборки.

2. Пронумеровать строки в каждой группе (например, отдельно спортсменов, принимающих участие в соревнованиях по шахматам, отдельно – по настольному теннису и т.д.).

3. Составить таблицу транзакций с отражением номера операции, суммы, конечного баланса (за транзакцию рассматривать, например, покупку/возврат в аптеке, покупку/возврат абонемента, покупку/возврат билета).

4. Дополнить таблицу с транзакциями дополнительными столбцами (например, процент от общей суммы и т.д.).

5. Модифицировать запрос из п.4 с использованием WINDOW для одинаковых выражений.

6. Отфильтровать результаты запроса из пункта 5 (используя подзапрос). Например, оставить транзакции, при которых баланс сета становился отрицательным.

**Ход работы**

1. Добавить нумерацию строк для выборки:

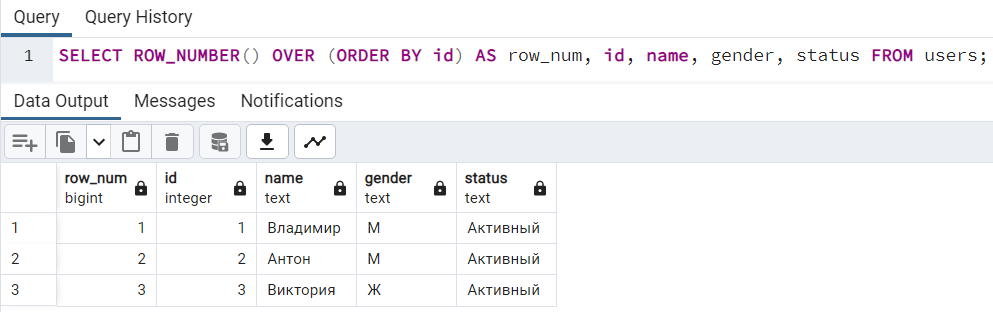


Рис.1. Добавление нумерации строк для выборки

1. Пронумеровать строки в каждой группе:

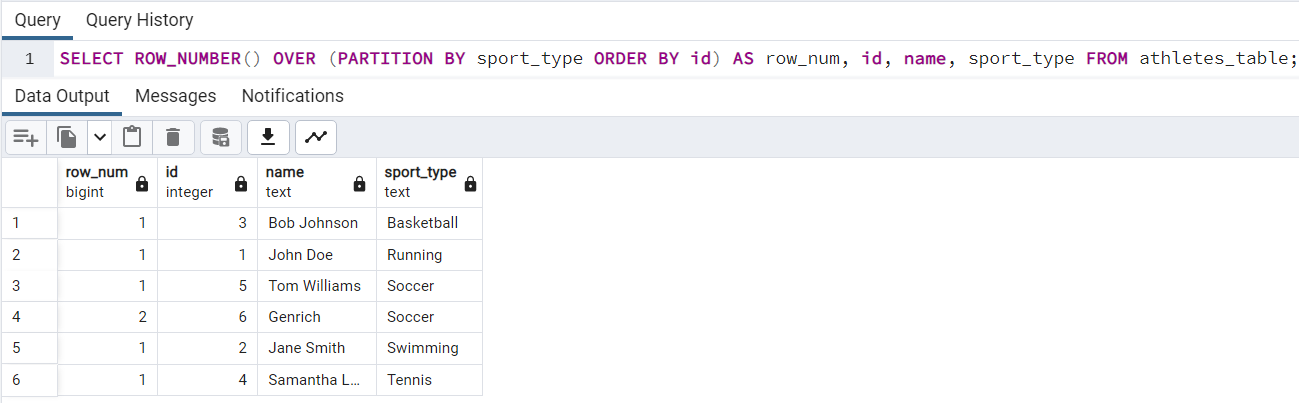


Рис.2. Нумерация строк в каждой группе

1. Составить таблицу транзакций:

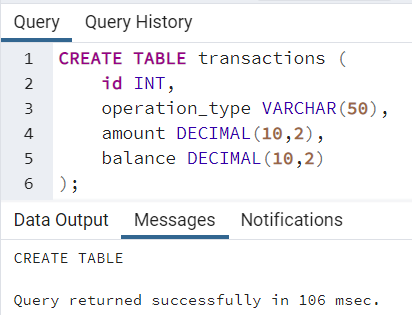


Рис.3. Составление таблицы транзакций

1. Дополнить таблицу с транзакциями дополнительными столбцами:

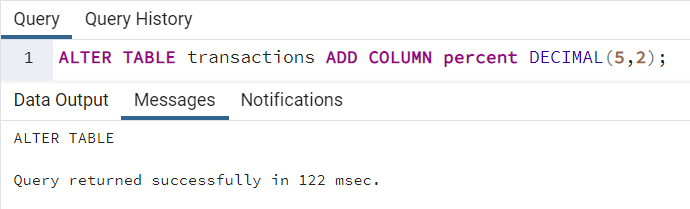


Рис.4. Дополнение таблицы с транзакциями дополнительными столбцами

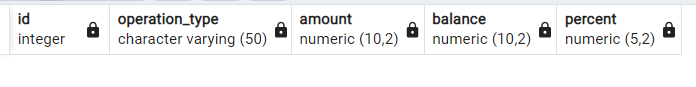


Рис.5. Результат дополнения таблицы

1. Модифицировать запрос из п.4 с использованием WINDOW:

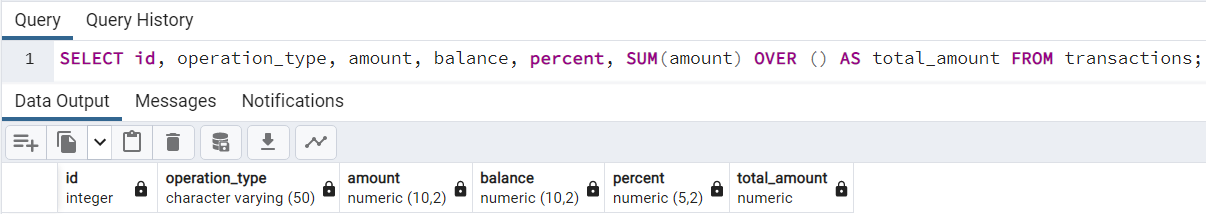


Рис.6. Модификация запроса

1. Отфильтровать результаты запроса из пункта 5:

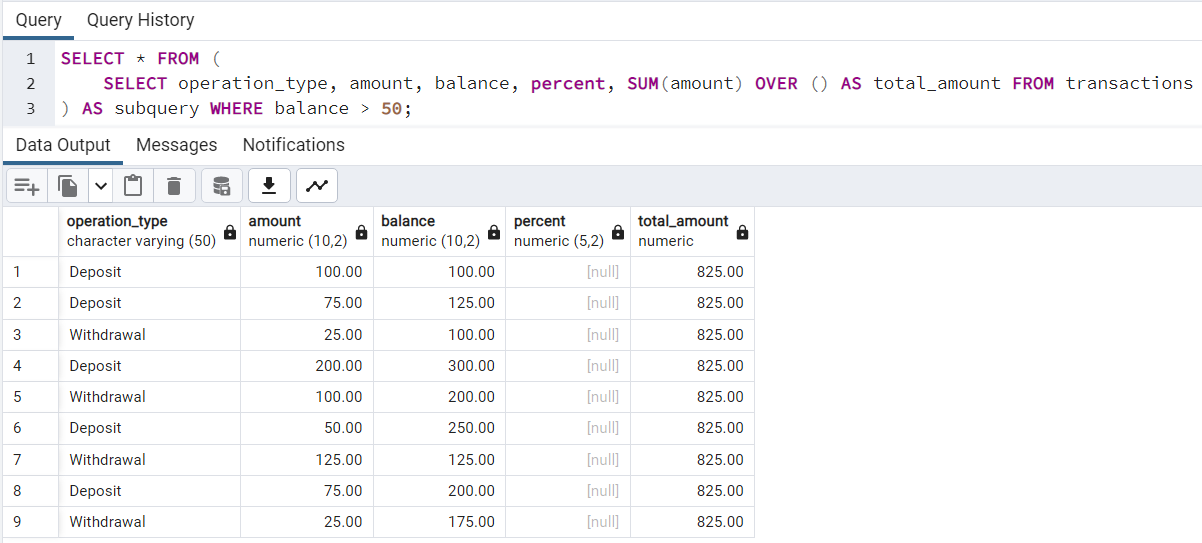
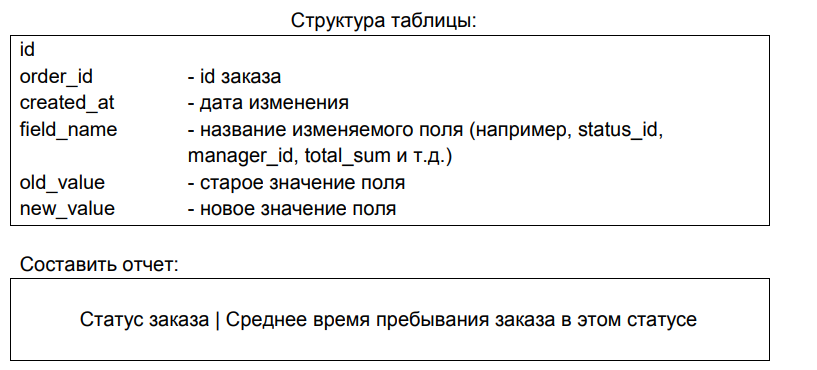


Рис.7. Отфильтрованный результат запроса из пункта 5

**Часть 2**

**Задача 3.1.** История изменения заказа

В таблице order\_history хранится история изменений все полей заказа



**Ход работы**

1. Создаем таблицу order\_history

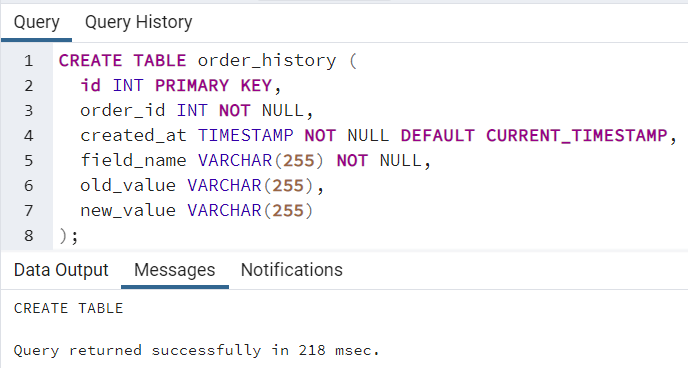


Рис.8. Создание таблицы

1. Заполнение строк таблицы



Рис.8. Заполнение строк таблицы

1. Составление отчета

“Статус заказа | Среднее время пребывания заказа в этом статусе”

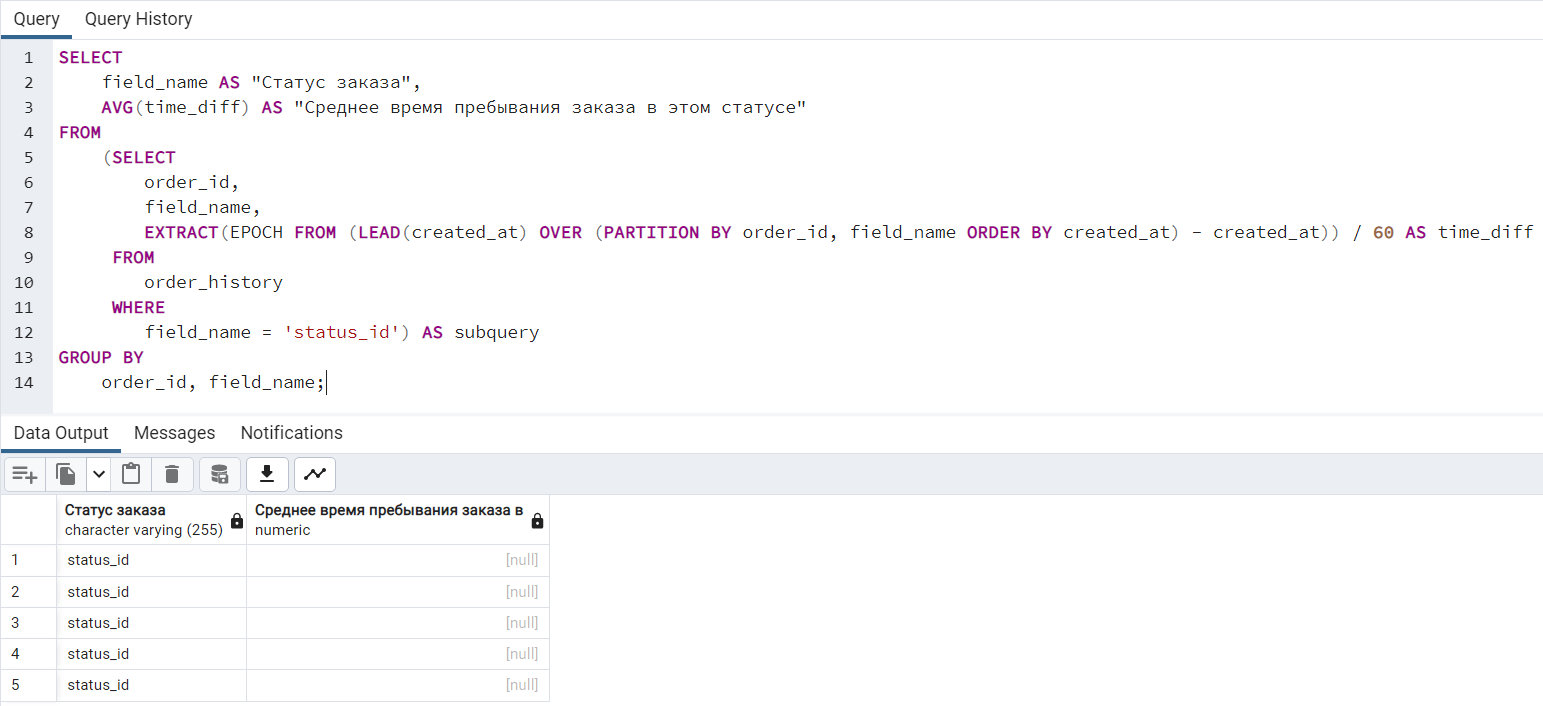
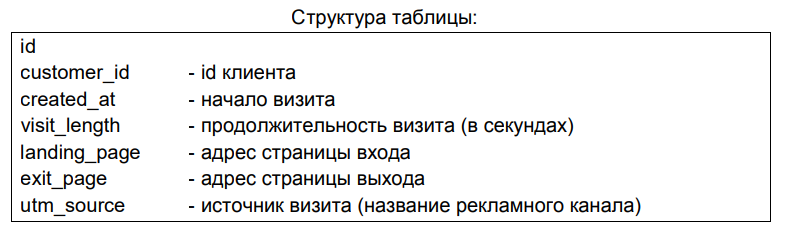


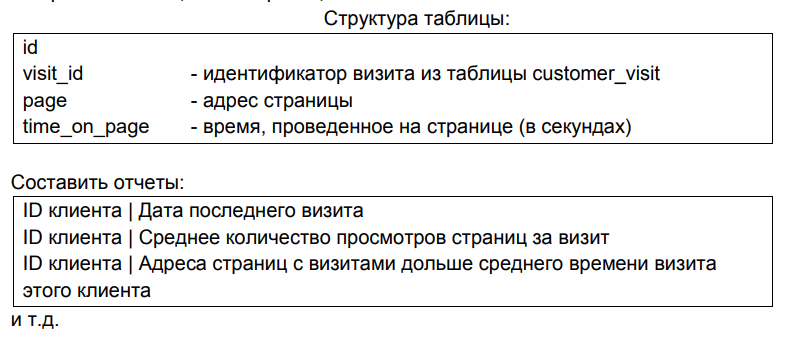
Рис.9. Отчет: Статус заказа | Среднее время пребывания заказа в этом статусе

**Задача 3.2.** Визиты клиентов

В таблице customer\_visit хранятся визиты клиентов на сайте, полученные из системы Google Analytics



Дополнительно есть таблица customer\_visit\_page, в которой хранятся конкретные посещенные страницы.



**Ход работы**

1. Создаем таблицу customer\_visit и customer\_visit\_page



Рис.10. Создание таблиц

1. Заполнение строк таблицы



Рис.11. Заполнение строк таблицы

1. Составить отчет: ID клиента | Дата последнего визита

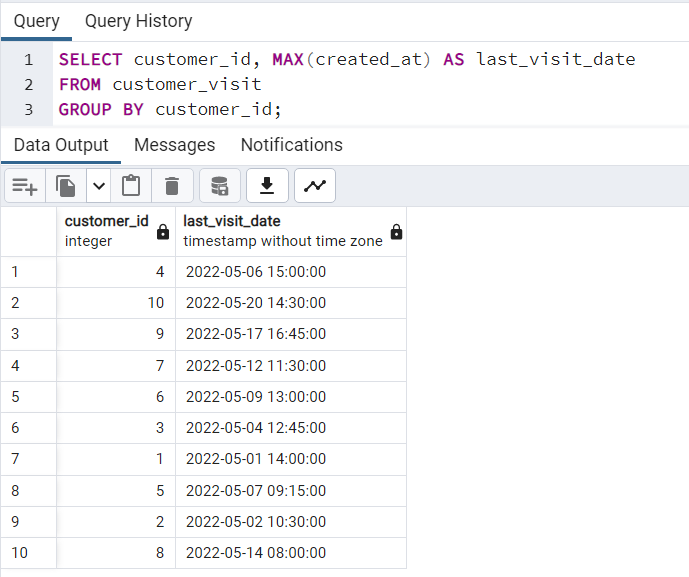


Рис.12. Отчет: ID клиента | Дата последнего визита

Этот запрос группирует записи в customer\_visit по customer\_id, а затем для каждой группы выбирает максимальное (самое последнее) значение created\_at, которое соответствует дате последнего визита клиента. Набор результатов будет иметь два столбца: customer\_id и last\_visit\_date.

1. Составить отчет: ID клиента | Среднее количество просмотров страниц за визит

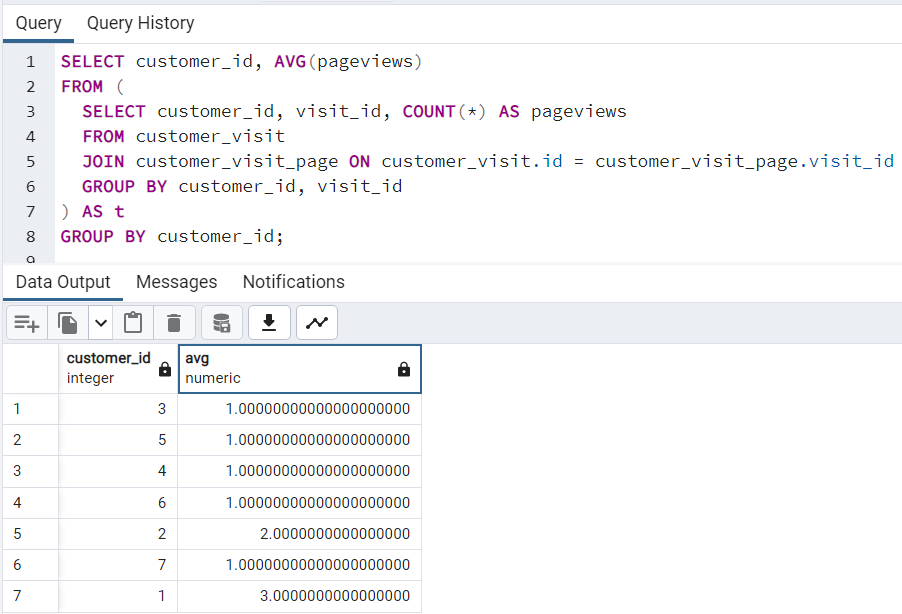


Рис.13. Отчет: ID клиента | Среднее количество просмотров страниц за визит

Этот запрос объединяет таблицы customer\_visit и customer\_visit\_page в столбцах id и visit\_id соответственно, а затем группирует посещения по клиенту и идентификатору посещения, чтобы подсчитать количество просмотров страниц для каждого посещения с помощью функции COUNT(\*). Затем он вычисляет среднее количество просмотров страниц за посещение для каждого клиента с помощью функции AVG() и возвращает результаты, сгруппированные по идентификатору клиента.

1. Составить отчет: ID клиента | Адреса страниц с визитами дольше среднего времени визита этого клиента

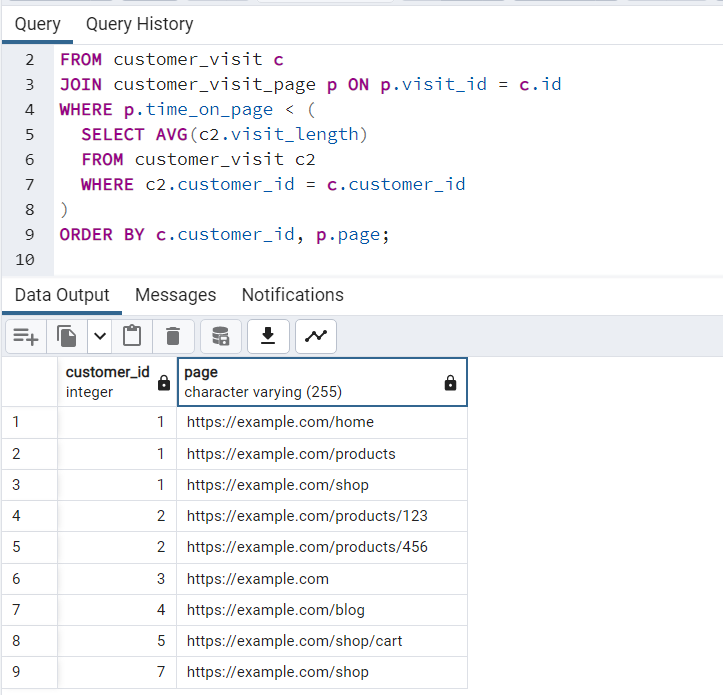


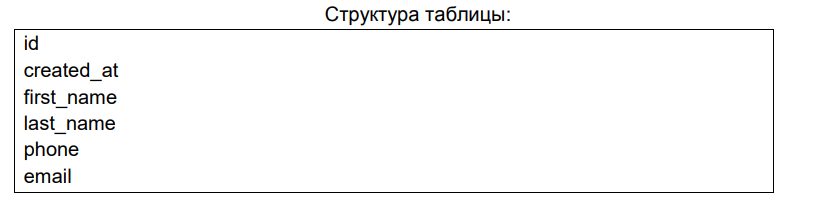
Рис.14. Отчет: ID клиента | Адреса страниц с визитами дольше среднего времени визита этого клиента

Этот запрос объединяет таблицы customer\_visit и customer\_visit\_page в столбце идентификатора посещения, а затем фильтрует строки, в которых время, проведенное на каждой странице, превышает среднюю продолжительность посещения для этого клиента. Подзапрос вычисляет среднюю продолжительность посещения для каждого клиента с помощью функции AVG(), а затем фильтрует строки с предложением WHERE, которое проверяет, превышает ли продолжительность посещения для текущей строки среднее значение для клиента.

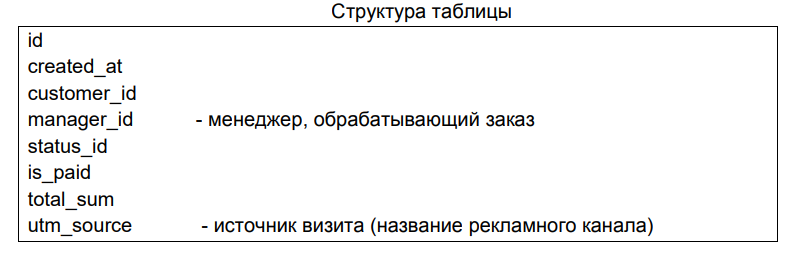
Предложение SELECT в запросе возвращает идентификатор клиента и страницы, посещенные этим клиентом, которые имели большую продолжительность, чем в среднем. Предложение ORDER BY сортирует результаты по идентификатору клиента и посещенной странице.

**Задача 3.3.** Расчет конверсии

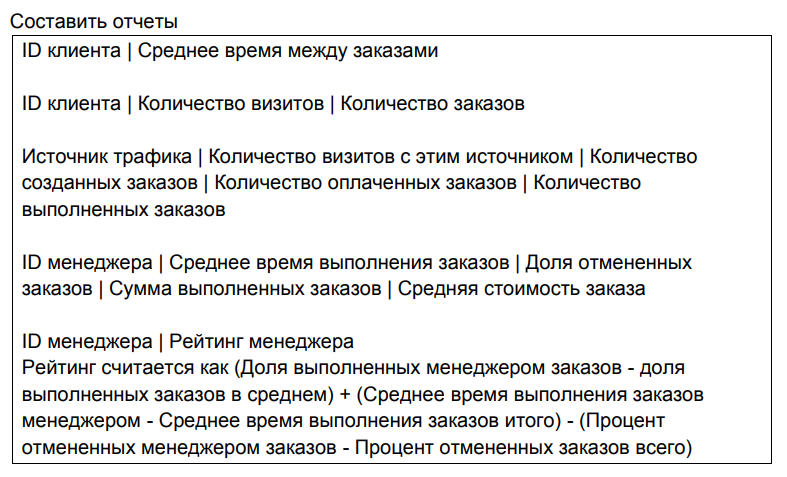
Есть таблицы с визитами, описанные выше. Также есть таблица customer, содержащая данные о клиенте



И таблица order, содержащая заказы:



Заказ считается выполненным, если находится в статусе “Выполнен” (с определенным id).



**Ход работы**

1. Создаем таблицу customer и order

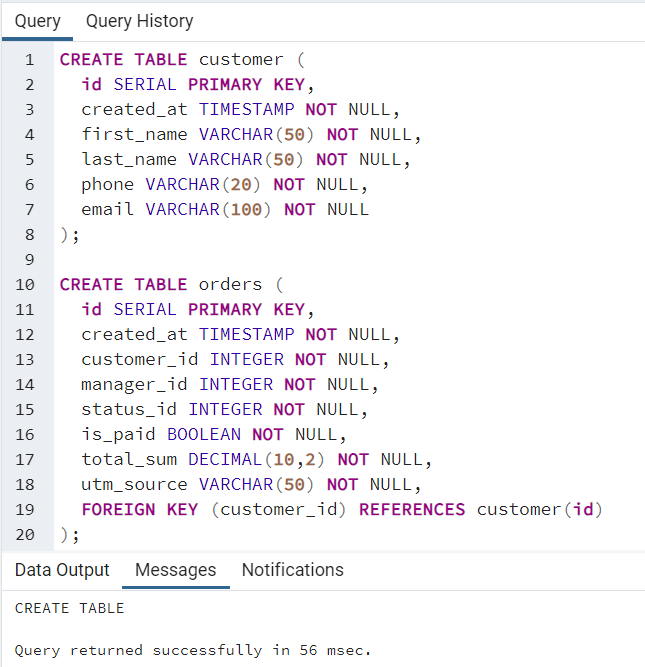


Рис.15. Создание таблиц

1. Заполнение строк таблицы



Рис.16. Заполнение строк таблицы

1. Составить отчет: ID клиента | Среднее время между заказами



Рис.17. Отчет: ID клиента | Среднее время между заказами

Запрос объединяет таблицы «клиент» и «заказы» и использует боковое соединение для поиска предыдущего заказа для каждого заказа каждого клиента. Затем он вычисляет разницу во времени между заказами и вычисляет среднее время между заказами для каждого клиента. Результат группируется по идентификатору клиента.

Обратите внимание, что разница во времени рассчитывается в секундах, а затем преобразуется в дни путем деления на 60 \* 60 \* 24. При необходимости вы можете изменить этот коэффициент преобразования, чтобы показать разницу во времени в других единицах измерения.

1. Составить отчет: ID клиента | Количество визитов | Количество заказов

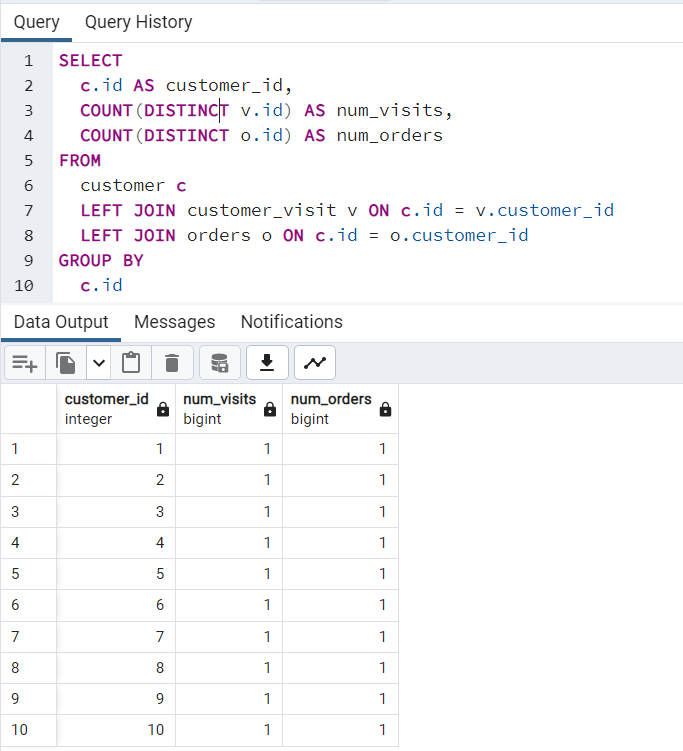


Рис.18. Отчет: ID клиента | Количество визитов | Количество заказов

Запрос объединяет таблицы «customer», «customer\_visit» и «orders» с помощью левых соединений, чтобы убедиться, что в результат включены все клиенты, даже если у них нет посещений или заказов. Затем он подсчитывает количество различных посещений и заказов для каждого клиента и группирует результат по идентификатору клиента.

Ключевое слово `DISTINCT` используется в функции `COUNT`, чтобы гарантировать, что каждое посещение и заказ учитываются только один раз. Без `DISTINCT` запрос будет подсчитывать несколько посещений и заказов для одного и того же клиента несколько раз.

1. Составить отчет: Источник трафика | Количество визитов с этим источником | Количество созданных заказов | Количество оплаченных заказов | Количество выполненных заказов

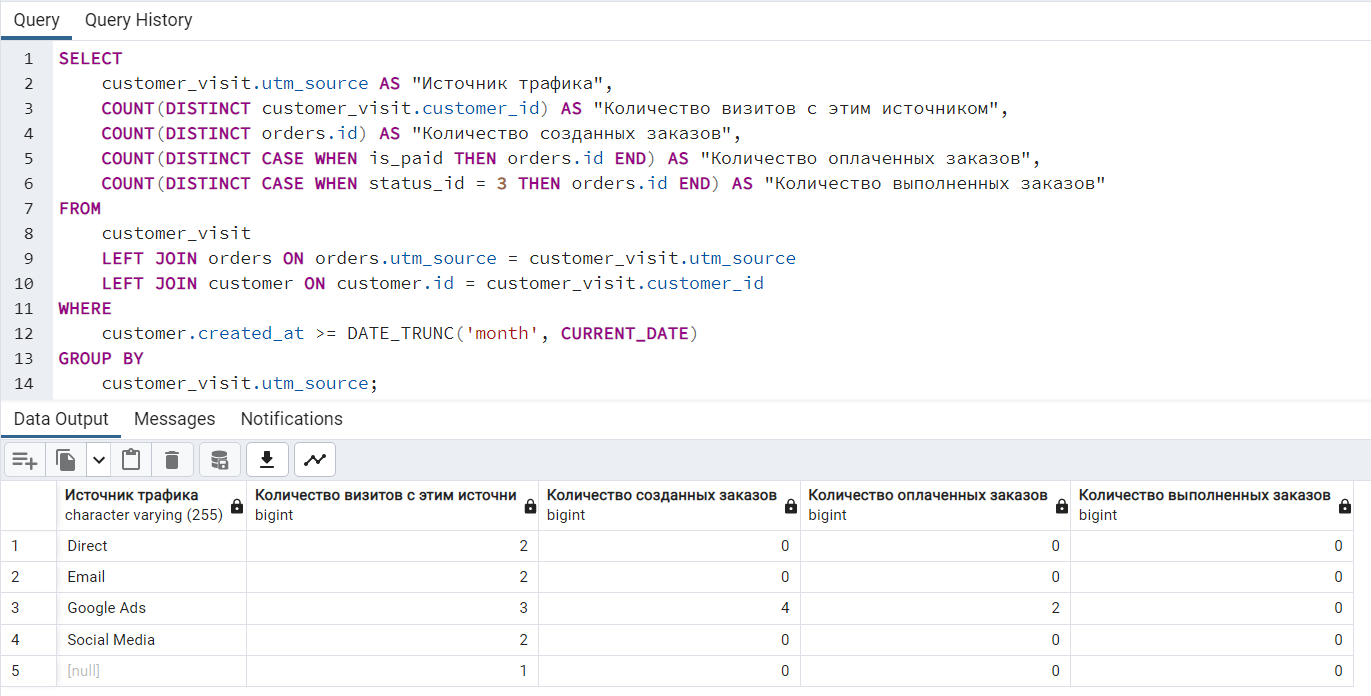


Рис.19. Отчет: Источник трафика | Количество визитов с этим источником | Количество созданных заказов | Количество оплаченных заказов | Количество выполненных заказов

1. Составить отчет: ID менеджера | Среднее время выполнения заказов | Доля отмененных заказов | Сумма выполненных заказов | Средняя стоимость заказа

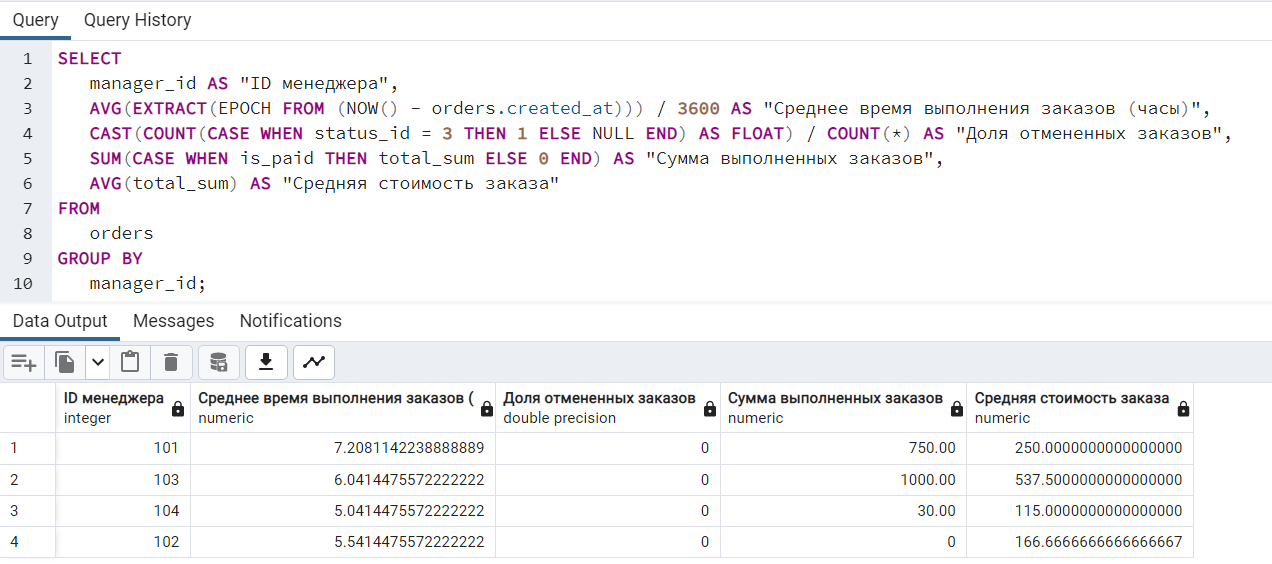


Рис.20. Отчет: ID менеджера | Среднее время выполнения заказов | Доля отмененных заказов | Сумма выполненных заказов | Средняя стоимость заказа

1. Составить отчет: ID менеджера | Рейтинг менеджера

Рейтинг считается как (Доля выполненных менеджером заказов - доля выполненных заказов в среднем) + (Среднее время выполнения заказов менеджером - Среднее время выполнения заказов итого) - (Процент отмененных менеджером заказов - Процент отмененных заказов всего)

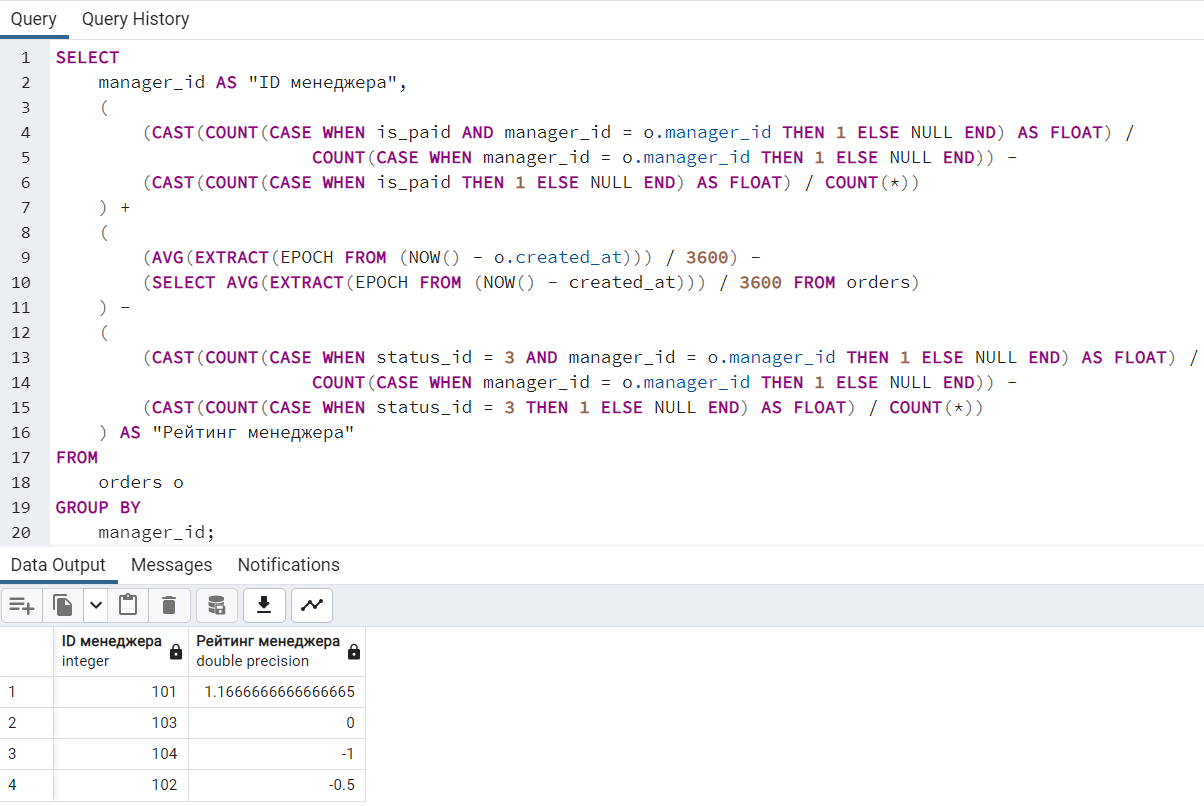


Рис.21. Отчет: ID менеджера | Рейтинг менеджера

Этот запрос рассчитает рейтинг каждого менеджера на основе предоставленной формулы и сгруппирует результаты по manager\_id. В полученной таблице будет две колонки: «ID менеджера» и «Рейтинг менеджера».