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

**Институт информационных технологий (ИИТ)**

**Кафедра цифровой трансформации (ЦТ)**

**ОТЧЕТ ПО ПРАКТИЧЕСКОЙ РАБОТЕ**

по дисциплине «Проектирование баз данных»

**Практическое занятие №1**

|  |  |  |  |
| --- | --- | --- | --- |
| Студенты группы ИНБО-01-17 | *ИКБО-20-23 Кузнецов Лев Андреевич* | (подпись) | |
| Ассистент | *Брайловский А.В.* | (подпись) | |
| Отчет представлен | «\_\_\_»\_\_\_\_\_\_\_\_2025 г. | |  | |

Москва 2025 г.

# СОДЕРЖАНИЕ

[постановка задачи 3](#_постановка_задачи)

[ход работы 4](#_ход_работы_1)

[ОТВЕТЫ НА ВОПРОСЫ 12](#_ОТВЕТЫ_НА_ВОПРОСЫ)

[ВЫВОД 13](#_ВЫВОД)

# ПОСТАНОВКА ЗАДАЧИ

**Цель**: сформировать и закрепить навыки работы с реляционными базами данных на примере СУБД PostgreSQL.

**Постановка задачи**: на основе логической модели данных, созданной в предыдущем семестре, выполнить необходимые шаги.

# ход работы

1. Анализ и описание ограничений целостности

Для данного пункта были составлены две таблицы на примере двух других таблиц: request и app\_user, - из логической модели из предыдущего семестра (таблицы 1-2).

*Таблица 1. Описание ограничений для таблицы request*

|  |  |  |  |
| --- | --- | --- | --- |
| **Название**  **столбца** | **Тип данных** | **Ограничение** | **Обоснование (Бизнес-**  **правило)** |
| id\_request | SERIAL | PRIMARY KEY | Уникальный идентификатор заявки,  генерируется  автоматически. |
| id\_app\_user | INT | FOREIGN KEY (app\_user) | Ссылка на пользователя. Заявка не может существовать без пользователя. |
| id\_transaction | INT | FOREIGN KEY (transaction) | Ссылка на транзакцию. Заявка не может существовать без транзакции |
| description | TEXT | NOT NULL | Описание заявки, включающее в себя всю необходимую информацию о приобретаемом курсе. В каждой заявке обязательное присутствует непустое поле описания. |

*Таблица 2. Описание ограничений для таблицы app\_user*

|  |  |  |  |
| --- | --- | --- | --- |
| **Название**  **столбца** | **Тип данных** | **Ограничение** | **Обоснование (Бизнес-**  **правило)** |
| id\_app\_user | SERIAL | PRIMARY KEY | Уникальный идентификатор заявки,  генерируется  автоматически. |
| personal\_data | TEXT | NOT NULL | Личная информация пользователя. Для использования приложения пользователь обязан ввести требуемые личные данные, по этой причине данный столбец не может не иметь значения. |

1. Создание структуры данных

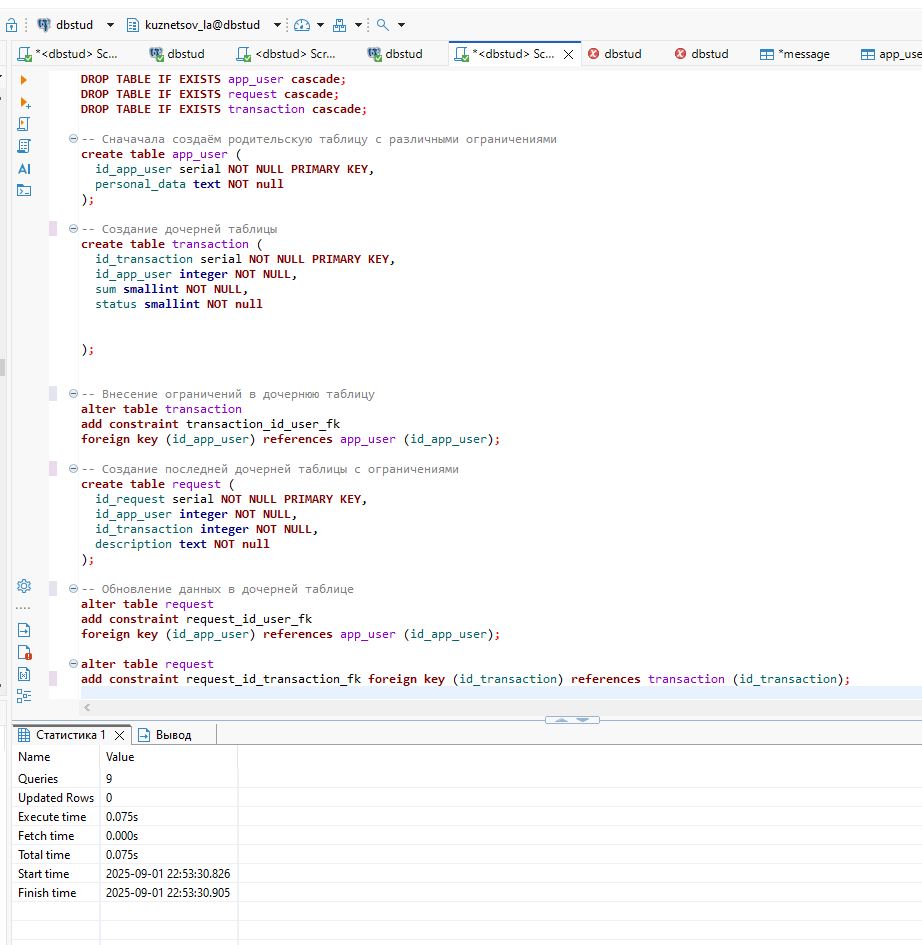


Рисунок 1 – Кодовый вид запроса и итоговый вывод

3. Заполнение таблиц данными (DML – Data Manipulation Language)

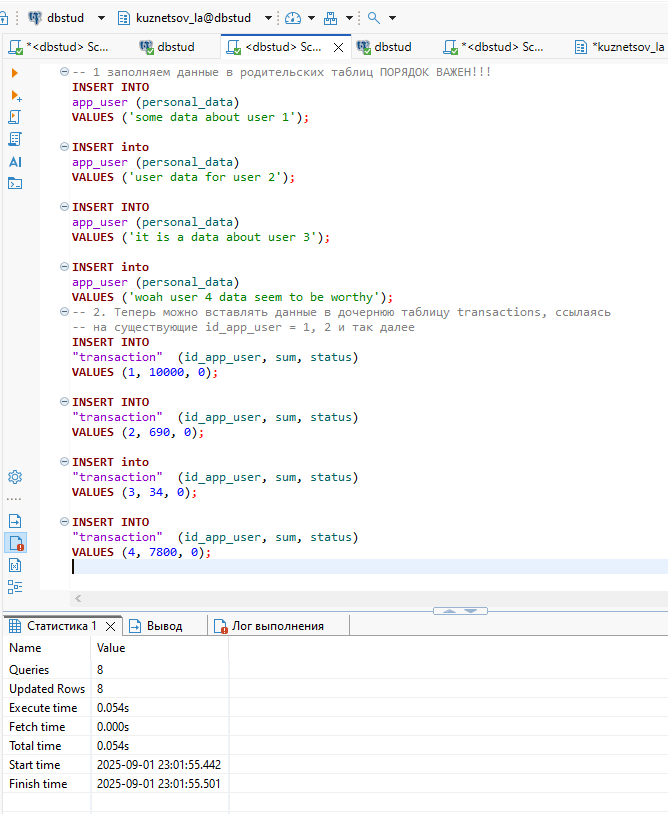


Рисунок 2 – Заполнение данных в итоговых таблицах

4. Составление запросов на выборку

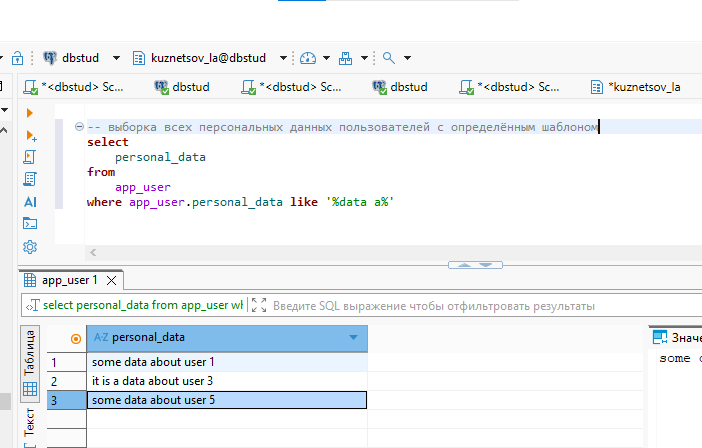


Рисунок 3 – Выборка при помощи %

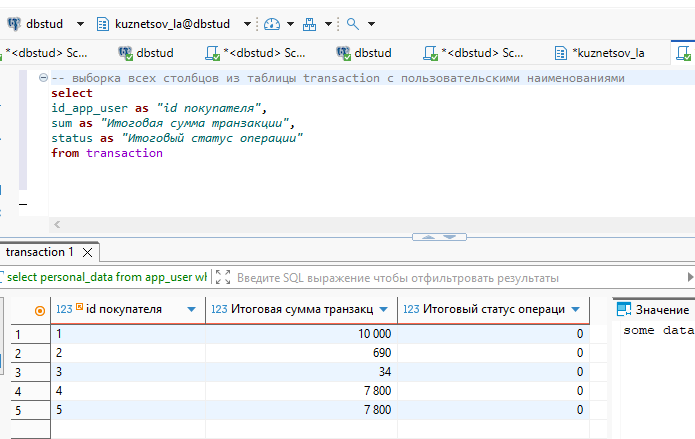


Рисунок 4 – Пользовательское именование столбцов

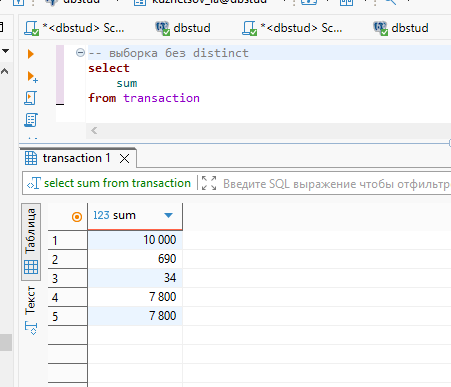


Рисунок 5 – Выборка по столбцу

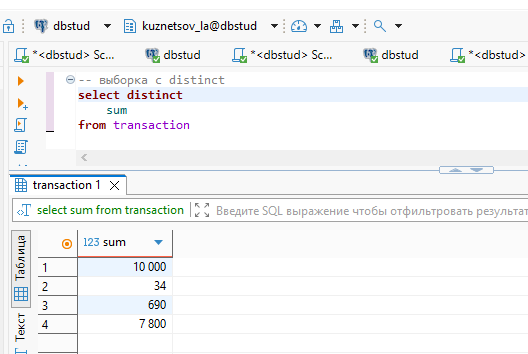


Рисунок 6 – Выборка по неповторяющемуся столбцу

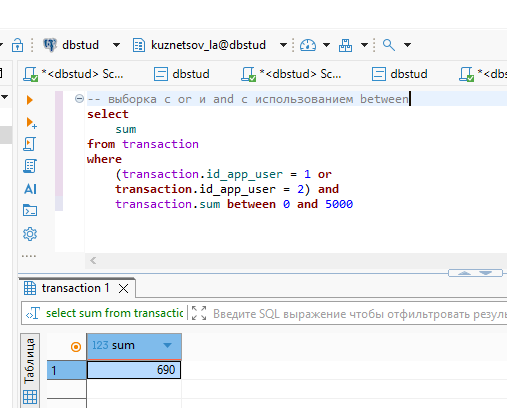


Рисунок 7 – Выборка с использованием or, and и between

На рисунке 7 представлена выборка с использованием between, что наглядно даёт понять принцип работы таких ключевых слов как min и max.

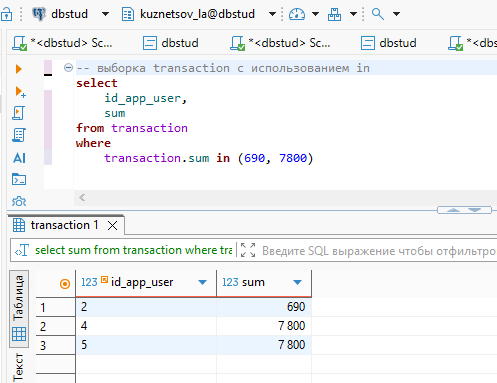


Рисунок 8 – Выборка с использованием in

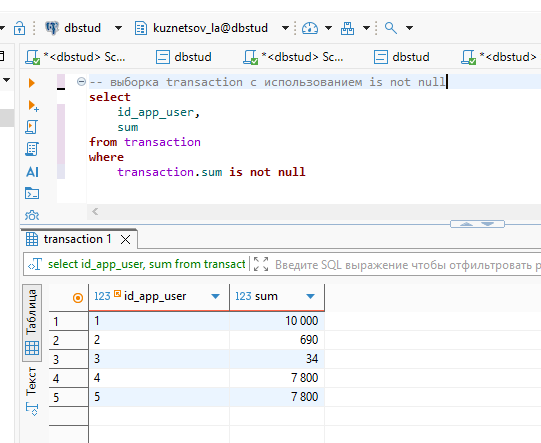


Рисунок 9 – Выборка при помощи where с is not null

5. Составление запросов на выборку

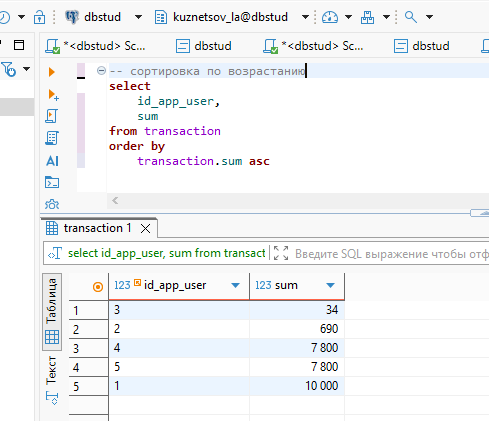


Рисунок 10 – Сортировка по возрастанию (ключевое слово - asc)

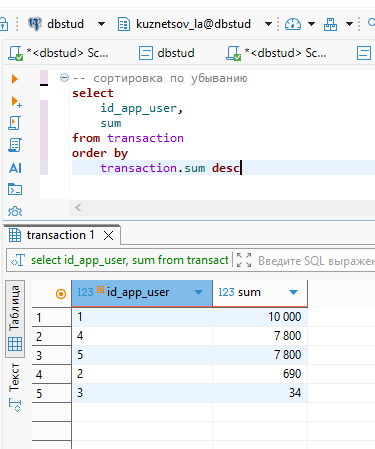


Рисунок 11 - Сортировка по убыванию (ключевое слово - desc)

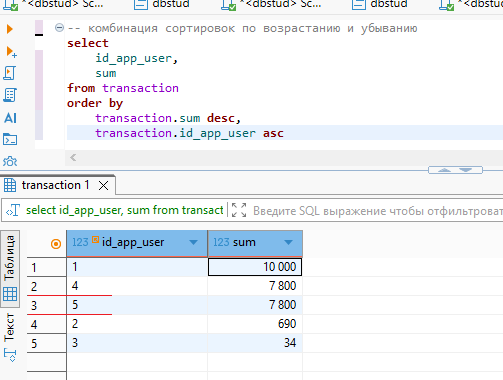


Рисунок 12 – Комбинация сортировок desc и asc

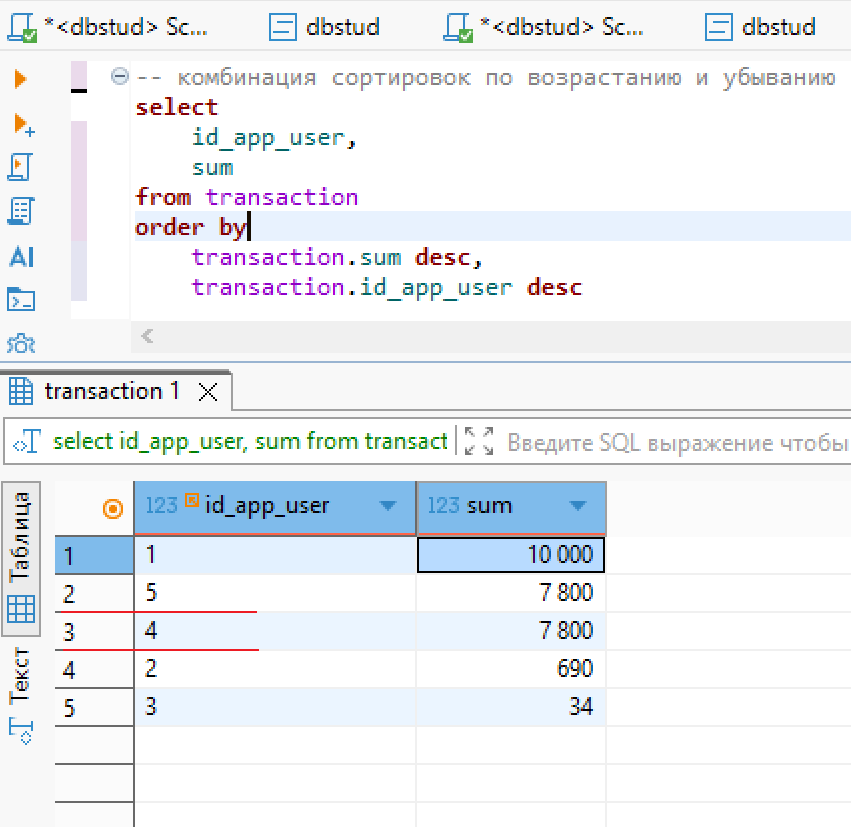


Рисунок 13 - Комбинация сортировок desc и desc

Как видно из рисунков 12-13 сначала производится сортировка по sum и при обнаружении дубликатов в sum далее будет проведена сортировка по id\_app\_user (обратите внимание на 2-ую и 3-ю строки в id\_app\_user в окне вывода на обоих рисунках).

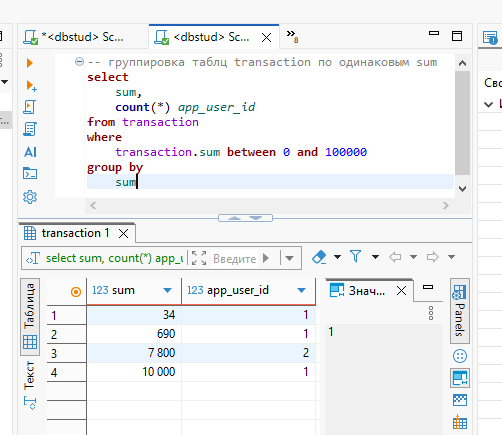


Рисунок 14 – Группировка таблиц по app\_user\_id

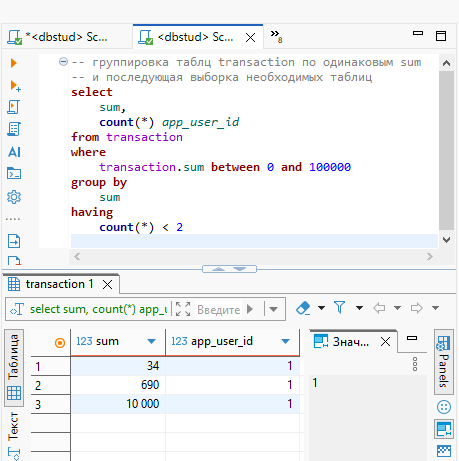


Рисунок 15 - Группировка таблиц по app\_user\_id с дополнительным условием

# ОТВЕТЫ НА ВОПРОСЫ

1. Отличие в том, что primary key – это комбинация unique и not null. primary key не допускает значения null, unique - допускает;
2. Дочерняя таблица по данному foreign key удалит все свои связанные дочерние таблицы;
3. from → where → group by → having → select → order by;
4. Потому что нельзя обратить к Nil псевдониму;
5. Золотое правило group by – любой столбец указанный в select должен находиться либо в group by, либо в агрегатной функции. Потому что так СУБД сможет сгруппировать столбцы по контролируемому правилу.
6. Отсортировать transaction по count(\*) app\_user\_id Найти все transaction с sum в пределах от 100 до 1000 и при этом итоговое кол-во id\_app\_user должно быть меньше 2.

# ВЫВОД

В ходе работы были получены и закреплены практическим путём знания и навыки по работе с СУБД.