Департамент образования и науки города Москвы

Государственное автономное образовательное учреждение высшего образования города Москвы

«Московский городской педагогический университет»

Институт цифрового образования

Департамент информатики, управления и технологий

ДИСЦИПЛИНА:

Инструменты для хранения и обработки больших данных

**Лабораторная работа №4.1**

**Тема:**

«Сравнение подходов хранения больших данных»

Выполнил: Егоров А. С., группа: АДЭУ-211

Преподаватель: Босенко Т. М.

Москва

2024

СОДЕРЖАНИЕ

[ВВЕДЕНИЕ 3](#_Toc183106475)

[ТЕОРЕТИЧЕКСАЯ ЧАСТЬ 4](#_Toc183106476)

[MongoDB 4](#_Toc183106477)

[PostgreSQL 4](#_Toc183106478)

[ПРАКТИЧЕСКАЯ ЧАСТЬ 6](#_Toc183106479)

[Индивидуальное задание 9](#_Toc183106480)

[ЗАКЛЮЧЕНИЕ 14](#_Toc183106481)

# ВВЕДЕНИЕ

В современном мире объемы данных растут экспоненциально, что приводит к необходимости использования эффективных методов их хранения и обработки. Существует два основных подхода к хранению больших данных:

1. Реляционные базы данных (например, PostgreSQL)

2. NoSQL базы данных (например, MongoDB)

Каждый из этих подходов имеет свои преимущества и недостатки, которые мы рассмотрим в ходе выполнения лабораторной работы.

Цель работы: сравнить производительность и эффективность различных подходов к хранению и обработке больших данных на примере реляционной базы данных PostgreSQL и документоориентированной базы данных MongoDB.

* Оборудование и программное обеспечение
* Компьютер с операционной системой Ubuntu.
* PostgreSQL.
* MongoDB.
* Python 3.x.
* Библиотеки: psycopg2, pymongo, pandas, matplotlib.

# ТЕОРЕТИЧЕКСАЯ ЧАСТЬ

Объем данных с каждым годом увеличивается, поэтому нужны\ы эффективные инструменты их хранения. Рассмотрим два популярных подхода к управлению данными: MongoDB (документо-ориентированная база данных) и PostgreSQL (реляционная база данных).

MongoDB — документоориентированная система управления базами данных, не требующая описания схемы таблиц. Считается одним из классических примеров NoSQL-систем, использует JSON-подобные документы и схему базы данных. Написана на языке C++.

Некоторые особенности MongoDB:

* Кроссплатформенность. Работает в операционных системах Windows, macOS и разных дистрибутивах Linux.
* Реплицирование. Данные в MongoDB не хранятся в одном месте — они поделены между разными серверами. Один из серверов ключевой, первичный, и на нём выполняются все основные операции. Другие серверы вторичные, на них копируют записи из основного.
* Система специальных запросов. Данные можно искать по полям и диапазонам, поддерживаются регулярные выражения, можно возвращать пользовательские функции.

PostgreSQL — это объектно-реляционная система управления базами данных (СУБД) с открытым исходным кодом. Она помогает создавать и хранить базы данных, а также работать с ними на языке SQL (Structured Query Language).

Некоторые особенности PostgreSQL:

* Поддержка стандарта SQL. Это значит, что команды и запросы для других баз данных будут работать и в PostgreSQL.
* Гибкость. Пользователи могут создавать функции, операторы, типы данных и индексные методы.
* Масштабируемость. PostgreSQL поддерживает горизонтальное и вертикальное масштабирование — можно работать с большими объёмами данных и с большим числом пользователей.
* Надёжность. Система предоставляет несколько уровней защиты данных и механизмов восстановления после сбоев, что гарантирует минимальные потери данных и стабильную работу системы.

# ПРАКТИЧЕСКАЯ ЧАСТЬ

1. Подготовительный этап: подключаемся через удаленный рабочий стол к виртуальной машине.
2. Запускаем контейнеры на основе настроек, определенных в файле docker-compose.ym.

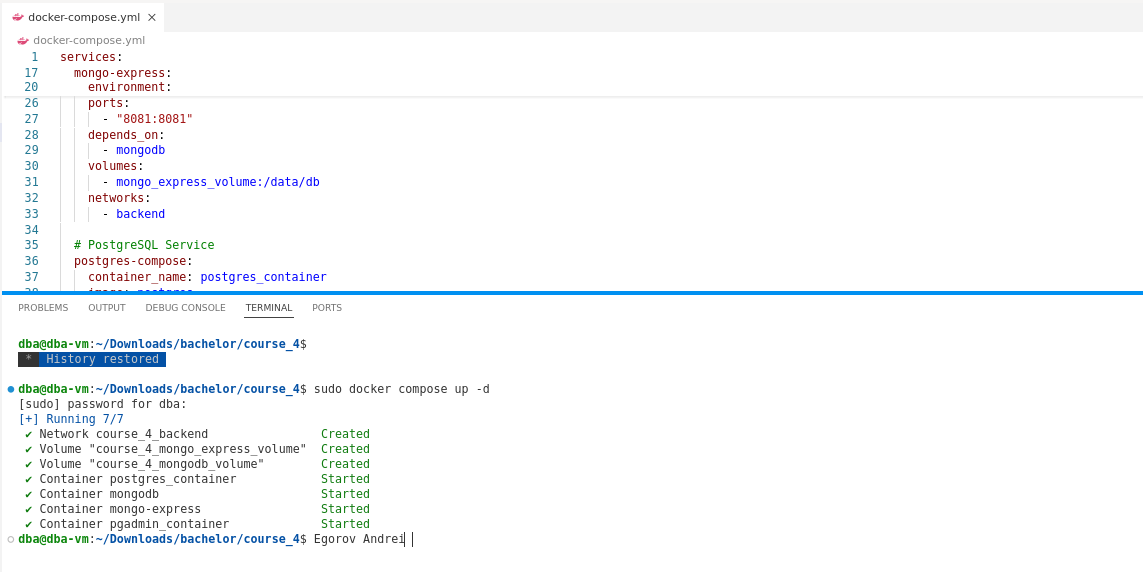


Рисунок 1 – Запуск контейнеров

1. Проверяем, что Mongo Express работает.

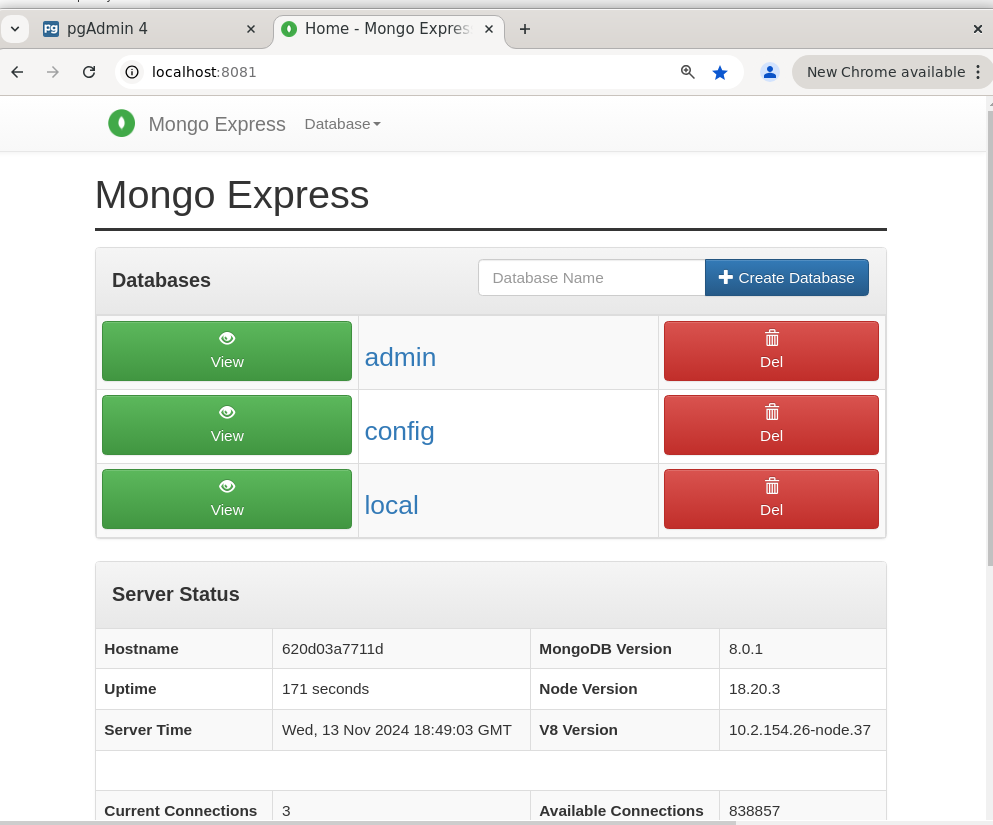


Рисунок 2 – Проверка работы Mongo Express

1. Проверяем, что PgAdmin работает.

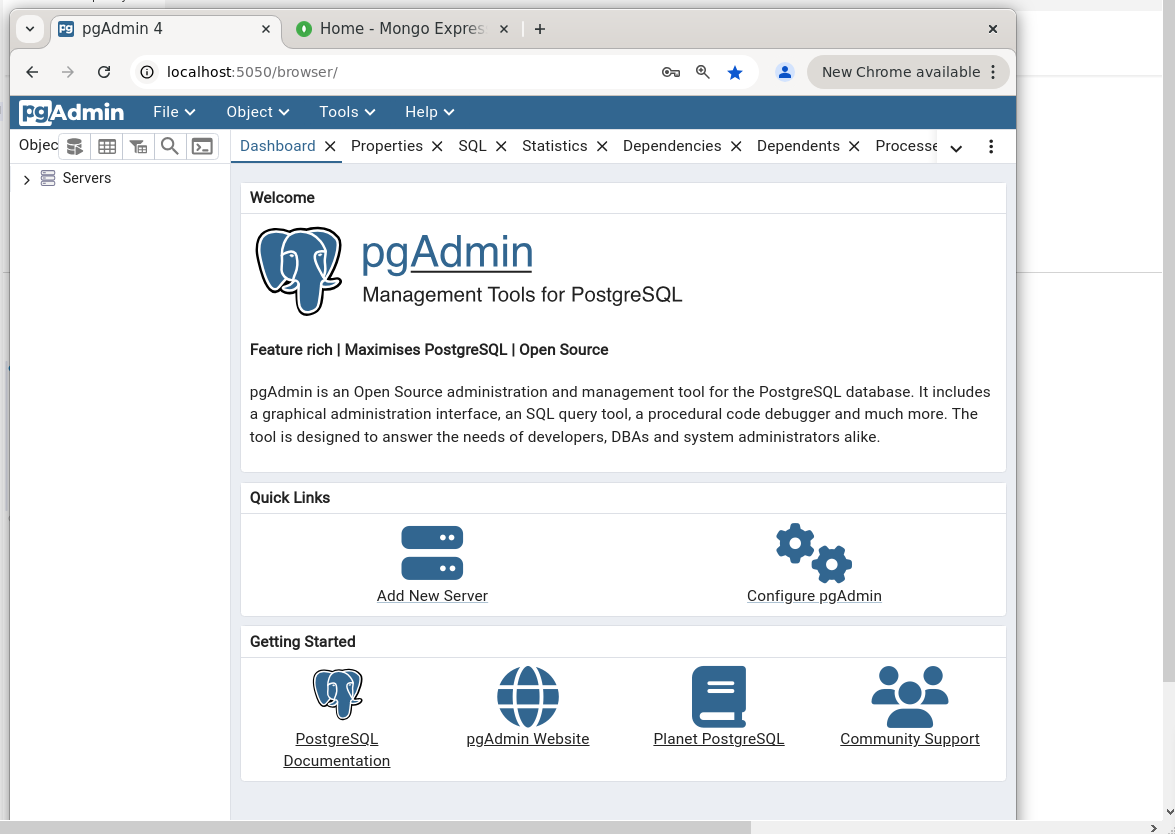


Рисунок 3 – Проверка работы PgAdmin

1. Создаем сервер EgorovAS.

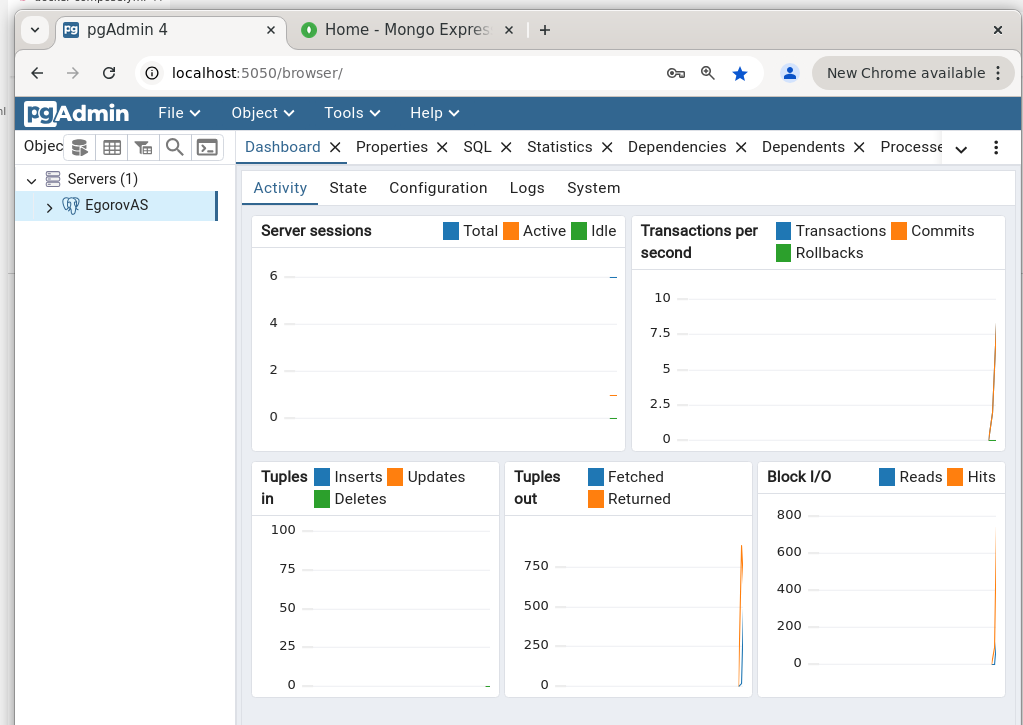


Рисунок 4 – Создание сервера EgorovAS

1. Добавляем в pgAdmin таблицу Sales.

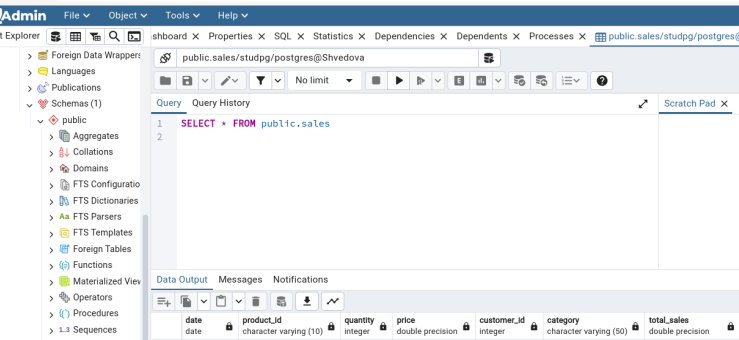
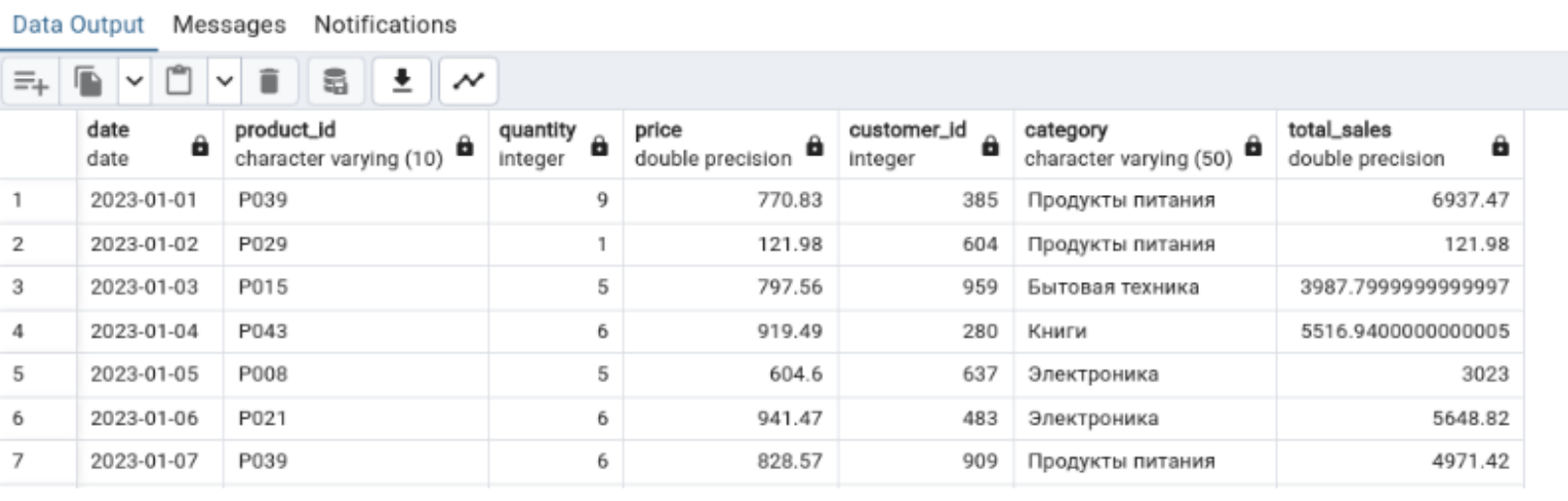


Рисунок 5 – Добавление таблицы sales

 Рисунок 6 – Содержание таблицы sales

## Индивидуальное задание

**Вариант 4. Оценить скорость вставки большого количества записей в базу данных транзакций финансовой организации.**

1. Сгенерируем фейковые данные согласно своему варианту. Для этого установим модуль faker и подключим другие необходимые библиотеки. Создадим таблицу из 1000 строк, состоящую из столбцов: id транзакции (transaction\_id); account\_from; account\_to; сумма (amount); валюта (currency) – рандомно выбор из доллара, евро и фунта; timestamp.

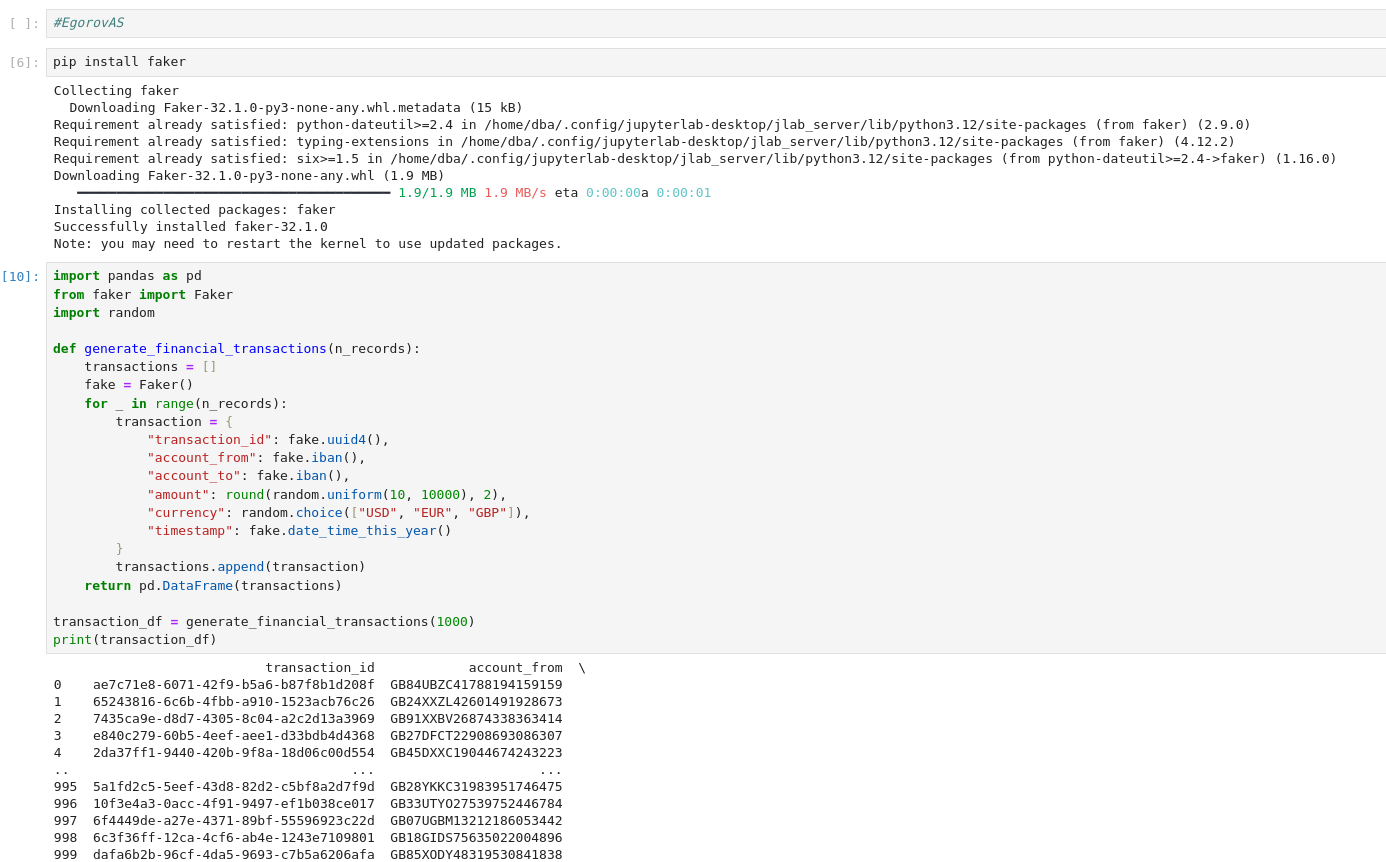


Рисунок 7 – Процесс генерации фейковых данных

1. Выгрузим таблицу в формате csv.

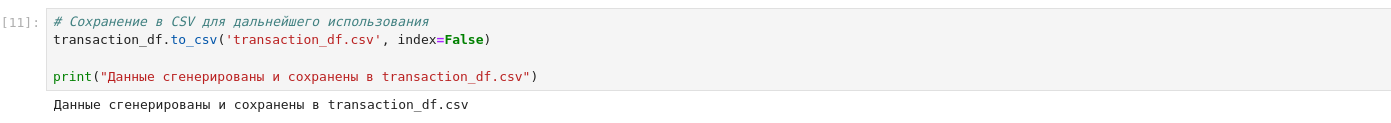


Рисунок 8 – Выгрузка таблицы

1. Загрузим сгенерированные данные в MongoDB. Как видно на рисунке ниже, загрузка прошла успешно.

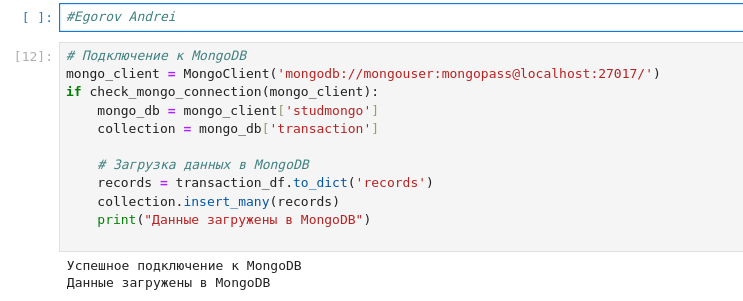


Рисунок 9 – Загрузка данных в MongoDB

1. Напишем функцию test\_mongodb, которая принимает количество записей n\_records и выполняет вставку этих записей в базу данных MongoDB и посмотрим, за какое время это все сделается. Итог: время вставки 10.000 записей в MongoDB: 2,9 секунд



Рисунок 10 – Определение времени вставки в MongoDB 10.000 записей

1. Определим время вставки в MongoDB 100.000 записей. Итог: время вставки 100000 записей в MongoDB: 19,85 секунд.



Рисунок 11 – Определение времени вставки в MongoDB 100.000 записей

1. Подключимся к PostreSQL и загрузим таблицу.

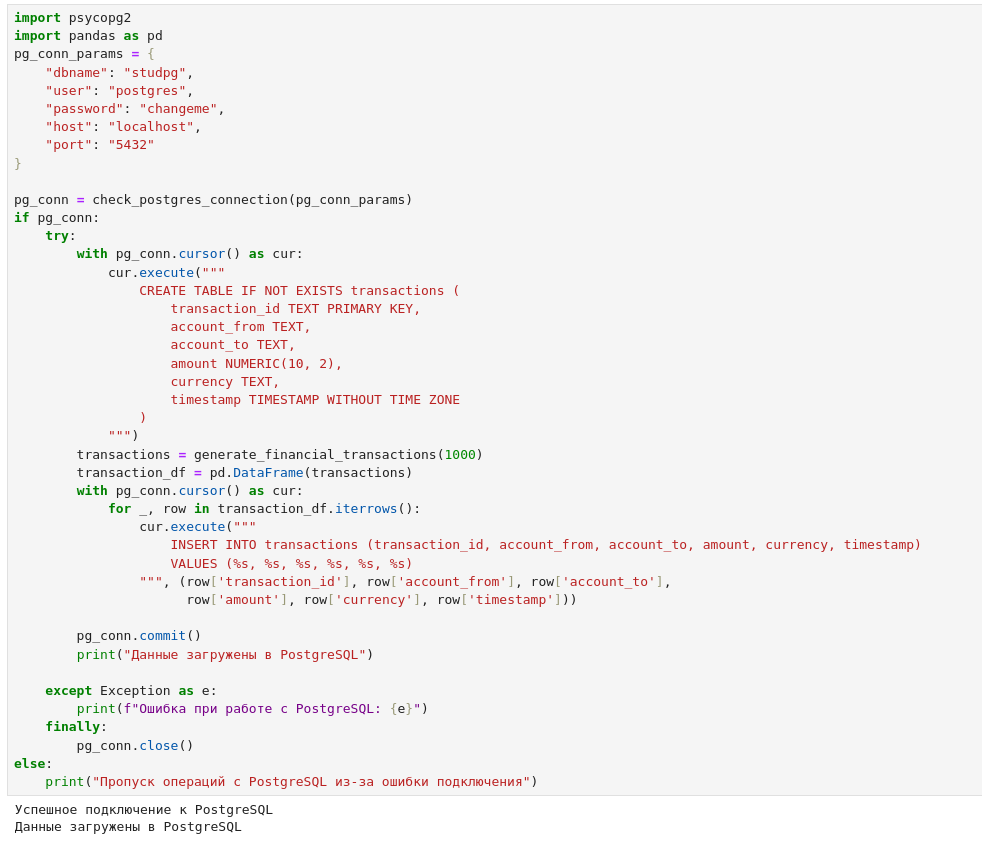


Рисунок 12 – Загрузка данных в PostreSQL

Проверяем, что таблица загрузилась в PostreSQL.

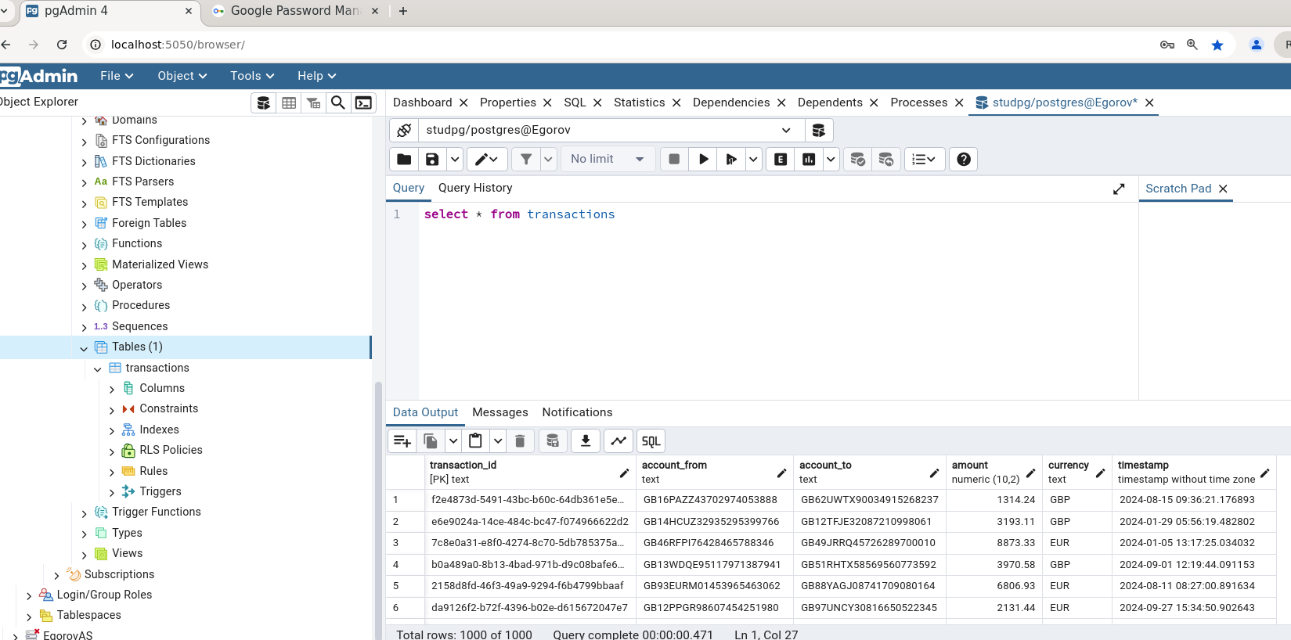


Рисунок 13 – Проверка загрузки таблицы в PostreSQL

Напишем функцию test\_postgresql, которая принимает количество записей n\_records и выполняет вставку этих записей в базу данных PostreSQL и посмотрим, за какое время это все сделается. Итог: время вставки 10000 записей в PostreSQL: 6,5 секунд.



Рисунок 14 – Определение времени вставки в PostreSQL 10000 записей

Время вставки 100000 записей в PostreSQL: 71,73 секунд.

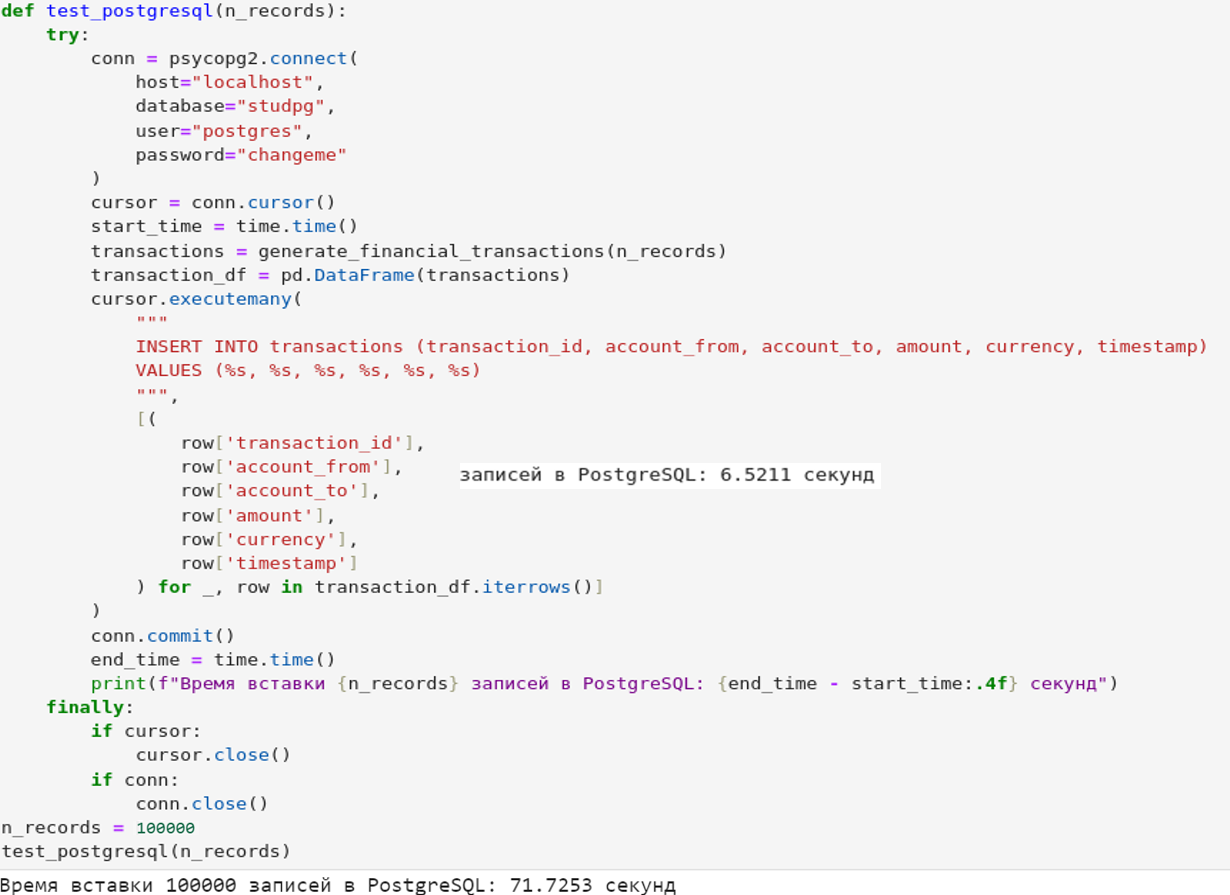


Рисунок 15 – Определение времени вставки в PostreSQL 100000 записей

Построим круговую диаграмму времени вставки записей в MongoDB и PostreSQL.

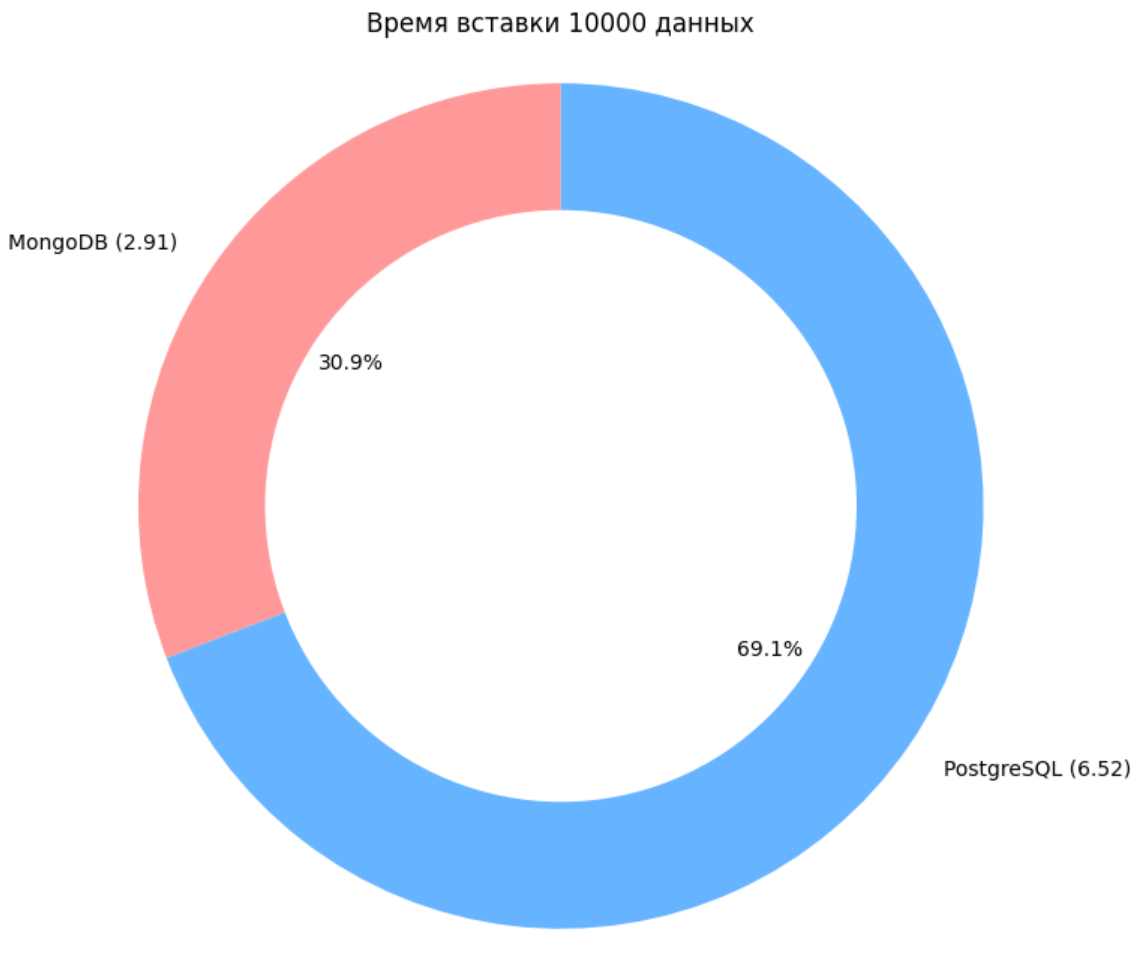


Рисунок 14 – Диаграмма вставки 10000 записей

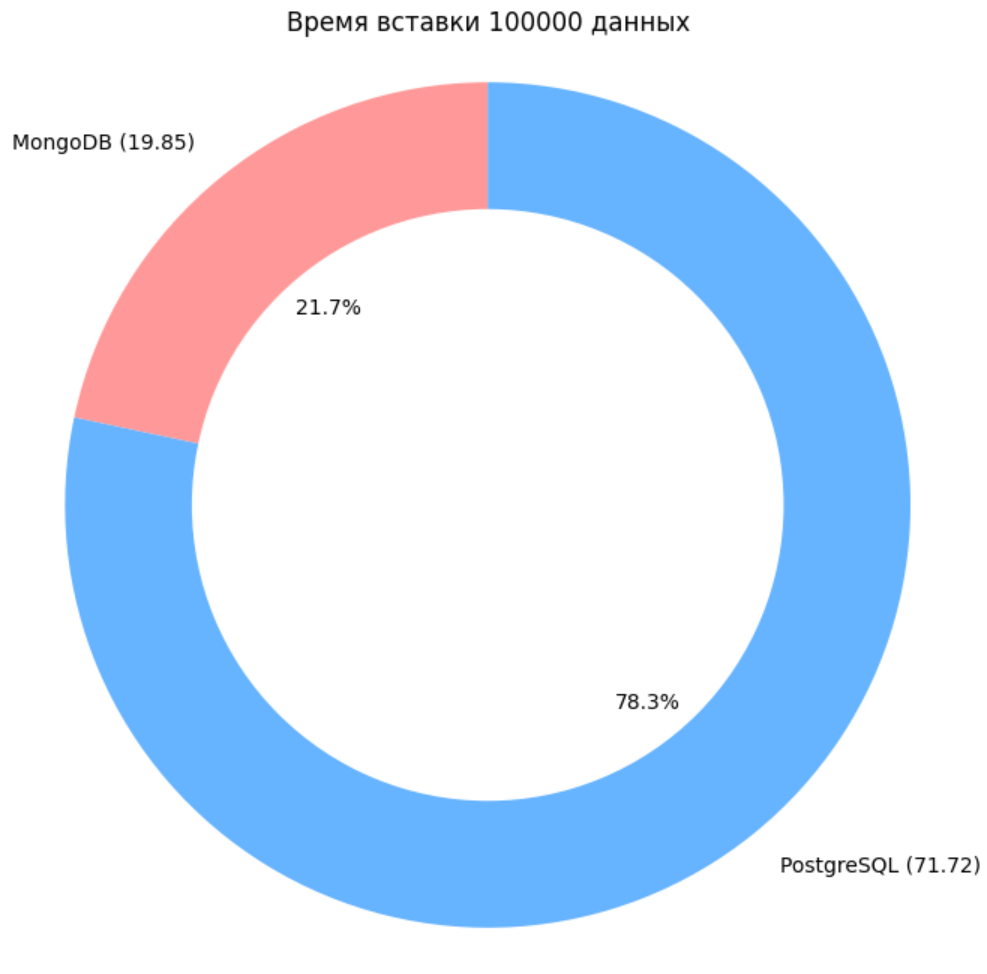


Рисунок 14 – Диаграмма вставки 100000 записей

Делаем вывод, что вставка данных в MongoDB почти в 2,5 раза быстрее, чем в PostreSQL.

# ЗАКЛЮЧЕНИЕ

По результатам оценки скорости вставки большого количества записей в базу данных MongoDB и PostgreSQL первый лидирует.

MongoDB представляет собой NoSQL базу данных, которая использует гибкую структуру документов в формате BSON для хранения данных. Это позволяет снизить накладные расходы на преобразование данных и позволяет более эффективно вставлять большое количество записей. Кроме того, MongoDB поддерживает параллельное выполнение операций записи, что также способствует увеличению производительности.

PostgreSQL является реляционной базой данных, которая использует таблицы для хранения данных. Это означает, что PostgreSQL может иметь больше ограничений и проверок целостности данных, что может замедлить процесс вставки большого количества записей.

Таким образом, из-за своей гибкой структуры и поддержки параллельного выполнения операций вставки, MongoDB может быть быстрее вставлять большое количество записей по сравнению с PostgreSQL.