



## МИНИСТЕРСТВО НАУКИ И ВЫСШЕГО ОБРАЗОВАНИЯ РОССИЙСКОЙ ФЕДЕРАЦИИ

Федеральное государственное бюджетное образовательное учреждение высшего образования

**«МИРЭА – Российский технологический университет»**

### **РТУ МИРЭА**

**Институт комплексной безопасности и цифровых технологий (ИКБ) Кафедра КБ-14**  
**«Цифровые технологии обработки данных»**

**Направление подготовки**

**09.03.02 Информационные системы и технологии**

**Практическая работа №4**

#### **Задание 1:**

1. Создать новую базу данных с именем wal\_db.
2. Создать таблицу orders с колонками:
  - a. id: уникальный идентификатор заказа (целое число, первичный ключ, автоинкремент)
  - b. customer\_name: имя клиента (varchar(255))
  - c. order\_date: дата размещения заказа (date)
  - d. total\_amount: общая сумма заказа (numeric)
3. Включить WAL в postgresql.conf и перезапустить сервер PostgreSQL.
4. Вставить в таблицу orders несколько примеров данных.
5. Узнать сколько байт занимают сгенерированные журнальные записи (**pg\_current\_wal\_insert\_lsn()**)
6. Изменить некоторые из существующих записей в таблице orders.
7. Удалить несколько записей из таблицы orders.
8. Проверить содержимое файлов журнала WAL, чтобы увидеть записи операций, выполненных в шагах 4-6 (Для проверки содержимого файлов журнала WAL можно воспользоваться утилитой pg\_waldump).
9. Настроить параметры, связанные с WAL, такие как max\_wal\_size и checkpoint\_timeout в postgresql.conf.

10. Перезапустите сервер PostgreSQL, чтобы применить изменения в конфигурации.
11. Повторите шаги 4-7, чтобы наблюдать изменения в файлах WAL из-за новых параметров конфигурации.

## **Задание 2.**

1. Создайте новую базу данных `benchmark` в PostgreSQL, которая будет использоваться для тестирования производительности с помощью `pgbench`.
2. Настройте базу данных `benchmark` используя утилиту `pgbench` с параметрами `-i` и `-s 50` (объясните что делают эти параметры).
3. Запустите команду `pgbench` на созданной таблице с разными параметрами, такими как количество клиентов, количество транзакций и длительность тестирования.
4. Проанализируйте результаты тестирования, смотрите на показатели, такие как количество транзакций в секунду, время выполнения транзакций и общее время работы.
5. Попробуйте изменить параметры `pgbench` и повторить тестирование, чтобы понять, как они влияют на производительность базы данных. Используйте различные значения для параметров, а также изменяйте конфигурационные параметры PostgreSQL, такие как `shared_buffers` и `work_mem`, чтобы увидеть, как это влияет на производительность.

## **Задание 3.**

1. Для базы данных из первого задания настройте выполнение контрольной точки раз в 30 секунд. Установите параметры `min_wal_size` и `max_wal_size` в 16 МБ.
2. Несколько минут с помощью утилиты `pgbench` подавайте нагрузку 100 транзакций/сек.
3. Измерьте, какой объем журнальных файлов был сгенерирован за это время. Оцените, какой объем приходится в среднем на одну контрольную точку.
4. Проверьте данные статистики: все ли контрольные точки выполнялись по расписанию? Как можно объяснить полученный результат?
5. Сбросьте настройки к значениям по умолчанию.