

**Министерство науки и высшего образования Российской Федерации
ФЕДЕРАЛЬНОЕ ГОСУДАРСТВЕННОЕ АВТОНОМНОЕ ОБРАЗОВАТЕЛЬНОЕ
УЧРЕЖДЕНИЕ ВЫСШЕГО ОБРАЗОВАНИЯ
НАЦИОНАЛЬНЫЙ ИССЛЕДОВАТЕЛЬСКИЙ УНИВЕРСИТЕТ ИТМО**

Факультет безопасности информационных технологий

Направление подготовки: 11.03.03

Образовательная программа: Безопасность информационных технологий

Дисциплина:

«Информационная безопасность баз данных»

ОТЧЕТ ПО ЛАБОРАТОРНОЙ РАБОТЕ №5

« Резервирование БД и восстановление по контрольным точкам »

Выполнил:

Рядовой Т.С., студент группы N3352, поток ИББД.N63 1.5

_____ (подпись)

Проверил:

Салихов Максим Русланович

_____ (отметка о выполнении)

_____ (подпись)

_____ (дата)

Санкт-Петербург
2025 г.

СОДЕРЖАНИЕ

Введение.....	3
1 Ход работы.....	4
1.1 Создание резервной копии БД	4
1.2 Внесение случайных изменений в таблицы	6
1.3 Создание контрольной точки и откат изменений	6
1.3.1 Создание контрольной точки	6
1.3.2 Восстановление из резервной копии.....	6
1.4 Анализ откаченных изменений.....	6
Заключение.....	8
Список источников	9

ВВЕДЕНИЕ

Цель работы: получение навыков по резервированию и восстановлению базы данных.

1 ХОД РАБОТЫ

Задание:

1. Создание резервной копии БД согласно выбранному расписанию.
2. Внесение случайных изменений в таблицы базы данных до момента создания контрольной точки.
3. Демонстрация процесса отката к последней контрольной точке и откат изменений, выполненных в пункте 2.
4. Анализ возможности просмотра изменений, которые были откачены, с помощью системы логирования СУБД или средств системы резервирования.

1.1 Создание резервной копии БД

Для создания резервной копии базы данных используется утилита `pg_dump`, которая входит в состав PostgreSQL.

Листинг 1 – Команда для создания резервной копии:

```
pg_dump -U postgres -F c -b -v -f backup.dump repair_workshop
```

Описание параметров:

- `-U postgres` — указание пользователя для подключения к БД.
- `-F c` — формат резервной копии (custom).
- `-b` — включение больших объектов в резервную копию.
- `-v` — вывод подробной информации о процессе.
- `-f backup.dump` — указание имени файла для резервной копии.
- `repair_workshop` — имя базы данных, для которой создается резервная копия.

```

➔ pg_dump -U postgres -F c -b -v -f backup.dump repair_workshop
pg_dump: последний системный OID: 16383
pg_dump: чтение расширений
pg_dump: выявление членов расширений
pg_dump: чтение схем
pg_dump: чтение пользовательских таблиц
pg_dump: чтение пользовательских функций
pg_dump: чтение пользовательских типов
pg_dump: чтение процедурных языков
pg_dump: чтение пользовательских агрегатных функций
pg_dump: чтение пользовательских операторов
pg_dump: чтение пользовательских методов доступа
pg_dump: чтение пользовательских классов операторов
pg_dump: чтение пользовательских семейств операторов
pg_dump: чтение пользовательских анализаторов текстового поиска
pg_dump: чтение пользовательских шаблонов текстового поиска
pg_dump: чтение пользовательских словарей текстового поиска
pg_dump: чтение пользовательских конфигураций текстового поиска
pg_dump: чтение пользовательских оболочек сторонних данных
pg_dump: чтение пользовательских сторонних серверов
pg_dump: чтение прав по умолчанию
pg_dump: чтение пользовательских правил сортировки
pg_dump: чтение пользовательских преобразований
pg_dump: чтение приведений типов
pg_dump: чтение преобразований
pg_dump: чтение информации о наследовании таблиц
pg_dump: чтение событийных триггеров
pg_dump: поиск таблиц расширений
pg_dump: поиск связей наследования
pg_dump: чтение информации о столбцах интересующих таблиц
pg_dump: поиск выражений по умолчанию для таблиц
pg_dump: поиск ограничений-проверок для таблиц
pg_dump: пометка наследованных столбцов в подтаблицах
pg_dump: чтение информации о секционировании
pg_dump: чтение индексов
pg_dump: пометка индексов в секционированных таблицах
pg_dump: чтение расширенной статистики
pg_dump: чтение ограничений
pg_dump: чтение триггеров
pg_dump: чтение правил перезаписи
pg_dump: чтение политик
pg_dump: чтение политик защиты на уровне строк
pg_dump: чтение публикаций
pg_dump: чтение информации о таблицах, включённых в публикации
pg_dump: чтение информации о схемах, включённых в публикации
pg_dump: чтение подписок
pg_dump: чтение больших объектов
pg_dump: чтение информации о зависимостях
pg_dump: сохранение кодировки (UTF8)
pg_dump: сохранение standard_conforming_strings (on)
pg_dump: сохранение search_path =
pg_dump: сохранение определения базы данных
pg_dump: выгрузка содержимого таблицы "public.clients"
pg_dump: выгрузка содержимого таблицы "public.employees"
pg_dump: выгрузка содержимого таблицы "public.main_log"
pg_dump: выгрузка содержимого таблицы "public.materials"
pg_dump: выгрузка содержимого таблицы "public.order_materials"
pg_dump: выгрузка содержимого таблицы "public.order_services"
pg_dump: выгрузка содержимого таблицы "public.orders"
pg_dump: выгрузка содержимого таблицы "public.payments"
pg_dump: выгрузка содержимого таблицы "public.reports"
pg_dump: выгрузка содержимого таблицы "public.secret_data"
pg_dump: выгрузка содержимого таблицы "public.services"

```

Рисунок 1 – Реализация команды pg_dump

1.2 Внесение случайных изменений в таблицы

Перед созданием контрольной точки внесем случайные изменения в таблицы базы данных.

Листинг 2 – Пример изменения данных в таблице "Заказы"

```
UPDATE orders SET status = 'Отменен' WHERE order_id = 1;
```

Листинг 3 – Пример удаления данных из таблицы "Клиенты"

```
DELETE FROM clients WHERE client_id = 2;
```

1.3 Создание контрольной точки и откат изменений

1.3.1 Создание контрольной точки

В PostgreSQL контрольная точка создается автоматически, но можно принудительно вызвать создание контрольной точки с помощью команды CHECKPOINT.

1.3.2 Восстановление из резервной копии

Для восстановления базы данных из резервной копии используется утилита pg_restore.

Листинг 4 – Команда для восстановления

```
pg_restore -U postgres -d repair_workshop -v backup.dump
```

Описание параметров:

- -U postgres — указание пользователя для подключения к БД;
- -d repair_workshop — имя базы данных, в которую восстанавливаются данные;
- -v — вывод подробной информации о процессе;
- backup.dump — имя файла резервной копии.

После восстановления базы данных изменения, внесенные в пункте 2, будут откачены.

1.4 Анализ откаченных изменений

Для анализа изменений, которые были откачены, можно использовать систему логирования, созданную в лабораторной работе №3.

Листинг 5 – Пример запроса к таблице-логу

```
SELECT * FROM main_log WHERE operation_type IN ('UPDATE', 'DELETE');
```

Результат запроса:

- В таблице-логе будут зафиксированы все изменения, которые были внесены в базу данных до создания контрольной точки;
- Это позволяет отследить, какие именно данные были изменены или удалены, и восстановить их вручную, если это необходимо.

ЗАКЛЮЧЕНИЕ

В ходе выполнения лабораторной работы были получены навыки по резервированию и восстановлению базы данных. Была создана резервная копия БД, внесены случайные изменения, продемонстрирован процесс отката к последней контрольной точке и проведен анализ откаченных изменений с помощью системы логирования.

СПИСОК ИСТОЧНИКОВ

1. Документация PostgreSQL [Электронный ресурс]. – URL:
<https://www.postgresql.org/docs/current/app-pgdump.html> (Дата обращения: 25.01.2025).
2. Резервное копирование в PostgreSQL [Электронный ресурс]. – URL:
<https://selectel.ru/blog/postgresql-backup-tools/> (Дата обращения: 25.01.2025).