

# Липецкий государственный технический университет

Факультет автоматизации и информатики  
Кафедра автоматизированных систем управления

Отчет по лабораторной работе № 7  
по дисциплине «OS Linux»  
на тему «Создание дампа БД и восстановление»

Студент

Группа АС-18-1

Руководитель

К.Н.

учёная степень, учёное звание

подпись, дата

подпись, дата

Сухоруков К.О.

фамилия, инициалы

Кургасов В.В.

фамилия, инициалы

Липецк 2020 г.

## СОДЕРЖАНИЕ

Цель работы . . . . .	2
1 Ход работы . . . . .	3
1.1 Создание дампа . . . . .	3
1.2 Восстановление БД из дампа . . . . .	6
Вывод . . . . .	8
Контрольные вопросы . . . . .	9

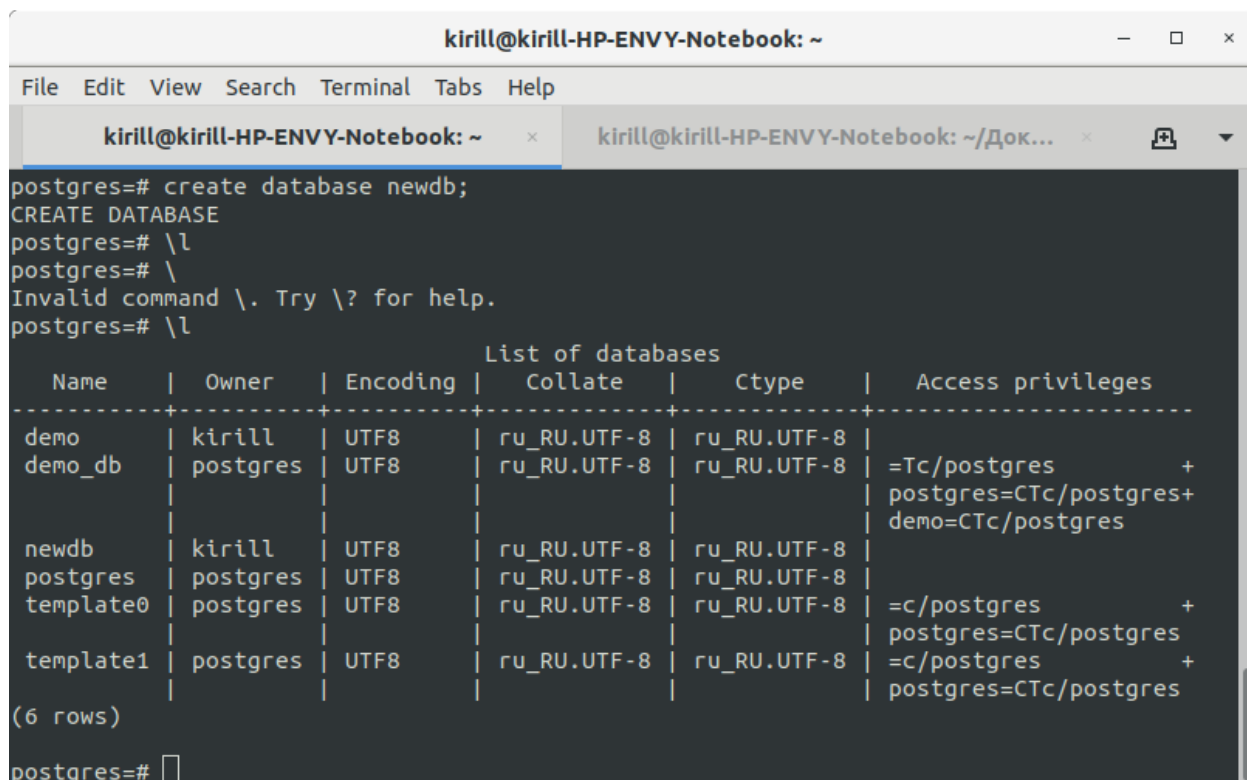
Цель работы

Получить основные навыки в создании БД, а также восстановление из дампа БД.

## 1 Ход работы

### 1.1 Создание дампа

Создадим тестовую базу данных с помощью команды `create <имя БД>`. Затем проследим, что БД успешно создавалась с помощью команды `\d`. Результат выполнения команды представлен на рисунке 1.1.



```
kirill@kirill-HP-ENVY-Notebook: ~
File Edit View Search Terminal Tabs Help
kirill@kirill-HP-ENVY-Notebook: ~
kirill@kirill-HP-ENVY-Notebook: ~/Док...

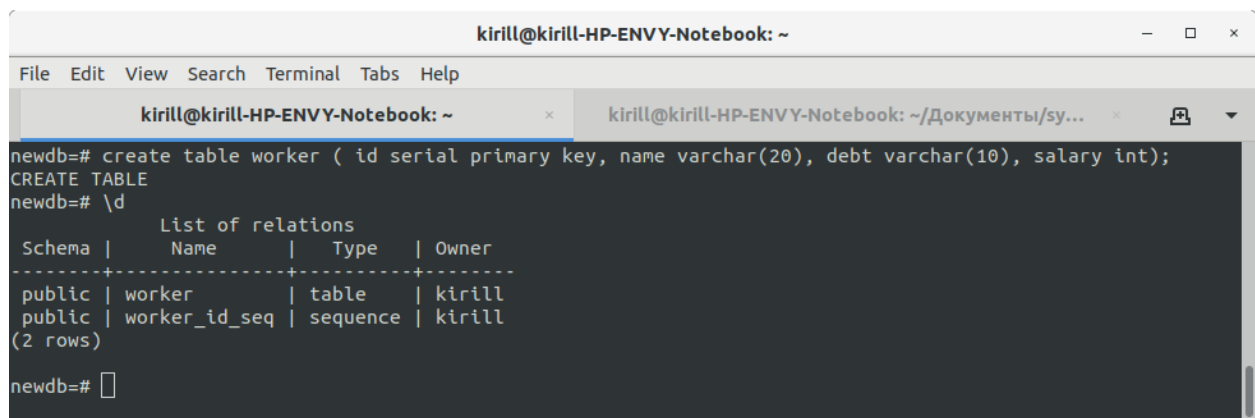
postgres=# create database newdb;
CREATE DATABASE
postgres=# \l
postgres=# \
Invalid command \. Try \? for help.
postgres=# \l

      List of databases
  Name  | Owner  | Encoding | Collate  | Ctype  | Access privileges
-----+-----+-----+-----+-----+-----
demo    | kirill | UTF8     | ru_RU.UTF-8 | ru_RU.UTF-8 |
demo_db | postgres | UTF8     | ru_RU.UTF-8 | ru_RU.UTF-8 | =Tc/postgres +
        |        |          |              |              | postgres=Ctc/postgres+
        |        |          |              |              | demo=Ctc/postgres
newdb   | kirill | UTF8     | ru_RU.UTF-8 | ru_RU.UTF-8 |
postgres | postgres | UTF8     | ru_RU.UTF-8 | ru_RU.UTF-8 |
template0 | postgres | UTF8     | ru_RU.UTF-8 | ru_RU.UTF-8 | =c/postgres +
        |        |          |              |              | postgres=Ctc/postgres
template1 | postgres | UTF8     | ru_RU.UTF-8 | ru_RU.UTF-8 | =c/postgres +
        |        |          |              |              | postgres=Ctc/postgres
(6 rows)

postgres=#
```

Рисунок 1.1 – Создание тестовой базы данных

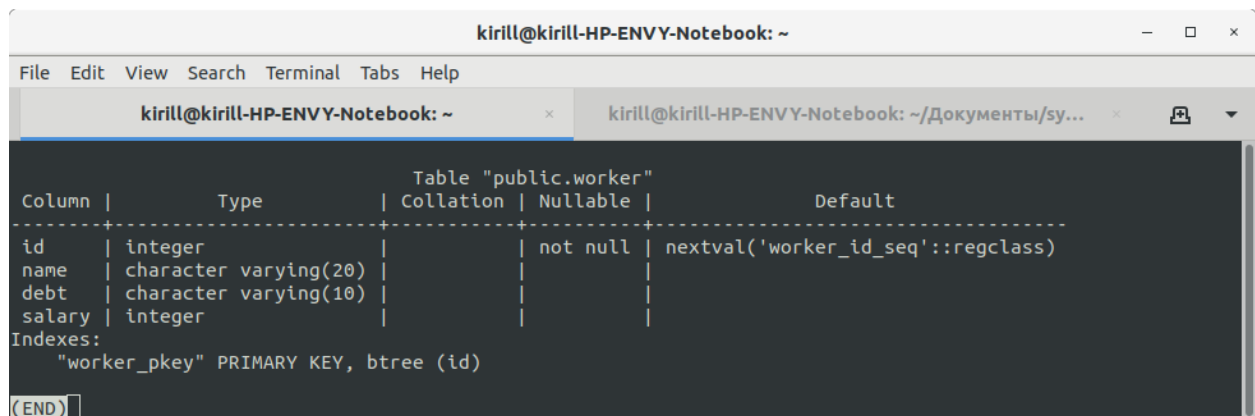
Создадим в новой базе данных таблицу с помощью команды `create table <имя таблицы> (<поля таблицы с указанием типа и имени>)`. Проверим созданную таблицу с помощью команды `\d`. Результат создания таблицы представлен на рисунке 1.2.



```
kirill@kirill-HP-ENVY-Notebook: ~  
File Edit View Search Terminal Tabs Help  
kirill@kirill-HP-ENVY-Notebook: ~ x kirill@kirill-HP-ENVY-Notebook: ~/Документы/sy... x  
newdb=# create table worker ( id serial primary key, name varchar(20), debt varchar(10), salary int);  
CREATE TABLE  
newdb=# \d  
List of relations  
Schema | Name | Type | Owner  
-----+-----+-----+-----  
public | worker | table | kirill  
public | worker_id_seq | sequence | kirill  
(2 rows)  
newdb=#
```

Рисунок 1.2 – Создание таблицы в БД

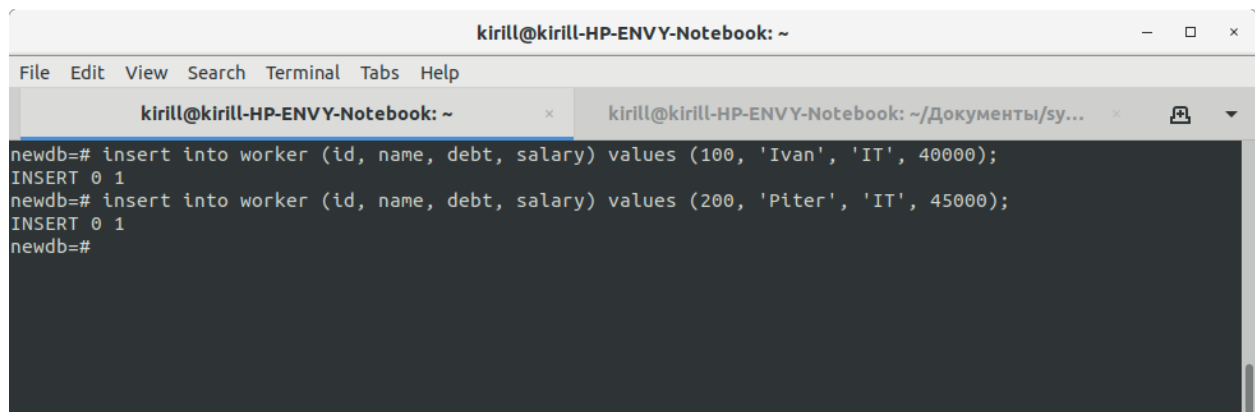
С помощью команды `\d <имя таблицы>` проверим структуру таблицы.



```
kirill@kirill-HP-ENVY-Notebook: ~  
File Edit View Search Terminal Tabs Help  
kirill@kirill-HP-ENVY-Notebook: ~ x kirill@kirill-HP-ENVY-Notebook: ~/Документы/sy... x  
Table "public.worker"  
Column | Type | Collation | Nullable | Default  
-----+-----+-----+-----+-----  
id | integer | | not null | nextval('worker_id_seq'::regclass)  
name | character varying(20) | | |  
debt | character varying(10) | | |  
salary | integer | | |  
Indexes:  
"worker_pkey" PRIMARY KEY, btree (id)  
(END)
```

Рисунок 1.3 – Структура таблицы worker

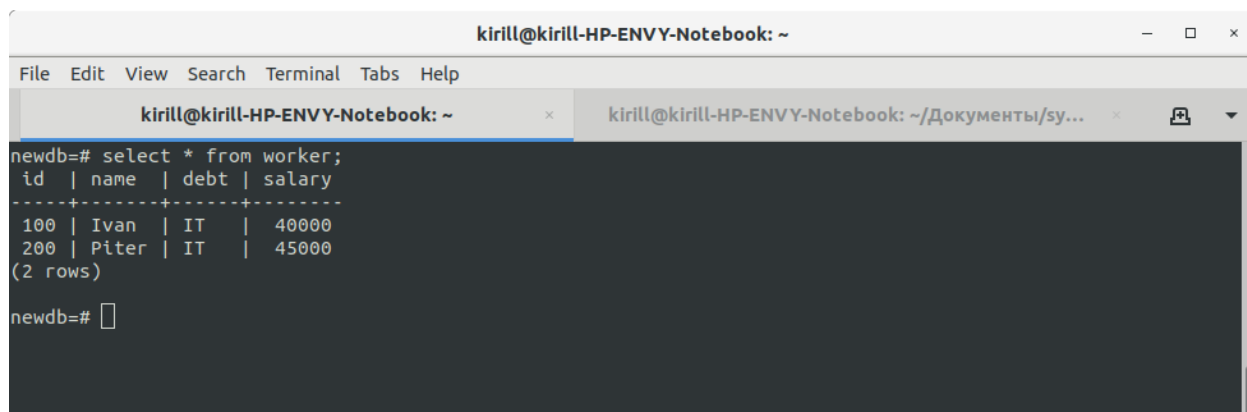
Добавим 2 записи в созданную таблицу с помощью команды `insert into <имя таблицы>(<поле1>, <поле2>, ...) values (<значение1>, <значение2>, ...)`. Результат выполнения данных команд представлен на рисунке 1.4.



```
kirill@kirill-HP-ENVY-Notebook: ~  
File Edit View Search Terminal Tabs Help  
kirill@kirill-HP-ENVY-Notebook: ~ x kirill@kirill-HP-ENVY-Notebook: ~/Документы/sy... x  
newdb=# insert into worker (id, name, debt, salary) values (100, 'Ivan', 'IT', 40000);  
INSERT 0 1  
newdb=# insert into worker (id, name, debt, salary) values (200, 'Piter', 'IT', 45000);  
INSERT 0 1  
newdb=#
```

Рисунок 1.4 – Добавление записей в таблицу

Проверим, что записи успешно добавлены с помощью команды `select * from <имя таблицы>`.

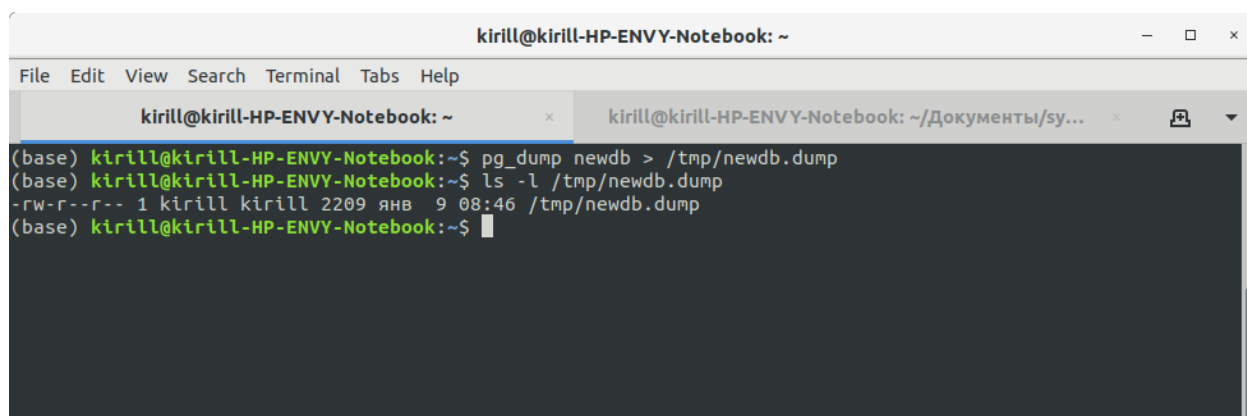


The screenshot shows a terminal window titled "kirill@kirill-HP-ENVY-Notebook: ~". The terminal displays the output of a SQL query executed in a database named "newdb". The query is `select * from worker;`. The result is a table with four columns: `id`, `name`, `debt`, and `salary`. There are two rows of data: one for Ivan with a salary of 40000, and one for Piter with a salary of 45000. The terminal prompt is `newdb=#`.

```
kirill@kirill-HP-ENVY-Notebook: ~  
File Edit View Search Terminal Tabs Help  
kirill@kirill-HP-ENVY-Notebook: ~ x kirill@kirill-HP-ENVY-Notebook: ~/Документы/sy... x  
newdb=# select * from worker;  
id | name | debt | salary  
-----+-----+-----+-----  
100 | Ivan | IT    | 40000  
200 | Piter | IT    | 45000  
(2 rows)  
newdb=#
```

Рисунок 1.5 – Проверка содержимого таблицы worker

Создадим дамп базы данных с помощью команды `pg_dump`. Результат выполнения команды представлен на рисунке 1.6.



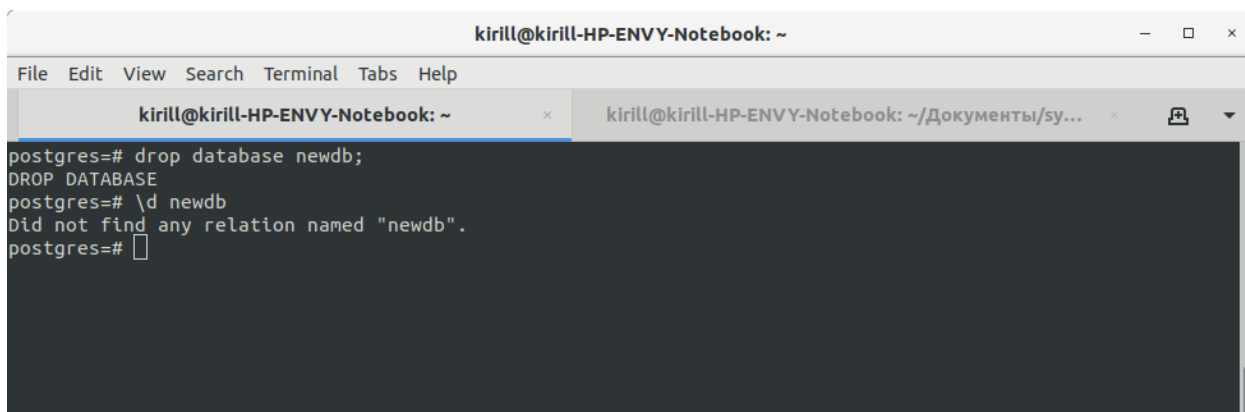
The screenshot shows a terminal window titled "kirill@kirill-HP-ENVY-Notebook: ~". The terminal displays the execution of two commands. First, the command `pg_dump newdb > /tmp/newdb.dump` is executed, creating a dump file. Then, the command `ls -l /tmp/newdb.dump` is executed, showing the file's permissions, owner, size, and creation date. The terminal prompt is `(base) kirill@kirill-HP-ENVY-Notebook:~$`.

```
kirill@kirill-HP-ENVY-Notebook: ~  
File Edit View Search Terminal Tabs Help  
kirill@kirill-HP-ENVY-Notebook: ~ x kirill@kirill-HP-ENVY-Notebook: ~/Документы/sy... x  
(base) kirill@kirill-HP-ENVY-Notebook:~$ pg_dump newdb > /tmp/newdb.dump  
(base) kirill@kirill-HP-ENVY-Notebook:~$ ls -l /tmp/newdb.dump  
-rw-r--r-- 1 kirill kirill 2209 янв  9 08:46 /tmp/newdb.dump  
(base) kirill@kirill-HP-ENVY-Notebook:~$
```

Рисунок 1.6 – Создание дампа БД

## 1.2 Восстановление БД из дампа

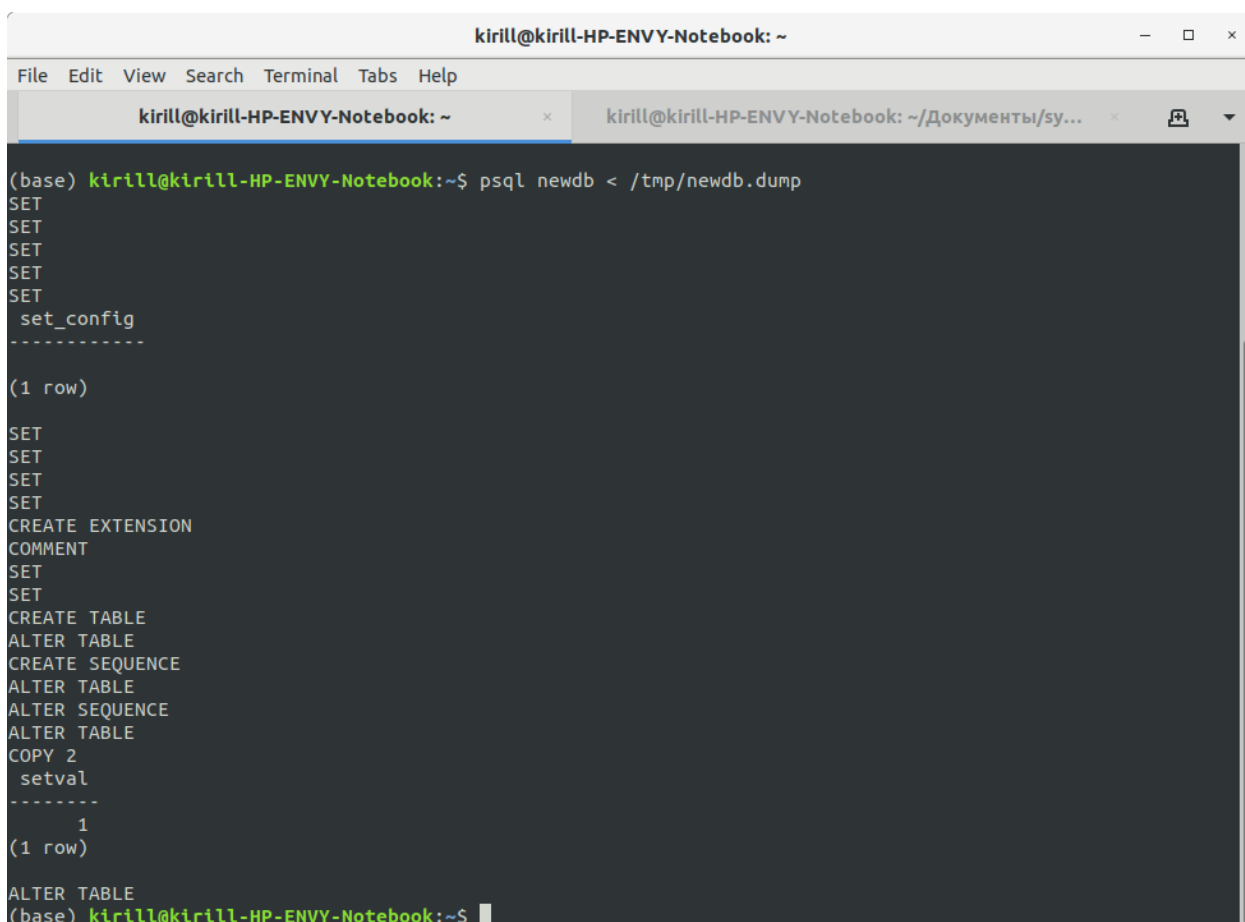
Удалим базу данных newdb с помощью команды dropdb. Результат выполнения команды представлен на рисунке 1.7.



```
kirill@kirill-HP-ENVY-Notebook: ~  
File Edit View Search Terminal Tabs Help  
kirill@kirill-HP-ENVY-Notebook: ~ x kirill@kirill-HP-ENVY-Notebook: ~/Документы/sy... x  
postgres=# drop database newdb;  
DROP DATABASE  
postgres=# \d newdb  
Did not find any relation named "newdb".  
postgres=#
```

Рисунок 1.7 – Удаление БД

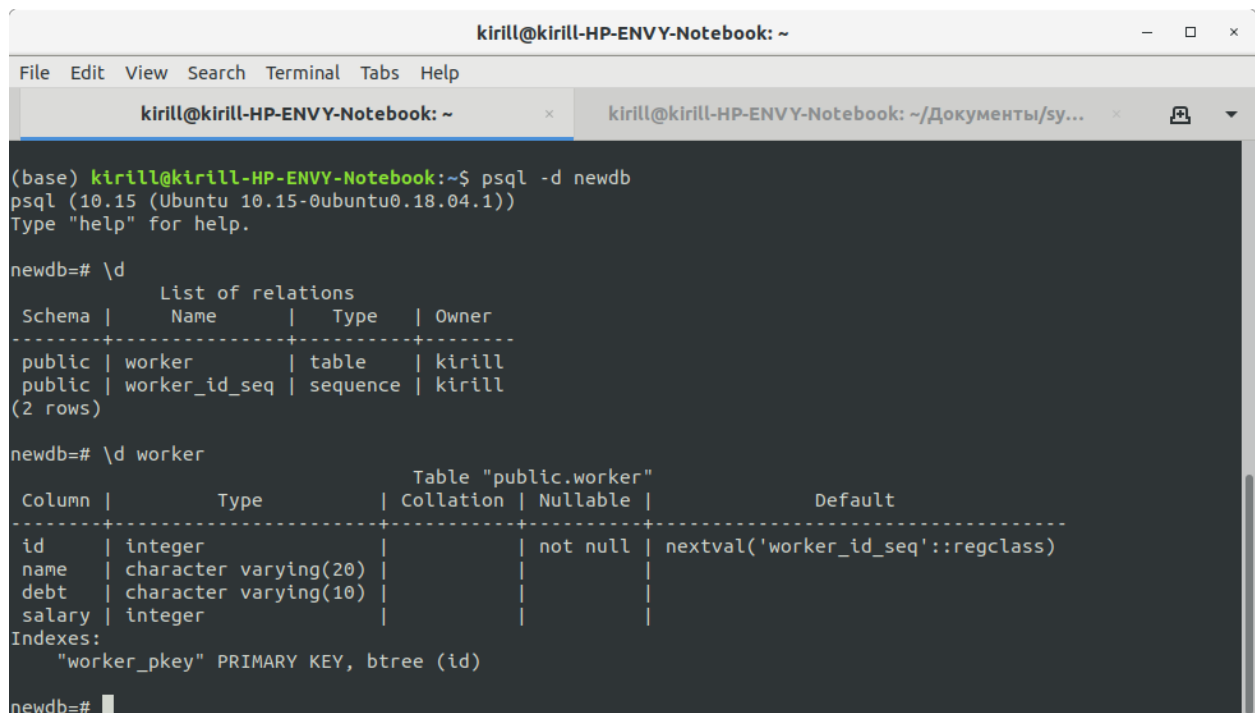
Заново создадим БД и затем восстановим ее из дампа с помощью команды psql <имя БД> < <путь до дампа>. Результат выполнения восстановления представлен на рисунке 1.8.



```
kirill@kirill-HP-ENVY-Notebook: ~  
File Edit View Search Terminal Tabs Help  
kirill@kirill-HP-ENVY-Notebook: ~ x kirill@kirill-HP-ENVY-Notebook: ~/Документы/sy... x  
(base) kirill@kirill-HP-ENVY-Notebook:~$ psql newdb < /tmp/newdb.dump  
SET  
SET  
SET  
SET  
SET  
set_config  
-----  
(1 row)  
  
SET  
SET  
SET  
SET  
CREATE EXTENSION  
COMMENT  
SET  
SET  
CREATE TABLE  
ALTER TABLE  
CREATE SEQUENCE  
ALTER TABLE  
ALTER SEQUENCE  
ALTER TABLE  
COPY 2  
setval  
-----  
1  
(1 row)  
  
ALTER TABLE  
(base) kirill@kirill-HP-ENVY-Notebook:~$
```

Рисунок 1.8 – Восстановление БД из дампа

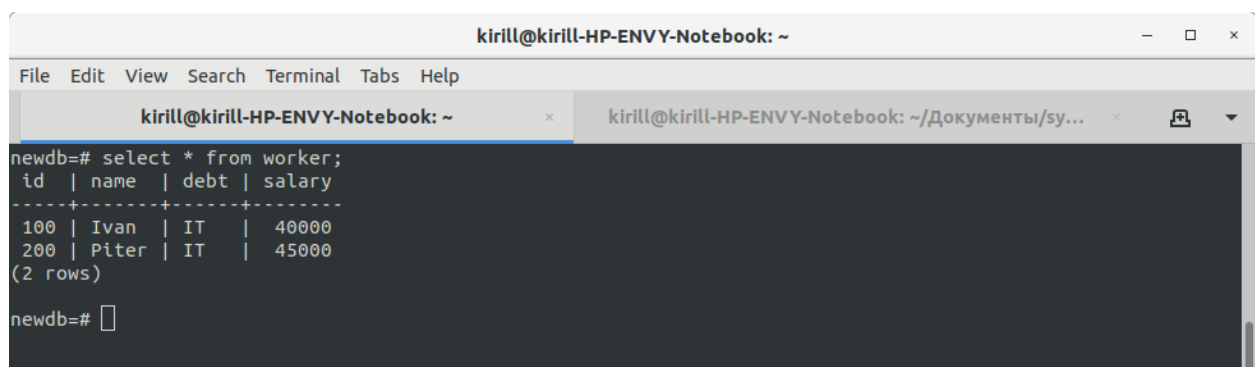
Проверим структуру восстановленной БД. Результат проверки представлен на рисунке 1.9.



```
kirill@kirill-HP-ENVY-Notebook: ~  
File Edit View Search Terminal Tabs Help  
kirill@kirill-HP-ENVY-Notebook: ~ x kirill@kirill-HP-ENVY-Notebook: ~/Документы/sy... x  
(base) kirill@kirill-HP-ENVY-Notebook:~$ psql -d newdb  
psql (10.15 (Ubuntu 10.15-0ubuntu0.18.04.1))  
Type "help" for help.  
  
newdb=# \d  
List of relations  
Schema | Name | Type | Owner  
-----+-----+-----+-----  
public | worker | table | kirill  
public | worker_id_seq | sequence | kirill  
(2 rows)  
  
newdb=# \d worker  
Table "public.worker"  
Column | Type | Collation | Nullable | Default  
-----+-----+-----+-----+-----  
id | integer | | not null | nextval('worker_id_seq'::regclass)  
name | character varying(20) | | |  
debt | character varying(10) | | |  
salary | integer | | |  
Indexes:  
"worker_pkey" PRIMARY KEY, btree (id)  
newdb=#
```

Рисунок 1.9 – Проверка структуры БД

Проверим содержимое таблицы worker. Результат проверки содержания представлен на рисунке 1.10.



```
kirill@kirill-HP-ENVY-Notebook: ~  
File Edit View Search Terminal Tabs Help  
kirill@kirill-HP-ENVY-Notebook: ~ x kirill@kirill-HP-ENVY-Notebook: ~/Документы/sy... x  
newdb=# select * from worker;  
id | name | debt | salary  
-----+-----+-----+-----  
100 | Ivan | IT | 40000  
200 | Piter | IT | 45000  
(2 rows)  
newdb=#
```

Рисунок 1.10 – Проверка содержимого таблицы worker



## Вывод

В ходе выполнения данной лабораторной работы были получены основные навыки создания БД, а также восстановление из дампа БД.

## Контрольные вопросы

1. Что такое ключ ssh? В чем преимущество их использования?

SSH-ключи используются для идентификации клиента при подключении к удалённому серверу. SSH-ключи представляют собой пару ключей – приватный и публичный. Приватный ключ хранится в закрытом доступе у клиента, публичный отправляется на сервер. Преимущество использования ключей в удобстве (не нужно запоминать пароли) и безопасности (взломать приватный ssh-ключ достаточно сложно).

2. Как сгенерировать ключи ssh в разных ОС?

Генерация ssh-ключа в ОС Linux возможна с помощью команды `sshkeygen`. В ОС Windows можно использовать программу PuTTY для генерации ssh-ключей и подключения по ssh-протоколу.

3. Возможно ли из «секретного» ключа сгенерировать «публичный» и/или наоборот?

Нет, невозможно.

4. Будут ли отличаться пары ключей, сгенерированные на одном ПК несколько раз с исходными условиями (наличие/отсутствие пароля на «секретный» ключ и т.п.)

Да, будут. Утилита `ssh-keygen` каждый раз случайно генерирует пару ключей.

5. Перечислите доступные ключи для `ssh-keygen.exe`

- DSA;
- RSA;
- ECDASA;
- Ed25519.

6. Можно ли использовать один «секретный» ключ доступа с разных ОС, установленных на одном ПК/на разных ПК?

Можно, но безопасность такого ключа уже не гарантирована.

7. Возможно ли организовать подключение «по ключу» ssh к системе с ОС Windows, в которой запущен OpenSSH сервер?

Да, возможно, с использованием программы PuTTY.

8. Какие известные Вам сервисы сети Интернет позволяют организовать доступ к ресурсам посредством SSH ключей?

GitHub