

**Липецкий государственный технический университет**

Факультет автоматизации и информатики

Кафедра Автоматизированных систем управления

**ЛАБОРАТОРНАЯ РАБОТА №8**

По дисциплине «ОС Linux»

Создание дампа БД и восстановление

Студент

Печенкин Д.В.

Группа ПИ-18

Руководитель

Доцент

Кургасов В.В.

Липецк 2021г

## Цель работы

Лабораторная работа предназначена для целей практического ознакомления с созданием дампа БД и восстановлением БД.

## Ход работы

После установки postgres, командой sudo -i -u postgres переходим на пользователя postgres

```
danirogue@daniel-vb:~$ sudo -i -u postgres  
postgres@daniel-vb:~$ █
```

Рисунок 1 – Переход на пользователя postgres

После перехода на пользователя, вводим команду psql для перехода в интерактивный режим и командной CREATE DATABASE lab8\_db; создадим базу данных (командой \l проверим её наличие).

```
postgres@daniel-vb:~$ psql  
psql (12.5 (Ubuntu 12.5-0ubuntu0.20.04.1))  
Type "help" for help.  
  
postgres=# CREATE DATABASE jacob_db;  
CREATE DATABASE  
postgres=# █
```

Рисунок 2 – Создание базы данных

List of databases						
Name	Owner	Encoding	Collate	Ctype	Access privileges	
jacob_db	postgres	UTF8	en_US.UTF-8	en_US.UTF-8		
postgres	postgres	UTF8	en_US.UTF-8	en_US.UTF-8		
template0	postgres	UTF8	en_US.UTF-8	en_US.UTF-8	=c/postgres	+
					postgres=CTc/postgres	
template1	postgres	UTF8	en_US.UTF-8	en_US.UTF-8	=c/postgres	+
					postgres=CTc/postgres	

Рисунок 3 – Проверка наличия созданной базы данных

Подключимся к созданной базе данных с помощью команды \connect lab8\_db и проверим количество существующих таблиц в ней командой \db.

```
postgres=# \connect jacob_db
You are now connected to database "jacob_db" as user "postgres".
jacob_db=# \dp
          Access privileges
 Schema | Name | Type | Access privileges | Column privileges | Policies
-----+-----+-----+-----+-----+-----+
(0 rows)

jacob_db=#
```

Рисунок 4 – Подключение к базе данных

Создадим новую таблицу запросом:

```
CREATE SEQUENCE user_id;
CREATE TABLE users
(id INTEGER PRIMARY KEY DEFAULT NEXTVAL ('user_id'),
login CHAR(16),
password VARCHAR(80));
```

```
jacob_db=# CREATE TABLE users ( id INTEGER PRIMARY KEY DEFAULT NEXTVAL('user_id'),login CHAR(16),password VARCHAR(80));
CREATE TABLE
jacob_db=# \dp
          Access privileges
 Schema | Name   | Type   | Access privileges | Column privileges | Policies
-----+-----+-----+-----+-----+-----+
 public | user_id | sequence |           |           |           |
 public | users   | table   |           |           |           |
(2 rows)
```

Рисунок 5 – Создание таблицы

Добавим новую строку в таблицу users запросом: INSERT INTO users(login,password) VALUES('jacob','12345');

```
jacob_db=# INSERT INTO users(login,password) VALUES('jacob', '12345');
INSERT 0 1
jacob_db#
```

Рисунок 6 – Добавление данных в таблицу

Проверим наличие созданных данных запросом SELECT \* FROM users;

```
jacob_db=# SELECT * FROM users;
 id |      login      | password
----+-----+-----+
   1 | jacob          | 12345
(1 row)
```

Рисунок 7 – Просмотр данных в таблице

Для создания дампа базы данных, пропишем команду: pg\_dump lab8\_db > /tmp/lab8.dump. Произойдет создание дампа по пути tmp/lab8.dump.

```
postgres@daniel-vb:~$ pg_dump jacob_db >/tmp/lab8.dump
postgres@daniel-vb:~$
```

Рисунок 8 – Создание дампа

Перейдем в папку tmp для проверки наличия дампа.

```
danibrogue@daniel-vb:~$ cd /tmp
danibrogue@daniel-vb:/tmp$ ls
config.err-DMMBJt
lab8.dump
ssh-aQjaDAzZ065
systemd-private-6538681356974da6ad4db88df6e25fea-apache2.service-9ww5ig
systemd-private-6538681356974da6ad4db88df6e25fea-colorerd.service-Td7muj
systemd-private-6538681356974da6ad4db88df6e25fea-fwupd.service-410d5f
systemd-private-6538681356974da6ad4db88df6e25fea-ModemManager.service-tMV06f
systemd-private-6538681356974da6ad4db88df6e25fea-switcheroo-control.service-rRcpCh
danibrogue@daniel-vb:/tmp$
```

Рисунок 9 – Проверка наличия дампа

Удалим базу данных командой DROP DATABASE lab8\_db; и проверим её отсутствие.

```
postgres=# DROP DATABASE jacob_db;
DROP DATABASE
```

Рисунок 10 – Удаление базы данных

List of databases						
Name	Owner	Encoding	Collate	Ctype	Access	privileges
postgres	postgres	UTF8	en_US.UTF-8	en_US.UTF-8	=c/postgres	+
template0	postgres	UTF8	en_US.UTF-8	en_US.UTF-8	postgres=CTc/postgres	
template1	postgres	UTF8	en_US.UTF-8	en_US.UTF-8	=c/postgres	+
					postgres=CTc/postgres	

(3 rows)

Рисунок 11 – Проверка на существование базы данных

Восстановим удаленную базу данных через сохраненный дамп командой:

psql lab8\_db\_recovery <tmp/lab8.dump (базу данных lab8\_db\_recovery сначала необходимо создать).

```
postgres@daniel-vb:~$ createdb -T template0 jacob_db_recovery
postgres@daniel-vb:~$ psql jacob_db_recovery </tmp/lab8.dump
SET
SET
SET
SET
SET
set_config
-----
(1 row)

SET
SET
SET
SET
CREATE SEQUENCE
ALTER TABLE
SET
SET
CREATE TABLE
ALTER TABLE
COPY 1
  setval
-----
      1
(1 row)

ALTER TABLE
postgres@daniel-vb:~$ █
```

Рисунок 12 – Восстановление базы данных

Проверим восстановленную базу данных

```
postgres@daniel-vb:~$ psql
psql (12.5 (Ubuntu 12.5-0ubuntu0.20.04.1))
Type "help" for help.

postgres=# \c jacob_db_recovery
You are now connected to database "jacob_db_recovery" as user "postgres".
jacob_db_recovery=# \dp
          Access privileges
 Schema |   Name    | Type  | Access privileges | Column privileges | Policies
-----+-----+-----+-----+
 public | user_id | sequence |                   |                   |
 public | users   | table   |                   |                   |
(2 rows)

jacob_db_recovery=# SELECT * FROM users
jacob_db_recovery-# ;
 id |      login      | password
----+-----+-----+
  1 | jacob           | 12345
(1 row)

jacob_db_recovery=#
```

Рисунок 13 – Проверка восстановленной базы данных

## Вывод

В ходе выполнения лабораторной работы было изучено создание дампа базы данных и её восстановление.