Министерство науки и высшего образования Российской Федерации

Федеральное государственное бюджетное образовательное учреждение высшего образования

ИРКУТСКИЙ НАЦИОНАЛЬНЫЙ ИССЛЕДОВАТЕЛЬСКИЙ ТЕХНИЧЕСКИЙ УНИВЕРСИТЕТ

ОТЧЕТ

к лабораторной работе по дисциплине:

|  |
| --- |
| **Администрирование информационных систем** |
| Администрирование баз данных в UbuntuServer |

наименование темы

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| Выполнил студент группы: | ИСМб-19-1 |  |  |  | Вовиков Д.Е. |
|  |  | подпись |  | Фамилия И.О |
| Проверил: |  |  |  |  | Хритова М.А. |
|  | должность |  | подпись |  | Фамилия И.О |

Иркутск 2023 г.

Содержание

[Содержание 2](#_Toc128834911)

[1 Введение 3](#_Toc128834912)

[2 Основные понятия СУБД MySQL и PostgreSQL 4](#_Toc128834913)

[3 Установка СУБД 6](#_Toc128834914)

[3.1 Установка MySQL 6](#_Toc128834915)

[3.2 Установка PostgreSQL 7](#_Toc128834916)

[4 Настройка СУБД 8](#_Toc128834917)

[4.1 Настройка MySQL 8](#_Toc128834918)

[4.1.1 Конфигурирование файлов 8](#_Toc128834919)

[4.1.2 Создание базы данных 9](#_Toc128834920)

[4.1.3 Создание роли 11](#_Toc128834921)

[4.2 Настройка PostgreSQL 14](#_Toc128834922)

[4.2.1 Конфигурирование базы данных 14](#_Toc128834923)

[4.2.2 Создание базы данных 15](#_Toc128834924)

[4.2.3 Создание роли 16](#_Toc128834925)

[5 Заключение 18](#_Toc128834926)

1 Введение

**Цель работы:** получить начальные навыки практического администрирования базы данных в операционной системе UbuntuServer, изучить СУБД MySQL и PostgreSQL.

**Требования к отчетным документам:** отчетный документ представляется в печатном или электронном виде, в форматах pdf, doc(x).

Отчет должен содержать следующие главы:

1. Введение
2. Теоретическая часть. Основные понятия СУБД MySQL и PostgreSQL.
3. Практическая часть. Установка СУБД MySQL и PostgreSQL в ОС UbuntuServer (привести использованные команды и скриншоты).
4. Практическая часть. Настройка PostgreSQL. Работа с базой данных (привести использованные команды и скриншоты).
5. Заключение (полученные результаты).

Объем отчетного документа 7 - 10 страниц.

**Задание:**

1. Самостоятельно найти информацию и изучить СУБД MySQL и PostgreSQL.
2. Прочитать и изучить информацию об установке СУБД MySQL и PostgreSQL для операционной системы UbuntuServer.

3. Инсталлировать MySQL и PostgreSQL в UbuntuServer.

4. Выполнить настройку PostgreSQL в UbuntuServer (выполнить создание роли, базы данных, таблиц, произвести добавление и удаление данных).

5. Написать отчет.

2 Основные понятия СУБД MySQL и PostgreSQL

MySQL — самая популярная в мире база данных с открытым исходным кодом.

База данных MySQL — это клиент-серверная система, состоящая из многопоточного SQL-сервера, поддерживающего различные серверные части, нескольких различных клиентских программ и библиотек, инструментов администрирования и широкого спектра прикладных программных интерфейсов (API).

Основные понятия:

**GNU General Public License** —  лицензия на свободное программное обеспечение, созданная в рамках проекта GNU в 1988 г., по которой автор передаёт программное обеспечение в общественную собственность.

**Репликация** - механизм синхронизации содержимого нескольких копий объекта (например, содержимого базы данных).

**WAMP** - комплектация пакетов программ, под соответствующую ОС, обозначаемую первой буквой.

**AppServ** — пакет инструментов, который поможет использовать системы управления контентом.

**LAMP** — набор (комплекс) серверного программного обеспечения, широко используемый во Всемирной паутине.

**XAMPP** — кроссплатформенная сборка веб-сервера (развитие LAMP), содержащая Apache, MySQL, интерпретатор скриптов PHP, язык программирования Perl и большое количество дополнительных библиотек, позволяющих запустить полноценный веб-сервер.

**VertrigoServ** — профессиональный пакет программ, предназначенных для установки и управления полноценного веб-сервера Apache на домашнем компьютере под управлением Microsoft Windows.

**MyISAM** — систем хранения данных в СУБД MySQL.

**InnoDB** — систем хранения данных в СУБД MySQL.

PostgreSQL — это реляционная СУБД с открытым кодом и свободным распространением. База данных Postgre базируется на стандартном языке запросов SQL, а точнее, на его расширении — процедурном языке PL/pgSQL. PostgreSQL работает в системах Unix (FreeBSD и Linux: дистрибутивах Ubuntu, Debian, CentOS) и Windows.

Основные возможности:

1)**Функции** — являются блоками кода, исполняемыми на сервере, а не на клиенте БД.

2)**Триггеры** — определяются как функции, инициируемые DML-операциями. Например, операция INSERT может запускать триггер, проверяющий добавленную запись на соответствия определённым условиям. При написании функций для триггеров могут использоваться различные языки программирования.

3)**Индексы** — в PostgreSQL имеется поддержка индексов следующих типов: B-дерево, хэш и т.д.

4)**Многоверсионность** — PostgreSQL поддерживает одновременную модификацию БД несколькими пользователями с помощью механизма Multiversion Concurrency Control (MVCC). Благодаря этому соблюдаются требования ACID и практически отпадает нужда в блокировках чтения.

5)**Пользовательские объекты** — PostgreSQL может быть расширен пользователем для собственных нужд практически в любом аспекте. есть возможность добавлять собственные преобразования типов; типы данных; домены (пользовательские типы с изначально наложенными ограничениями); функции; индексы; операторы (включая переопределение уже существующих); процедурные языки и т.д.

Основные понятия:

**AIX** - UNIX-подобная операционная система компании IBM.

**BSD-система** — система распространения программного обеспечения в исходных кодах, созданная для обмена опытом между учебными заведениями.

**HP-UX** — проприетарная версия операционной системы UNIX фирмы HP.

**IRIX** — операционная система, используемая на рабочих станциях и серверах фирмы Silicon Graphics (SGI).

**DML** — это семейство компьютерных языков, используемых в компьютерных программах или пользователями баз данных для получения, вставки, удаления или изменения данных в базах данных.

**B-дерево** - структура данных, дерево поиска. С точки зрения внешнего логического представления, сбалансированное, сильно ветвистое дерево. Часто используется для хранения данных во внешней памяти.

**Хэш** — это структура данных, реализующая интерфейс ассоциативного массива, а именно, она позволяет хранить пары (ключ, значение) и выполнять три операции: операцию добавление новой пары, операцию поиска и операцию удаления пары по ключу.

**Multiversion** **Concurrency** **Control** — механизм обеспечения параллельного доступа к БД, заключающийся в предоставлении каждому пользователю так называемого «снимка» БД, обладающего тем свойством, что вносимые пользователем изменения в БД невидимы другим пользователям до момента фиксации транзакции.

**ACID** — описывает требования к транзакционной системе, обеспечивающие более надежную и предсказуемую ее работу.

3 Установка СУБД

3.1 Установка MySQL

Для установки MySQL применялась команда: sudo apt-get install mysql-server.

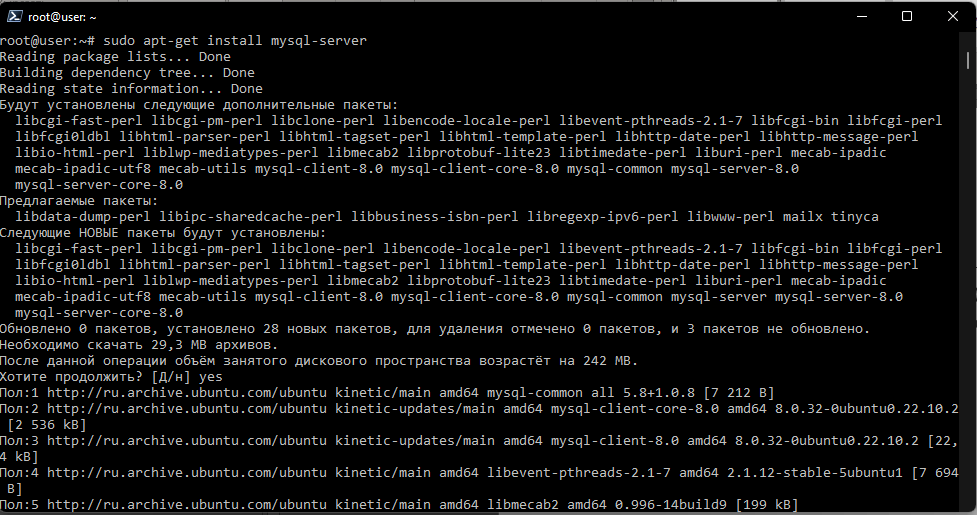


Рисунок 1 – Использование команды sudo apt-get install mysql-server

Для установки консольного клиента MySQL использовалась команда sudo apt-get install mysql-client.

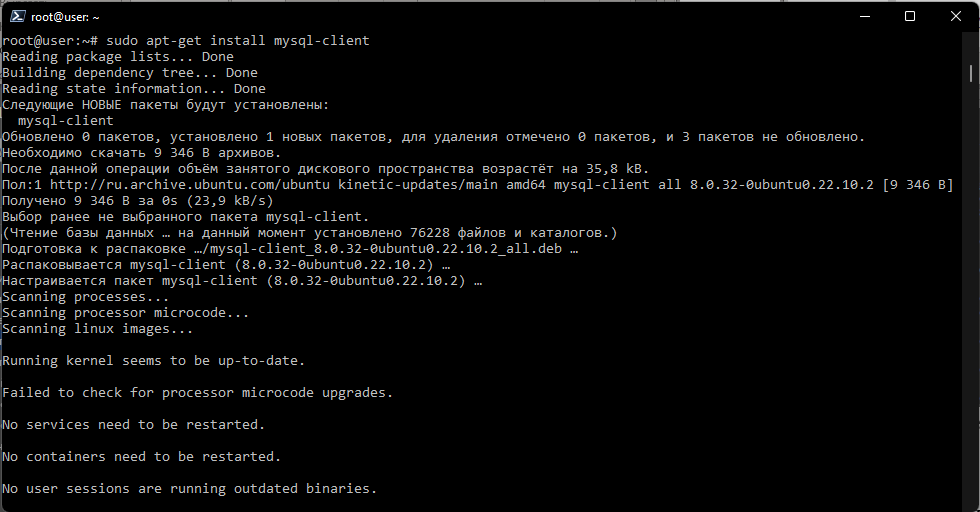


Рисунок 2 – Использование команды sudo apt-get install mysql-client

3.2 Установка PostgreSQL

Для установки PostgreSQL с набором часто используемых библиотек применялась команда sudo apt-get install postgresql postgresql-contrib.

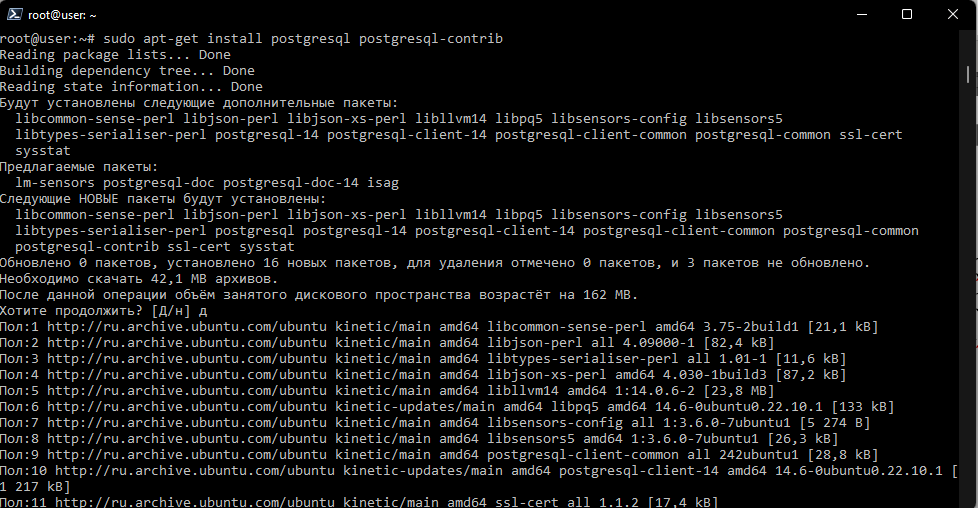


Рисунок 3 – Использование команды sudo apt-get install postgresql postgresql-contrib

4 Настройка СУБД

4.1 Настройка MySQL

4.1.1 Конфигурирование файлов

По умолчанию MySQL на платформе Ubuntu взаимодействует некорректно с кириллицей, так как установленной кодировкой по-умолчанию установлена latin1. Для изменения целевой кодировки по необходимо отредактировать файл my.cnf в операционной системе.

Первым этапом необходимо добавить в секцию [mysqld] следующие строки:

skip-character-set-client-handshake

character-set-server = utf8

init-connect='SET NAMES utf8'

collation-server=utf8\_general\_ci

Далее необходимо добавить в секциях [client] и [mysqldump] строчку

default-character-set=utf8

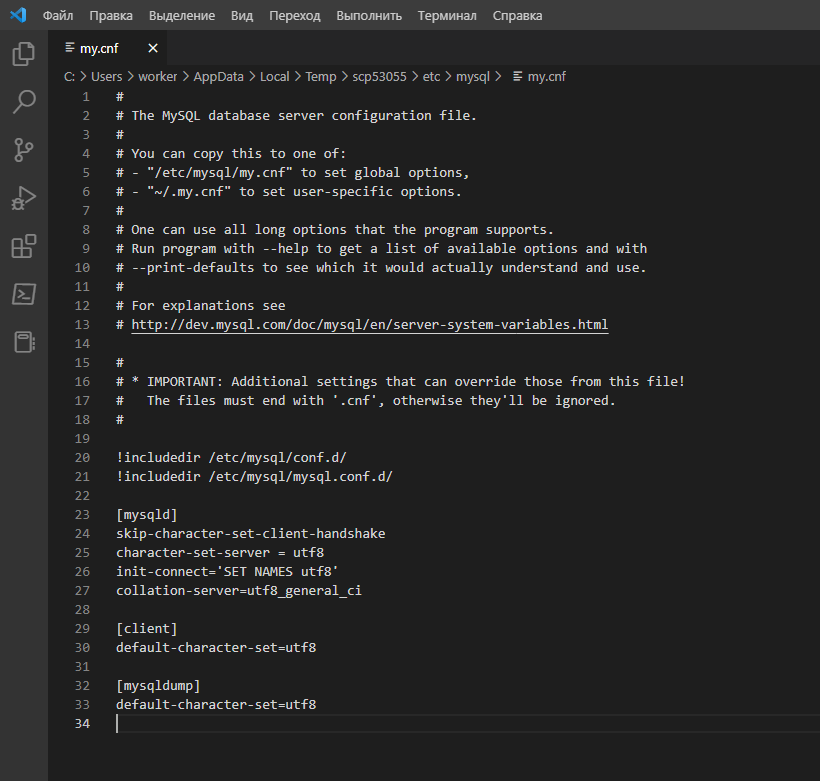


Рисунок 4 – Редактирование конфигурационного файла

Завершающим этапом выступает перезагрузка сервера MySQL командой sudo service mysql restart.

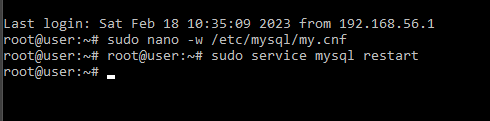


Рисунок 5 – Перезагрузка сервера

4.1.2 Создание базы данных

Для входа в оболочку MySQL используется команда sudo mysql.

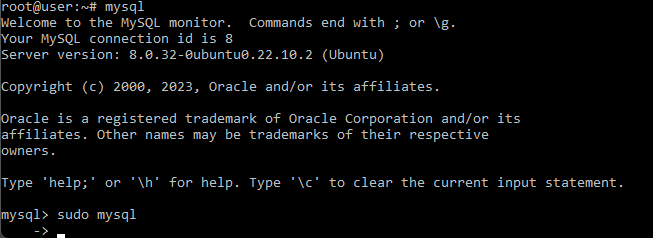


Рисунок 6 – Вход в оболочку MySQL

Чтобы создать новую базу данных MySQL используется команда CREATE DATABASE test\_db; test\_db в данной команде является названием создаваемой базы данных.

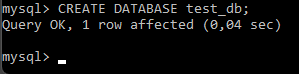


Рисунок 7 – Создание базы данных

Для просмотра всех существующих баз данных применяется команда SHOW DATABASES;

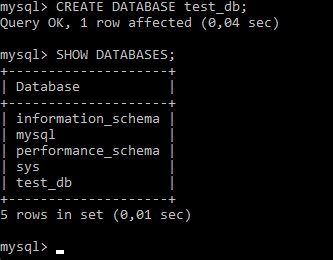


Рисунок 8 – Просмотр списка баз данных

Чтобы выбрать базу данных перед началом сеанса MySQL, используется команду USE test\_db;

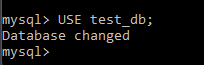


Рисунок 9 – Выбор базы данных

Создание таблицы осуществлялась командой

CREATE TABLE Student (id INT PRIMARY KEY AUTO\_INCREMENT, fullname VARCHAR(255) NOT NULL, student\_group VARCHAR(12) NOT NULL);

ALTER TABLE Student AUTO\_INCREMENT=1;

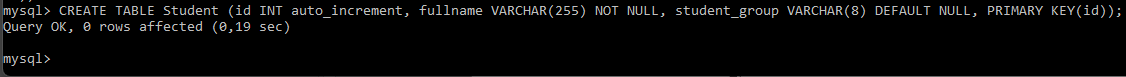


Рисунок 10 – Создание таблицы

Для просмотра списка колонок созданной таблицы использовалась команда show columns in Student;

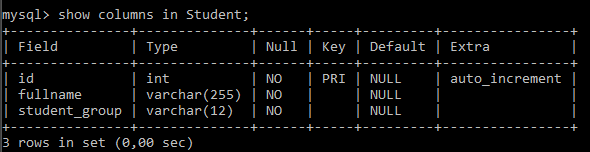


Рисунок 11 – Результат создания таблицы

Для внесения записей в таблицу применялась стандартная команда Insert.

INSERT INTO Student (fullname, student\_group) VALUES ('Иванов Иван Иванович', 'ИСМб-19-1');

INSERT INTO Student (fullname, student\_group) VALUES ('Пупкин Степан Григорьевич', 'ИНб-19-1');

INSERT INTO Student (fullname, student\_group) VALUES ('Иванова Кристина Давыдовна', 'ИНб-19-1');

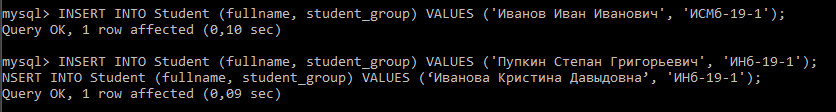


Рисунок 12 – Добавления записей

Просмотр записей в таблице Student осуществлялся с помощью команды select\* from Student;

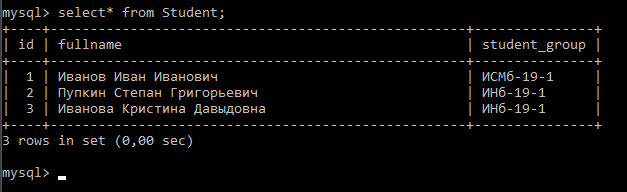


Рисунок 13 – Результат добавления записей

4.1.3 Создание роли

Новый пользователь в MySQL добавляется командой CREATE USER 'tester'@'localhost' IDENTIFIED BY '123';

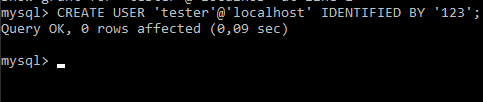


Рисунок 14 – Создание пользователя

Проверить права доступа роли можно с помощью команды show grants for 'tester'@'localhost';

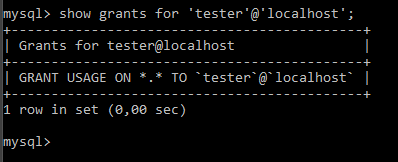


Рисунок 15 – Возможности пользователя

Для создания роли применяется команда create role selector\_role;

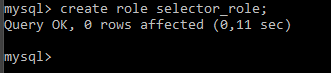


Рисунок 16 – Создание роли

Назначение прав пользования роли осуществляется с помощью команды grant select on test\_db.\* to selector\_role;

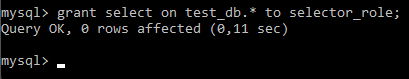


Рисунок 17 – Задание возможностей

Предоставление роли созданному пользователю осуществляется с помощью команды grant selector\_role to 'tester'@'localhost';

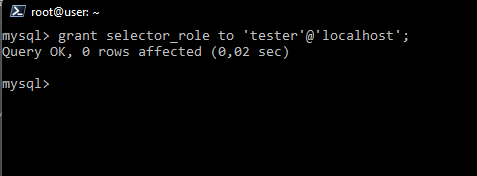


Рисунок 18 – Привязка пользователя к роли

В результате возможности пользователя были обновлены.

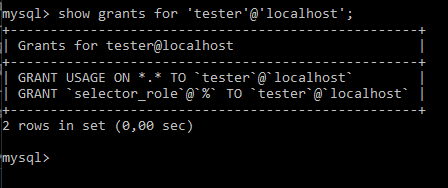


Рисунок 19 – Результат присвоения роли

Вход в созданного пользователя sudo mysql -u tester -p

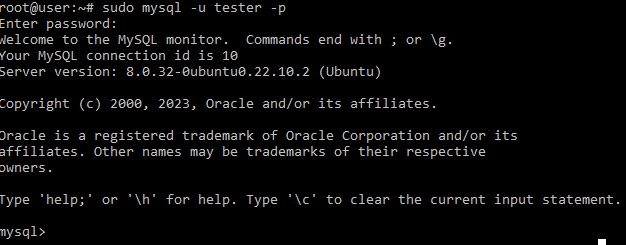


Рисунок 20 – Вход в пользователя Tester

Проверка установленной роли из доступных осуществляется с помощью вызова Select current\_role();

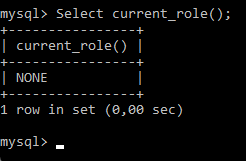


Рисунок 21 – Проверка роли

Установка роли для текущего пользователя осуществляется с помощью команды set role selector\_role;

Таким образом пользователь использует выбранную роль

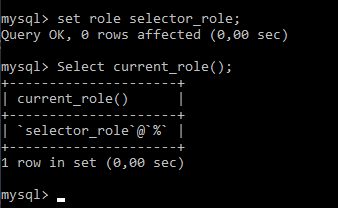


Рисунок 22 – Результат выбора роли

Далее была выбрана тестовая таблица, в которой просмотрены таблицы.

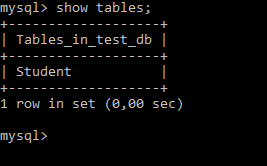


Рисунок 23 – Вызов таблиц базы данных

Так как при установлении прав роли было разрешено исключительно просматривать таблиц, то удаление данных из базы данных недоступно.

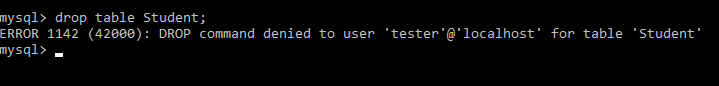


Рисунок 24 – Попытка удаления таблицы

Просмотр содержимого таблицы, в свою очередь, доступен для данной роли.

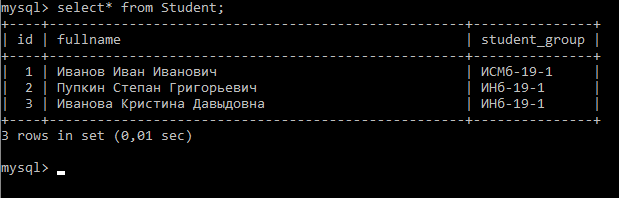


Рисунок 25 – Просмотр таблицы пользователем с заданной ролью

Удаление данных из таблицы также заблокировано.

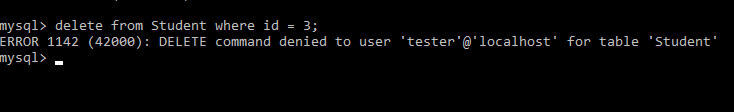


Рисунок 26 – Попытка удаления строки в таблице

Созданную роль можно присвоить пользователю по умолчанию. Для этого необходимо использовать команду set default role selector\_role to 'tester'@'localhost'; от имени администратора базы данных.

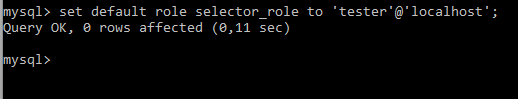


Рисунок 27 – Попытка удаления строки в таблице

4.2 Настройка PostgreSQL

4.2.1 Конфигурирование базы данных

Запуск сервиса PostgresSQL осуществляется с помощью команды sudo systemctl start postgresql.service

Проверить статус сервиса можно командой sudo systemctl status postgresql.service

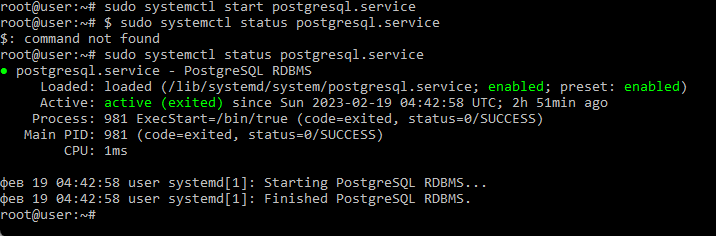


Рисунок 28 – Проверка статуса сервиса

Вход в администратора баз данных sudo -u postgres psql

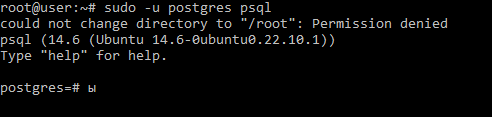


Рисунок 29 – Вход с помощью администратора

4.2.2 Создание базы данных

Чтобы создать новую базу данных PostgreSQL используется команда CREATE DATABASE test\_db; test\_db в данной команде является названием создаваемой базы данных.

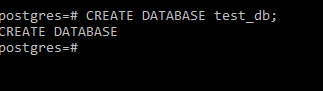


Рисунок 30 – Создание базы данных

Вывести список баз данных осуществляется командой \l. После вызова команды в списке баз данных созданная test\_db отображена на последней строке.

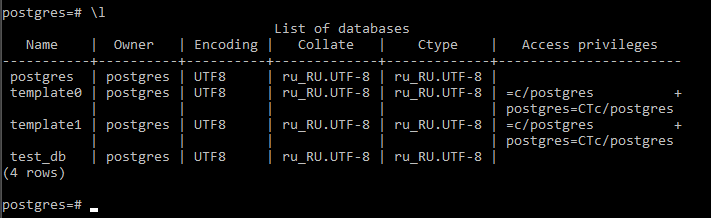


Рисунок 31 – Список баз данных

Подключение к созданной базе данных осуществляется командой \c test\_db



Рисунок 32 – Подключение к базе данных

Создание таблицы осуществляется командой CREATE TABLE Student (id SERIAL PRIMARY KEY, fullname VARCHAR(255) NOT NULL, student\_group VARCHAR(12) NOT NULL);

Для вывода имеющихся таблиц в базе данных применяется команда \dt

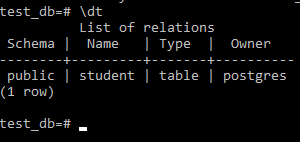


Рисунок 33 – Результат создания таблицы

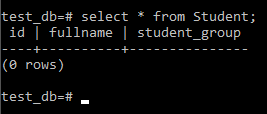


Рисунок 34 – Результат создания таблицы

Добавление данных в базу данных осуществляется с помощью команды Insert.

INSERT INTO Student (fullname, student\_group) VALUES ('Иванов Иван Иванович', 'ИСМб-19-1');

INSERT INTO Student (fullname, student\_group) VALUES ('Пупкин Степан Григорьевич', 'ИНб-19-1');

INSERT INTO Student (fullname, student\_group) VALUES ('Иванова Кристина Давыдовна', 'ИНб-19-1');

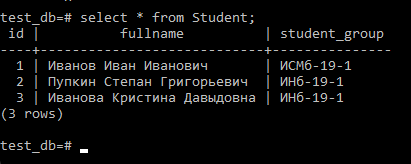


Рисунок 35 – Результат записи данных в таблицу

Удаление данных осуществляется командой delete from Student where id=3;

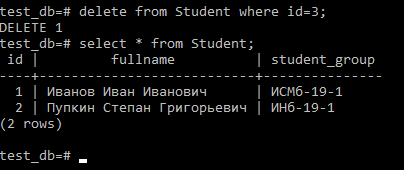


Рисунок 36 – Результат удаления записи из таблицы

4.2.3 Создание роли

Для создания пользователя используется команда CREATE USER tester WITH PASSWORD '123';

Для создания роли с возможностью «Только чтения» используется набор команд

CREATE ROLE tester\_role;

GRANT CONNECT ON DATABASE test\_db TO tester\_role;

GRANT SELECT ON TABLE Student TO tester\_role;

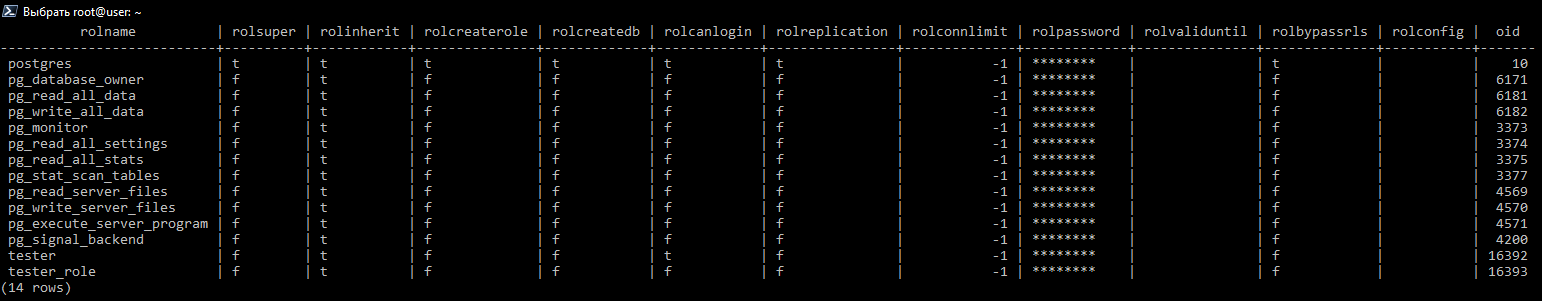


Рисунок 37 – Результат создания роли

Привязка роли к пользователю осуществляется командой GRANT tester\_role TO tester;



Рисунок 38 – Результат привязки роли

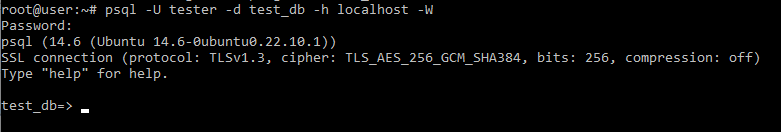


Рисунок 39 – Авторизация в созданного пользователя

Созданный пользователь обладает возможностью просмотра таблиц в закрепленной базе данных

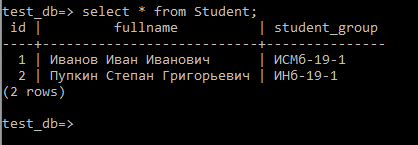


Рисунок 40 – Просмотр таблицы пользователем Tester

Операции удаления и добавления, в свою очередь, для пользователя с данной ролью недоступны.

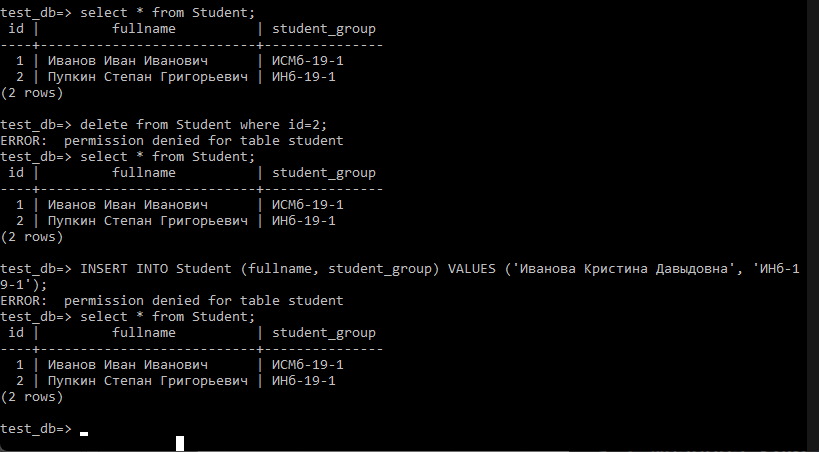


Рисунок 41 – Проверка удаления и добавления в таблицу

5 Заключение

В результате выполнения лабораторной работы были получены навыки практического администрирования баз данных в операционной системе UbuntuServer, изучены СУБД MySQL и PostgreSQL, команды и конфигурационные настройки, используемые для администрирования баз данных, имеющиеся базы данных были установлены и настроены для практического применения.