



МИНОБРНАУКИ РОССИИ  
Федеральное государственное бюджетное образовательное учреждение  
высшего образования  
«МИРЭА – Российский технологический университет»  
**РТУ МИРЭА**

---

## ОТЧЁТ ПО ПРАКТИКЕ В ПРОФИЛЬНОЙ СФЕРЕ

**Тема практики: «Администрирование PostgreSQL»**

Отчет представлен к  
рассмотрению:

Студент группы ИКБО-50-23

«31» мая 2025 г.

\_\_\_\_\_  
(подпись)

Павлов. Н.С.

Отчет утвержден.

Допущен к защите:

Руководитель практики

«31» мая 2025 г.

\_\_\_\_\_  
(подпись)

Тарланов А.Т.

Москва 2025 г.



МИНОБРНАУКИ РОССИИ  
Федеральное государственное бюджетное образовательное учреждение  
высшего образования  
«МИРЭА – Российский технологический университет»  
**РТУ МИРЭА**

---

---

**ИНДИВИДУАЛЬНОЕ ЗАДАНИЕ НА ПРАКТИКУ В ПРОФИЛЬНОЙ  
СФЕРЕ**

**Студенту 2 курса учебной группы ИКБО-50-23  
Павлову Никите Сергеевичу**

**Время практики:** с 26 мая 2025 г. по 31 мая 2024 г. \_\_\_\_\_

**Должность на практике:** студент \_\_\_\_\_

**1. ЦЕЛЕВАЯ УСТАНОВКА:** администрирование PostgreSQL

**2. СОДЕРЖАНИЕ ПРАКТИКИ:**

2.1 Введение

2.2 Основная часть

2.3 Заключение

**3. ОРГАНИЗАЦИОННО-МЕТОДИЧЕСКИЕ УКАЗАНИЯ:** в процессе практики рекомендуется использовать периодические издания и отраслевую литературу годом издания не старше 10 лет.

**СОГЛАСОВАНО:**

Руководитель практики  
«26» мая 2025 г.

\_\_\_\_\_  
(подпись)

(Тарланов А.Т.)

Задание получил  
«26» мая 2025 г.

\_\_\_\_\_  
(подпись)

(Павлов Н.С.)



МИНОБРНАУКИ РОССИИ  
Федеральное государственное бюджетное образовательное учреждение  
высшего образования  
«МИРЭА – Российский технологический университет»  
**РТУ МИРЭА**

**РАБОЧИЙ ГРАФИК ПРОВЕДЕНИЯ  
практики в профильной сфере**

студента Павлова Никиты Сергеевича 2 курса группы ИКБО-50-23.

№	Сроки выполнения	Этап	Отметка о выполнении
1	26.05.2025	Подготовительный этап, включающий в себя организационное собрание (Вводная лекция о порядке организации и прохождения практики)	выполнено
2	с 26.05.2025 по 29.05.2025	Выполнение задания по практике в соответствии с выданным заданием студента. (Мероприятия по сбору, обработке и структурированию материала, выполнение поставленной задачи)	выполнено
3	с 29.05.2025 по 30.05.2025	Подготовка отчета по практике (Оформление материалов отчета в полном соответствии с требованиями на оформление учебных работ студентов)	выполнено
4	31.05.2025	Представление отчета по практике к защите посредством загрузки на портал ДПО ( <a href="https://online-dpo.mirea.ru/">https://online-dpo.mirea.ru/</a> )	выполнено

**Согласовано:**

Руководитель практики \_\_\_\_\_ /Тарланов А.Т./

Обучающийся \_\_\_\_\_ / Павлов Н.С./

## СОДЕРЖАНИЕ

1. ВВЕДЕНИЕ .....	5
2 ОСНОВНАЯ ЧАСТЬ.....	6
2.1 ТЕОРЕТИЧЕСКАЯ ЧАСТЬ .....	6
2.2 ПРАКТИЧЕСКАЯ ЧАСТЬ.....	8
2.2.1 Установка PostgreSQL 17 на Ubuntu 24.04 .....	8
2.2.2 Проверка статуса службы PostgreSQL.....	8
2.2.3 Подключение к серверу через psql.....	8
2.2.4 Создание пользователя с правами суперпользователя .....	9
2.2.5 Создание базы данных и подключение под новым пользователем.....	9
2.2.6 Создание таблицы bicycle .....	10
2.2.7 Начальное заполнение таблицы .....	11
2.2.8 Проверка данных после начального заполнения.....	12
2.2.9 Дополнение таблицы новыми данными .....	12
2.2.10 Финальная проверка данных .....	13
4 ЗАКЛЮЧЕНИЕ .....	14
СПИСОК ЛИТЕРАТУРЫ.....	15

## 1. ВВЕДЕНИЕ

Современные приложения, особенно в эпоху микросервисной архитектуры распределенных систем, требуют надежных, масштабируемых и гибких решений для управления данными. PostgreSQL, являясь объектно-реляционной системой управления базами данных с открытым исходным кодом, занимает ключевое место в экосистеме DevOps благодаря своей стабильности, расширяемости и поддержке сложных операций. Её использование позволяет не только эффективно хранить и обрабатывать данные, но и интегрировать процессы администрирования в конвейеры непрерывной интеграции и доставки (CI/CD), что критически важно для автоматизации и ускорения жизненного цикла разработки.

Администрирование баз данных в контексте DevOps выходит за рамки традиционных задач настройки и оптимизации. Оно подразумевает автоматизацию рутинных операций, таких как развертывание кластеров, настройка репликации, управление резервным копированием, мониторинг производительности и обеспечение безопасности. PostgreSQL, обладая богатым набором инструментов и возможностей, становится идеальной платформой для реализации этих задач, позволяя согласовать работу с данными с принципами Infrastructure as Code (IaC) и гибкого управления конфигурациями

## 2 ОСНОВНАЯ ЧАСТЬ

### 2.1 ТЕОРЕТИЧЕСКАЯ ЧАСТЬ

#### Особенности PostgreSQL, значимые для DevOps:

1. **Расширяемость:** Поддержка пользовательских типов данных, функций и расширений позволяет адаптировать СУБД под специфические требования проектов.
2. **Транзакционная надежность:** ACID-совместимость и механизм WAL (Write-Ahead Logging) гарантируют целостность данных даже в условиях высокой нагрузки.
3. **Репликация и кластеризация:** Встроенные решения для потоковой репликации и инструменты вроде Patroni упрощают построение отказоустойчивых кластеров.
4. **Интеграция с DevOps-инструментами:** Совместимость с Ansible, Terraform, Kubernetes и системами мониторинга (Prometheus, Grafana) обеспечивает seamless-взаимодействие в CI/CD-цепочках

#### Преимущества использования PostgreSQL в DevOps:

1. **Автоматизация развертывания:** Использование IaC-подходов позволяет быстро создавать и масштабировать экземпляры БД в облачных и гибридных средах.
2. **Безопасность и аудит:** Гибкие ролевые модели, SSL-шифрование и интеграция с Vault обеспечивают соответствие стандартам compliance.
3. **Эффективное управление ресурсами:** Возможности параллельной обработки запросов и настройки индексов снижают нагрузку на инфраструктуру.
4. **Резервное копирование и восстановление:** Инструменты pg\_dump, pg\_basebackup и WAL-архивация минимизируют риски потери данных.
5. **Мониторинг и оптимизация:** Анализ производительности через pg\_stat\_statements и настройка запросов повышают отзывчивость приложений.

## **Основные компоненты экосистемы PostgreSQL для администрирования:**

- Процессы СУБД: Postmaster, Writer, Checkpointer, WAL Sender/Receiver.
- Конфигурационные файлы: postgresql.conf, pg\_hba.conf, recovery.conf.
- Утилиты командной строки: psql, pg\_ctl, pgbench.
- Инструменты оркестрации: Patroni, repmgr, PostgreSQL Operator для Kubernetes.

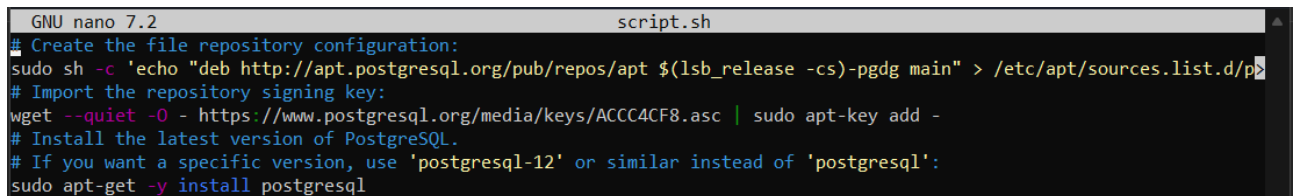
В рамках практики основное внимание было уделено автоматизации процессов администрирования: создание Ansible-ролей для развертывания кластеров, настройка потоковой репликации, интеграция с системами мониторинга, а также разработка скриптов для резервного копирования и восстановления. Эти задачи направлены на сокращение ручного вмешательства, повышение отказоустойчивости и обеспечение согласованности окружений на всех этапах CI/CD.

Использование PostgreSQL в связке с DevOps-практиками не только ускоряет delivery приложений, но и формирует основу для построения надежной, масштабируемой и безопасной data-инфраструктуры, что подтверждается результатами, достигнутыми в ходе производственной практики.

## 2.2 ПРАКТИЧЕСКАЯ ЧАСТЬ

### 2.2.1 Установка PostgreSQL 17 на Ubuntu 24.04

Для установки PostgreSQL 17 на Ubuntu 24.04 был добавлен официальный репозиторий PostgreSQL, так как версия 17 отсутствует в стандартных репозиториях ОС. После настройки источника пакетов выполнена установка сервера PostgreSQL и сопутствующих утилит. Это обеспечило доступ к стабильной и поддерживаемой версии СУБД, совместимой с требованиями проекта.

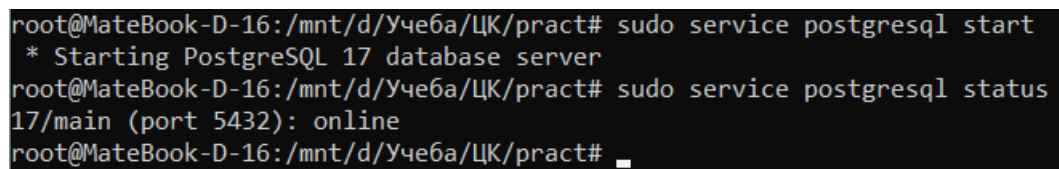


```
GNU nano 7.2 script.sh
# Create the file repository configuration:
sudo sh -c 'echo "deb http://apt.postgresql.org/pub/repos/apt $(lsb_release -cs)-pgdg main" > /etc/apt/sources.list.d/pgdg.list'
# Import the repository signing key:
wget --quiet -O - https://www.postgresql.org/media/keys/ACCC4CF8.asc | sudo apt-key add -
# Install the latest version of PostgreSQL.
# If you want a specific version, use 'postgresql-12' or similar instead of 'postgresql':
sudo apt-get -y install postgresql
```

Рисунок 1 – Скрипт установки PostgreSQL

### 2.2.2 Проверка статуса службы PostgreSQL

После установки проверен статус системного демона postgresql.service. Служба успешно запущена и активна, что подтверждает корректность установки и готовность к работе.



```
root@MateBook-D-16:/mnt/d/Учеба/ЦК/pract# sudo service postgresql start
* Starting PostgreSQL 17 database server
root@MateBook-D-16:/mnt/d/Учеба/ЦК/pract# sudo service postgresql status
17/main (port 5432): online
root@MateBook-D-16:/mnt/d/Учеба/ЦК/pract#
```

Рисунок 2 – Запуск службы PostgreSQL и проверка работоспособности

### 2.2.3 Подключение к серверу через psql

Интерактивная консоль psql использована для управления базой данных. Подключение выполнено от имени системного пользователя postgres, что является стандартным подходом для начальной настройки. Проверка соединения (\conninfo) подтвердила корректность аутентификации и параметров подключения.



```

root@MateBook-D-16:/mnt/d/Учеба/ЦК/pract# sudo -i -u postgres
Welcome to Ubuntu 24.04 LTS (GNU/Linux 5.10.16.3-microsoft-standard-WSL2 x86_64)

* Documentation:  https://help.ubuntu.com
* Management:    https://landscape.canonical.com
* Support:       https://ubuntu.com/pro

System information as of Sat May 24 11:09:17 MSK 2025

System load:  0.01          Processes:            18
Usage of /:   0.8% of 250.98GB Users logged in:         0
Memory usage: 2%           IPv4 address for eth0: 172.26.68.211
Swap usage:   0%

This message is shown once a day. To disable it please create the
/var/lib/postgresql/.hushlogin file.
postgres@MateBook-D-16:~$

```

Рисунок 3 – Вход в аккаунт postgres

```

postgres@MateBook-D-16:~$ psql
psql (17.5 (Ubuntu 17.5-1.pgdg24.04+1))
Type "help" for help.

postgres=# \conninfo
You are connected to database "postgres" as user "postgres" via socket in "/var/run/postgresql" at port "5432".
postgres=# \q
postgres@MateBook-D-16:~$

```

Рисунок 4 – Проверка работоспособности консоли

## 2.2.4 Создание пользователя с правами суперпользователя

С помощью утилиты createuser создан новый пользователь nikita с правами суперпользователя. Это позволяет выполнять административные задачи без использования учётной записи postgres, что повышает безопасность системы.

```

postgres@MateBook-D-16:~$ createuser --interactive
Enter name of role to add: nikita
Shall the new role be a superuser? (y/n) y
postgres@MateBook-D-16:~$ man createuser
postgres@MateBook-D-16:~$

```

Рисунок 5 – Создание нового пользователя

## 2.2.5 Создание базы данных и подключение под новым пользователем

После создания пользователя выполнено подключение к PostgreSQL под учётной записью nikita. Автоматически создана одноимённая база данных, что соответствует поведению PostgreSQL при первом входе пользователя. Проверка подключения (\conninfo) подтвердила доступ к новой БД.

```

root@MateBook-D-16:/mnt/d/Учеба/ЦК/pract# sudo -i -u nikita
Welcome to Ubuntu 24.04 LTS (GNU/Linux 5.10.16.3-microsoft-standard-WSL2 x86_64)

* Documentation:  https://help.ubuntu.com
* Management:    https://landscape.canonical.com
* Support:       https://ubuntu.com/pro

System information as of Sat May 24 11:25:42 MSK 2025

System load:  0.0           Processes:      18
Usage of /:   0.8% of 250.98GB Users logged in: 0
Memory usage: 2%           IPv4 address for eth0: 172.26.68.211
Swap usage:   0%

This message is shown once a day. To disable it please create the
/home/nikita/.hushlogin file.
nikita@MateBook-D-16:~$ psql
psql (17.5 (Ubuntu 17.5-1.pgdg24.04+1))
Type "help" for help.

nikita=# \conninfo
You are connected to database "nikita" as user "nikita" via socket in "/var/run/postgresql" at port "5432".
nikita=#

```

Рисунок 6 – Создание базы данных под учетной записью nikita

## 2.2.6 Создание таблицы bicycle

В базе данных создана таблица bicycle с полями:

- bicycle\_id – Первичный ключ, имеет тип serial для автоматического обновления индекса.
- brand – Наименование производителя, имеет тип varchar (50), не может быть нулевым.
- model – Наименование модели, имеет тип varchar (50), не может быть нулевым.
- year\_of\_release – Год выпуска модели, имеет тип smallint, не может быть нулевым.
- color – Цвет рамы, имеет тип varchar (30), не может быть нулевым.

Проверка структуры (\d, \dt) подтвердила корректность создания таблицы и связанной последовательности для bicycle\_id.

```

nikita=# CREATE TABLE bicycle (
nikita(# bicycle_id serial PRIMARY KEY,
nikita(# brand varchar(50) NOT NULL,
nikita(# model varchar(50) NOT NULL,
nikita(# year_of_release smallint NOT NULL,
nikita(# color varchar(30) NOT NULL
nikita(# );
CREATE TABLE
nikita=# \d

```

List of relations			
Schema	Name	Type	Owner
public	bicycle	table	nikita
public	bicycle_bicycle_id_seq	sequence	nikita

(2 rows)

```

nikita=# \dt

```

List of relations			
Schema	Name	Type	Owner
public	bicycle	table	nikita

(1 row)

Рисунок 7 – Создание и просмотр базы данных

### 2.2.7 Начальное заполнение таблицы

В таблицу добавлены три тестовые записи:

- Производитель: Scott, Модель: Spark RC World Cup EVO, Год выпуска: 2025, Цвет: carbon black;
- Производитель: Colnago, Модель: Y1Rs, Год выпуска: 2025, Цвет: UAE Emirates;
- Производитель: Specialized, Модель: S-Works Epic 8, Год выпуска: 2025, Цвет: Gloss Carbon;

Операция вставки (INSERT) выполнена успешно, что подтверждено сообщением INSERT 0 3.

```

nikita=# INSERT INTO bicycle (brand, model, year_of_release, color) VALUES
nikita-# ('Scott', 'Spark RC World Cup EVO', 2025, 'carbon black'),
nikita-# ('Colnago', 'Y1Rs', 2025, 'UAE Emirates'),
nikita-# ('Specialized', 'S-Works Epic 8', 2025, 'Gloss Carbon');
INSERT 0 3

```

Рисунок 8 – Операция заполнения таблицы данными

### 2.2.8 Проверка данных после начального заполнения

Запрос `SELECT * FROM bicycle;` вывел все добавленные записи, что подтвердило целостность данных.

```
nikita=# SELECT * FROM bicycle;
```

bicycle_id	brand	model	year_of_release	color
1	Scott	Spark RC World Cup EVO	2025	carbon black
2	Colnago	Y1Rs	2025	UAE Emirates
3	Specialized	S-Works Epic 8	2025	Gloss Carbon

```
(3 rows)
```

Рисунок 9 – Использование оператора SELECT для вывода записей на экран

### 2.2.9 Дополнение таблицы новыми данными

В таблицу добавлены дополнительные записи, включая топовое снаряжение от различных производителей за 2024 и 2025 года. Это имитирует реальное использование БД и демонстрирует масштабируемость структуры.

```
nikita=# INSERT INTO bicycle (brand, model, year_of_release, color) VALUES
nikita=# -- Шоссейные велосипеды
nikita=# ('Specialized', 'S-Works Tarmac SL8', 2024, 'Red/Black'),
nikita=# ('Trek', 'Emonda SLR 9', 2024, 'Blue/White'),
nikita=# ('Canyon', 'Aeroad CFR', 2025, 'Stealth Black'),
nikita=# ('Pinarello', 'Dogma F', 2024, 'Team Colors'),
nikita=# ('Cervelo', 'S5', 2025, 'Black/Red'),
nikita=#
nikita=# -- Горные велосипеды (MTB)
nikita=# ('Santa Cruz', 'Hightower 3', 2024, 'Gloss Green'),
nikita=# ('Yeti', 'SB160', 2025, 'Turquoise'),
nikita=# ('Trek', 'Fuel EX 9.9', 2024, 'Matte Black'),
nikita=# ('Specialized', 'S-Works Epic EVO', 2025, 'Carbon/Red'),
nikita=# ('Canyon', 'Lux CFR', 2024, 'Yellow/Black'),
nikita=#
nikita=# -- Гравел-велосипеды
nikita=# ('Cannondale', 'SuperSix EVO SE', 2024, 'Olive Green'),
nikita=# ('Specialized', 'S-Works Crux', 2025, 'White/Black'),
nikita=# ('Trek', 'Checkpoint SLR 9', 2024, 'Blue/Grey'),
nikita=# ('Cervelo', 'Aspero-5', 2025, 'Gloss Black'),
nikita=# ('Scott', 'Addict Gravel Tuned', 2024, 'Red/White'),
nikita=#
nikita=# -- Городские велосипеды
nikita=# ('VanMoof', 'S5', 2024, 'Light Grey'),
nikita=# ('Gazelle', 'Ultimate C8+ HMB', 2025, 'Dark Blue');
INSERT 0 17
```

Рисунок 10 – Добавление новых данных в таблицу БД

### 2.2.10 Финальная проверка данных

Повторный запрос `SELECT * FROM bicycle;` отобразил 20 записей, включая новые данные. Это подтверждает успешное выполнение операций обновления и отсутствие конфликтов при масштабировании

```
nikita=# SELECT * FROM bicycle;
```

bicycle_id	brand	model	year_of_release	color
1	Scott	Spark RC World Cup EVO	2025	carbon black
2	Colnago	Y1Rs	2025	UAE Emirates
3	Specialized	S-Works Epic 8	2025	Gloss Carbon
4	Specialized	S-Works Tarmac SL8	2024	Red/Black
5	Trek	Emonda SLR 9	2024	Blue/White
6	Canyon	Aeroad CFR	2025	Stealth Black
7	Pinarello	Dogma F	2024	Team Colors
8	Cervelo	S5	2025	Black/Red
9	Santa Cruz	Hightower 3	2024	Gloss Green
10	Yeti	SB160	2025	Turquoise
11	Trek	Fuel EX 9.9	2024	Matte Black
12	Specialized	S-Works Epic EVO	2025	Carbon/Red
13	Canyon	Lux CFR	2024	Yellow/Black
14	Cannondale	SuperSix EVO SE	2024	Olive Green
15	Specialized	S-Works Crux	2025	White/Black
16	Trek	Checkpoint SLR 9	2024	Blue/Grey
17	Cervelo	Aspero-5	2025	Gloss Black
18	Scott	Addict Gravel Tuned	2024	Red/White
19	VanMoof	S5	2024	Light Grey
20	Gazelle	Ultimate C8+ HMB	2025	Dark Blue

(20 rows)

Рисунок 11 – Использование оператора SELECT для вывода всех записей БД

#### 4 ЗАКЛЮЧЕНИЕ

В ходе прохождения практики я углубил свои знания в области СУБД и получил ценный практический опыт работы с PostgreSQL. Мне удалось не только ознакомиться с теоретическими основами работы с реляционными базами данных, но и на практике установить и настроить PostgreSQL на своей системе. Я добавил официальный репозиторий PostgreSQL, выполнил установку актуальной версии СУБД, создал новую роль, а также базу данных и таблицы с пользовательской структурой.

В рамках практики я научился выполнять базовые операции с PostgreSQL: создавать и заполнять таблицы, выполнять выборки данных, удалять записи, а также переключаться между ролями и использовать psql-интерфейс для администрирования. Особенно полезным оказался опыт настройки окружения для взаимодействия с базой данных и понимание принципов работы PostgreSQL-ролей и прав доступа.

Выполнение всех этапов способствовало развитию навыков работы с системами управления базами данных, администрирования в Linux-среде и повысило уверенность в применении PostgreSQL в будущих проектах. Полученные знания будут полезны как при разработке программных решений, так и при организации хранения и обработки данных.

Считаю, что поставленные задачи выполнены в полном объеме, а цели практики достигнуты.

## СПИСОК ЛИТЕРАТУРЫ

1. Риз Д. PostgreSQL: настройка и масштабирование / пер. с англ. А. Слинкина. — М.: ДМК Пресс, 2021. — 398 с.
2. Методические рекомендации по проведению итоговой аттестации “DevOps-инженер с нуля” / ПАО «Ростелеком». — Версия 1.0.4. — 2025. — 79 с.
3. Official PostgreSQL Documentation [Электронный ресурс]. — URL: <https://www.postgresql.org/docs> (дата обращения: 02.05.2025).
4. ГОСТ Р 7.0.100-2018. Библиографическая запись. Библиографическое описание. Общие требования и правила составления : национальный стандарт Российской Федерации : дата введения 2019-07-01 / Федеральное агентство по техническому регулированию. – Изд. официальное. – Москва : Стандартинформ, 2018. – 124 с.