

**Выполнил:** студент 4 курса группы ПИЖ-б-о-22-1 Плугатырев Владислав Алексеевич

## Тема:

Установка PostgreSQL и базовое управление сервером.

## Цель работы:

Освоить основные способы установки PostgreSQL (из пакетов и исходных кодов) в предустановленном окружении; получить практические навыки создания кластера баз данных, запуска и остановки сервера, а также проверки его состояния; научиться использовать утилиты `initdb`, `pg_ctl`, `pg_checksums` и `psql` для базового взаимодействия с СУБД.

---

## Порядок выполнения работы

### 1. Установка из пакетов (через Homebrew) и управление кластером

```
admin@MacBook-Air-Admin ~ % brew install postgresql@16
[==> Fetching downloads for: postgresql@16
==> Downloading https://ghcr.io/v2/homebrew/core/postgresql/16/manifests/16.10
##### 100.0%
==> Fetching dependencies for postgresql@16: krb5 and gettext
==> Downloading https://ghcr.io/v2/homebrew/core/krb5/manifests/1.22.1
##### 100.0%
==> Fetching krb5
==> Downloading https://ghcr.io/v2/homebrew/core/krb5/blobs/sha256:902884745603
##### 100.0%
==> Downloading https://ghcr.io/v2/homebrew/core/gettext/manifests/0.26
##### 100.0%
==> Fetching gettext
==> Downloading https://ghcr.io/v2/homebrew/core/gettext/blobs/sha256:b25ac1e62
##### 100.0%
==> Fetching postgresql@16
==> Downloading https://ghcr.io/v2/homebrew/core/postgresql/16/blobs/sha256:b6e
##### 100.0%
==> Installing dependencies for postgresql@16: krb5 and gettext
==> Installing postgresql@16 dependency: krb5
```

Рисунок 1.1 — Установка PostgreSQL 16 версии

```
[admin@MacBook-Air-Admin ~ % brew services start postgresql@16
==> Successfully started `postgresql@16` (label: homebrew.mxcl.postgresql@16)
[admin@MacBook-Air-Admin ~ % brew services list
Name                                Status User File
mongodb-community@8.0 started admin ~/Library/LaunchAgents/homebrew.mxcl.mongod
munity@8.0.plist
postgresql@16 started admin ~/Library/LaunchAgents/homebrew.mxcl.postgr
16.plist
unbound                             none
```

*Рисунок 1.2 — Запуск сервера и проверка статуса*

```
admin@MacBook-Air-Admin ~ % pgrep -l postgres
19584 postgres
[19587 postgres
19588 postgres
19590 postgres
[19591 postgres
19592 postgres
admin@MacBook-Air-Admin ~ % lsof -i :5432
COMMAND PID USER FD TYPE DEVICE SIZE/OFF NODE NAME
postgres 19584 admin 7u IPv6 0x1aa76578e534d5a7 0t0 TCP localhost:postgresql (LISTEN)
postgres 19584 admin 8u IPv4 0x135d06bee384f6e 0t0 TCP localhost:postgresql (LISTEN)
```

*Рисунок 1.3 — Проверка кластера*

```
[admin@MacBook-Air-Admin ~ % pg_checksums --check -D /opt/homebrew/var/postgresql@16
Обработка контрольных сумм завершена
Просканировано файлов: 948
Просканировано блоков: 2853
Неверные контрольные суммы: 0
Версия контрольных сумм данных: 1
```

*Рисунок 1.4 — Контрольные суммы*

```
[admin@MacBook-Air-Admin ~ % psql -U postgres -d postgres
psql (16.10 (Homebrew))
Введите "help", чтобы получить справку.

postgres=# SELECT version();
version
-----
 PostgreSQL 16.10 (Homebrew) on aarch64-apple-darwin24.4.0, compiled by Apple clang version 17.0.6
 700.0.13.3), 64-bit
(1 строка)

postgres=# \q
admin@MacBook-Air-Admin ~ % █
```

*Рисунок 1.5 — Проверка работоспособности*

## 2. Установка из исходных кодов

```

admin@MacBook-Air-Admin postgresql-16.10 % mkdir -p ~/pgdata16
admin@MacBook-Air-Admin postgresql-16.10 % initdb -D ~/pgdata16 -U postgres --data-checksums
The files belonging to this database system will be owned by user "admin".
This user must also own the server process.

The database cluster will be initialized with locale "ru_RU.UTF-8".
The default database encoding has accordingly been set to "UTF8".
The default text search configuration will be set to "russian".

Data page checksums are enabled.

fixing permissions on existing directory /Users/admin/pgdata16 ... ok
creating subdirectories ... ok
selecting dynamic shared memory implementation ... posix
selecting default max_connections ... 100
selecting default shared_buffers ... 128MB
selecting default time zone ... Europe/Moscow
creating configuration files ... ok
running bootstrap script ... ok
performing post-bootstrap initialization ... ok
syncing data to disk ... ok

initdb: warning: enabling "trust" authentication for local connections
initdb: hint: You can change this by editing pg_hba.conf or using the option -A, or --auth-local and --auth-host, the next time you run initdb.

Success. You can now start the database server using:

    pg_ctl -D /Users/admin/pgdata16 -l logfile start

admin@MacBook-Air-Admin postgresql-16.10 % █

```

*Рисунок 1.6 — Конфигурация, сборка и создание каталога PGDATA и инициализация кластера*

```

admin@MacBook-Air-Admin postgresql-16.10 % pg_ctl -D ~/pgdata16 -l ~/pgdata16/logfile start
waiting for server to start.... done
server started
admin@MacBook-Air-Admin postgresql-16.10 % pg_ctl -D ~/pgdata16 status
pg_ctl: server is running (PID: 43532)
/Users/admin/pg16_source_install/bin/postgres "-D" "/Users/admin/pgdata16"
admin@MacBook-Air-Admin postgresql-16.10 % psql -U postgres -d postgres -p 5555 -h localhost
psql (16.10)
Type "help" for help.

postgres=# SELECT version();
               version
-----
PostgreSQL 16.10 on aarch64-apple-darwin24.3.0, compiled by Apple clang version 16.0.0 (clang-1600.0.26.6), 64-bit
(1 row)

postgres=# \q

```

*Рисунок 1.7 — Запуск сервера, проверка статуса и подключение*

## 3. Объяснение важности контрольных сумм (Data Checksums)

Контрольные суммы страниц данных — это механизм защиты целостности данных на физическом уровне. При включенной опции `--data-checksums` PostgreSQL рассчитывает и сохраняет контрольную сумму для каждой страницы данных при записи на диск. При последующем чтении этой страницы сервер заново вычисляет контрольную сумму и сравнивает ее с сохраненным значением.

### Зачем это нужно:

1. **Обнаружение повреждений данных:** Основная цель — выявление повреждений.

2. **Предотвращение «тихого» повреждения:** это самый опасный тип повреждения, при котором данные на диске повреждаются, но система продолжает работать, не подозревая об этом. Без контрольных сумм такие данные могут быть прочитаны и попасть в отчеты или резервные копии, что приведет к порче бэкапов и принятию неверных решений на основе искаженной информации.

Контрольные суммы не предотвращают повреждение данных, а лишь обнаруживают его.

---

## 4. Сравнение способов установки

### Установка из пакетов (через Homebrew)

- **Простота и скорость:** требует минимальных усилий — одной команды `brew install postgresql@16`, зависимости разрешаются автоматически.
- **Ограниченный контроль:** управление через высокоуровневые команды-обертки (`brew services`), что скрывает детали работы системы.
- **Идеально для:** разработки, тестирования и быстрого развертывания.

### Установка из исходного кода

- **Время и сложность:** процесс занимает десятки минут из-за компиляции, требует ручного разрешения зависимостей.
  - **Полный контроль:** возможность выбора компонентов, указания путей, портов, оптимизаций под платформу.
  - **Управление через нативные утилиты:** `pg_ctl`, что требует глубокого понимания СУБД.
- 

## Вывод

В ходе работы были освоены основные способы установки PostgreSQL (из пакетов и исходных кодов) в предустановленном окружении; получены практические навыки создания кластера баз данных, запуска и остановки сервера, а также проверки его состояния; научился использовать утилиты `initdb`, `pg_ctl`, `pg_checksums` и `psql` для базового взаимодействия с СУБД.