

# Практическая работа. Создание кластера базы данных PostgreSQL

В этой практической работе вы создадите кластер еще одной управляемой БД, на этот раз PostgreSQL, подключитесь к ней и загрузите в нее данные. Вы сможете убедиться, что принципы организации сервисов управляемых БД в Yandex Cloud практически одинаковы. Хорошо изучив один из них, вы будете легко ориентироваться и в остальных.

## Создание кластера

Создание кластера управляемой базы данных PostgreSQL аналогично созданию кластера базы данных MySQL.

Перейдите в сервис управляемых баз данных PostgreSQL и нажмите кнопку **Создать кластер**.

В появившемся окне настроек задайте необходимые параметры.

1. Имя кластера и его описание. Выберите уникальное в облаке имя кластера. Описание опционально, поэтому можно оставить это поле пустым.
2. В поле **Окружение** выберите `PRODUCTION`.
3. Выберите версию PostgreSQL и класс хоста.
4. Выберите размер и тип сетевого хранилища.
5. Задайте атрибуты базы данных.
6. Выберите из списка сеть, в которой будут находиться хосты кластера (для подключения потребуются публичные хосты).
7. В блоке **Хосты** добавьте ещё два хоста в других зонах доступности для обеспечения отказоустойчивости кластера. База автоматически реплицируется.
8. В блоке **Дополнительные настройки** задайте время начала резервного копирования и включите доступ из консоли управления.
9. Нажмите кнопку **Создать кластер**.

## Подключение

Как и в случае с MySQL, к хостам кластера Managed Service for PostgreSQL можно подключиться двумя способами.

### **Через интернет**

Если вы настроили публичный доступ для нужного хоста, то подключиться к нему можно с помощью SSL-соединения.

### **С виртуальных машин Yandex Cloud**

Они должны быть расположены в той же облачной сети. Если к хосту нет публичного доступа, для подключения с таких виртуальных машин SSL-соединение использовать необязательно. Обратите внимание, что если публичный доступ в вашем кластере настроен только для некоторых хостов, автоматическая смена мастера может привести к тому, что вы не сможете подключиться к мастеру из интернета. Установите клиент для подключения к БД PostgreSQL. Команда установки в Ubuntu:

Скопировать код

```
sudo apt update && sudo apt install -y postgresql-client
```

Скачайте сертификат для подключения к БД PostgreSQL:

Скопировать код

```
mkdir -p ~/.postgresql
```

```
wget "https://storage.yandexcloud.net/cloud-certs/CA.pem" -O  
~/.postgresql/root.crt
```

```
chmod 0600 ~/.postgresql/root.crt
```

Пример команды для подключения можно посмотреть в консоли управления, нажав на кнопку **Подключиться** на странице кластера.

Подключение с SSL происходит при помощи следующей команды:

```
psql "host=<FQDN_хоста> \  
port=6432 \  
sslmode=verify-full \  
dbname=<имя базы данных> \  
user=<имя пользователя базы данных> \  
target_session_attrs=read-write"
```

## **Загрузка данных в базу данных из CSV**

Одним из способов добавления данных в базу является их загрузка из csv-файла.

Предположим, вы используете БД для организации работы транспортной службы интернет-магазина. Вам нужно добавить в базу

таблицу, содержащую данные о расстояниях между складом и пунктами самовывоза, а также о стандартном времени доставки товаров со склада в эти пункты. Создадим csv-файл, например DTM.csv, который содержит такие данные (100 - код склада, 101-109 - коды пунктов, Time - стандартное время доставки в минутах, Distance - расстояние в километрах):

Скопировать код

```
"depot","store","time","distance"
"100","101",31,12
"100","102",38,17
"100","103",56,33
"100","104",70,60
"100","105",41,25
"100","106",21,8
"100","107",33,14
"100","108",62,42
"100","109",45,29
```

Важные моменты при миграции из CSV:

1. Названия колонок в файле и в таблице необязательно совпадают.
2. Файл содержит заголовок, который не нужно импортировать.
3. Первые 2 колонки конвертируем из строк (string) в целые числа (int).

PostgreSQL позволяет импортировать данные из файла несколькими способами:

1. Командой `copy`.
2. Через функции `pl/pgsql`.
3. Средствами другого языка, например Python.

Воспользуемся первым способом.

Сначала нам понадобится создать таблицу, в которую будет осуществлена миграция данных. Подключитесь к БД согласно инструкциям выше. Выполните следующую команду:

Скопировать код

```
CREATE TABLE dtm (
    id serial PRIMARY KEY,
    depot int NOT NULL,
    store int NOT NULL,
    time int NOT NULL,
```

```
distance int NOT NULL  
);
```

Загрузите данные:

Скопировать код

```
\copy dtm(depot,store,time,distance) from '/<путь к файлу>/DTM.csv'  
DELIMITERS ',' CSV HEADER;
```

В этой команде мы учли те моменты, о которых говорили вначале:

1. dtm (depot, store, time, distance) маппинг колонок связывает колонки в файле с колонками в таблице, их имена могут не совпадать
2. CSV HEADER показывает, что заголовок импортировать не нужно
3. Колонки в таблице уже имеют правильные типы данных, конвертация будет выполнена автоматически.

В консоли управления на странице кластера перейдите на вкладку **SQL**. Введите пароль пользователя БД и нажмите кнопку **Подключиться**. Выберите таблицу `dtm`, чтобы убедиться, что добавление данных выполнено правильно.