

Урок 1. Что такое Postgres

Всем привет! В этом модуле мы рассмотрим одну из самых популярных СУБД — Postgres. Разберём, как разворачивать БД с помощью docker, поговорим о синхронном и асинхронном подключении к базе и инструментах для работы с миграциями.

PostgreSQL — это объектно-реляционная система баз данных с открытым исходным кодом. Она использует и расширяет язык SQL в сочетании со многими функциями, которые безопасно хранят и масштабируют самые сложные рабочие нагрузки данных. PostgreSQL пользуется популярностью из-за проверенной архитектуры, надёжности, целостности данных, большого набора функций и расширяемости. PostgreSQL работает на всех основных операционных системах, совместима с ACID и имеет мощные дополнения, например, поддержка географических объектов. Это указано в документации к базе данных.

ACID — это общепринятые требования к транзакционной системе. Их мы обсудим чуть позже.

Первая версия СУБД выпущена в 1989 году. База данных совершенствовалась на протяжении нескольких лет, а в 1993 году, когда вышла версия 4.2, проект был закрыт. Но, несмотря на официальное прекращение существования, открытый код позволил другим разработчикам взяться за его развитие. Они заменили язык запросов POSTQUEL на SQL, который к тому времени стал общепринятым. Проект получил название Postgres95.

К 1996 году стало ясно, что название Postgres95 не выдержит испытание временем. Тогда появилось новое имя — PostgreSQL, которое отражает связь с оригинальным проектом POSTGRES и с переходом на SQL. Именно поэтому PostgreSQL произносится как «постгрес-ку-эль» или просто «постгрес», но только не «постгре».

Разберём плюсы и минусы этого решения и сравним его с ещё одной популярной СУБД Mysql и со Sqlite, с которым вы уже знакомы.

Рассмотрим формулировку объектно-реляционная СУБД. Относительно реляционных СУБД введение объектно-ориентированного подхода даёт разработчикам новые возможности. Например, создание новых типов, операторов и методов доступа, которые определяются пользователем. Мы можем определить кастомные типы данных и использовать их как атрибут таблицы или для вспомогательных операций и методов получения данных. Помимо этого, в PostgreSQL поддерживаются сложные объекты, которые

представляют собой наборы других объектов. Также появилась возможность наследовать объекты базы данных друг от друга.

В официальной документации СУБД описаны преимущества PostgreSQL. Наиболее важным является надёжность. PostgreSQL позволяет настраивать резервирование, то есть сохранять копии данных вне основного места их содержания. Так мы можем получить backup базы и восстановиться на заданный момент времени в прошлом. Для отката БД нужны различные виды репликации — синхронная, асинхронная, каскадная. Репликация — это процесс поддержания двух или более наборов данных в согласованном состоянии. Это важно, если вы хотите создать реплику БД, чтобы использовать её при потере основной базы. Также репликация пригодится, чтобы распределить входные запросы между основной базой и её репликами.

СУБД позволяет работать по защищённому SSL-соединению. SSL обеспечивает безопасный канал связи между двумя машинами или устройствами, которые работают через интернет или внутреннюю сеть. Также СУБД предоставляет большое количество методов аутентификации. Например, аутентификация по паролю, клиентским сертификатам или с помощью внешних сервисов. Для каждого пользователя БД настраивается разграничение доступа к объектам, определяются роли доступа.

PostgreSQL обеспечивает полную поддержку свойств ACID и эффективную изоляцию транзакций.

ACID (Atomicity — атомарность, consistency — консистентность, isolation — изолированность, durability — стойкость) — это стандартный набор свойств, которые гарантируют надёжность транзакции.

Транзакция — набор последовательных операций с базой данных, который представляет собой логическую единицу работы с данными. Этот набор операций можно сравнить с архивом файлов: либо передаётся весь архив, либо ни один файл не передаётся.

Атомарность гарантирует, что каждая транзакция будет выполнена полностью или не будет выполнена совсем. Если при выполнении одного из запросов в транзакции будет допущена ошибка, то отменится вся транзакция.

Консистентность — свойство, которое вытекает из предыдущего. За счёт атомарности транзакций БД остаётся консистентной. До выполнения транзакции и после неё база остаётся согласованной — успешно отработают триггеры, каскадирование, правила и ограничения.

Следующее свойство гарантирует, что все транзакции будут выполняться изолированно. При выполнении одной из них параллельные не должны влиять на её результат. Стандарт SQL определяет четыре уровня изоляции

транзакций. Один из них — Read uncommitted, его ещё называют «грязное» чтение. При нём могут быть считаны добавленные или изменённые данные из другой транзакции, которая впоследствии откатится.

И последнее преимущество — стойкость. Если пользователь получил от БД подтверждение, что транзакция выполнена, то его изменения не отменятся из-за сбоя.

В PostgreSQL используется механизм многоверсионного управления одновременным доступом (MVCC — Multi-Version Concurrency Control). Он позволяет обходиться без блокировок во всех случаях, кроме одновременного изменения одной и той же строки данных в нескольких процессах. При этом читающие транзакции никогда не блокируют пишущие транзакции и наоборот. Это гарантирует одинаковый результат как при одновременном, так и при последовательном выполнении транзакций.

Эта СУБД отличается хорошей масштабируемостью и производительностью. PostgreSQL эффективно использует современную архитектуру многоядерных процессоров — производительность растёт практически линейно с увеличением количества ядер. В версиях, начиная с 9.6, СУБД может параллельно читать данные, соединять их и выполнять агрегации. Это повышает возможности использования аппаратных средств для ускорения операций.

Сюда же можно отнести кроссплатформенность, доступность, поддержку json и геометрических форматов данных, большое сообщество. Проведём небольшой аналитический обзор и сравним три разных СУБД: Sqlite, PostgreSQL и MySQL.

	Sqlite	PostgreSQL	MySQL
Количество типов данных	5	41	34
Система аутентификации, роли доступа	-	+	+
Масштабируемость	-	+	+
Многопроцессорность	-	+	+
SQL — совместимость	+	+	-

Объектно-ориентированность	-	+	-
----------------------------	---	---	---

MySQL не пытается полностью реализовать стандарты SQL, она не является полностью совместимой с SQL. Из-за этого могут возникать проблемы при интеграции с другими РСУБД. У СУБД есть особенности синтаксиса, которые отличаются от стандарта SQL.

В этом видео мы познакомились с объектно-реляционной СУБД PostgreSQL, обсудили её преимущества и сравнили с двумя другими СУБД. В следующем видео мы поднимем docker-контейнер с СУБД, подключимся к базе через интерфейс Pycharm и разберём особенности структуры PostgreSQL.