

# Часть 2. Инструменты для работы с SQL в Python

08.12.2020

# РСУБД

РСУБД - Реляционные системы управления базами данных

Самыми популярными РСУБД сейчас являются:

- **SQLite:** очень мощная встраиваемая РСУБД.
- **MySQL:** самая популярная и часто используемая РСУБД.
- **PostgreSQL:** самая продвинутая и гибкая РСУБД.

# SQLite

SQLite - это встроенная библиотека, которая реализует автономный, берсерверный, нулевой конфигурации, транзакционный механизм СУБД SQL. Это база данных, которая настроена на нуль, что означает, как и другие базы данных, которые вам не нужно настраивать в вашей системе. Занимает очень мало места - около 250 кб.

## Особенности SQLite:

- быстрее, чем MySQL
- НЕ поддерживает соединения типа «клиент-сервер»
- Для локального использования (как следствие предыдущего пункта)
- Факт, что база данных - это единственный файл, делает её легко переносимой.
- В большинстве сопадает со стандартом SQL 92



# SQLite: когда использовать?

Когда стоит использовать SQLite:

- **Встроенные приложения:** все портируемые не предназначенные для масштабирования приложения — например, локальные однопользовательские приложения, мобильные приложения или игры.
- **Система доступа к дисковой памяти:** в большинстве случаев приложения, часто производящие прямые операции чтения/записи на диск, можно перевести на SQLite для повышения производительности.
- **Тестирование:** отлично подойдёт для большинства приложений, частью функционала которых является тестирование бизнес-логики.

Когда не стоит использовать SQLite:

- **Многопользовательские приложения:** если вы работаете над приложением, доступом к БД в котором будут одновременно пользоваться несколько человек, лучше выбрать полнофункциональную РСУБД — например, MySQL.
- **Приложения, записывающие большие объёмы данных:** одним из ограничений SQLite являются операции записи. Эта РСУБД допускает единовременное исполнение лишь одной операции записи.

# SQLite

Неподдерживаются следующие возможности:

- RIGHT и FULL OUTER JOIN. Реализован только LEFT OUTER JOIN.
- Частично реализован ALTER TABLE. Доступны только RENAME TABLE и ADD COLUMN.
- Частичная поддержка триггеров. Доступен только FOR EACH ROW триггеры.
- Запись во VIEWS. В SQLite VIEWS доступны только на чтение. Частично обходится через триггеры.
- В силу реализации базы данных, как единственного файла и отхода от концепции «клиент-сервер», не используются возможности GRANT и REVOKE.
- По умолчанию отключены внешние ключи. Это сделано для обратной совместимости.

# MySQL

MySQL — это самая популярная из всех крупных серверных БД. Разобраться в ней очень просто, да и в сети о ней можно найти большое количество информации. Хотя MySQL и не пытается полностью реализовать SQL-стандарты, она предлагает широкий функционал.

## Особенности MySQL:

- Поддержка соединения типа «клиент-сервер»
- Низкая скорость разработки
- без проблем работает с интерфейсом API



# MySQL: когда использовать?

Когда стоит использовать MySQL:

- **Распределённые операции:** когда вам нужен функционал бóльший, чем может предоставить SQLite, стоит использовать MySQL.
- **Высокая безопасность:** функции безопасности MySQL предоставляют надёжную защиту доступа и использования данных.
- **Веб-сайты и приложения:** большая часть веб-ресурсов вполне может работать с MySQL, несмотря на ограничения. Этот инструмент весьма гибок и прост в обращении, что только на руку в длительной перспективе.
- **Кастомные решения:** если вы работаете над очень специфичным продуктом, MySQL подстроится под ваши потребности благодаря широкому спектру настроек и режимов работы.

Когда не стоит использовать MySQL:

- **SQL-совместимость:** поскольку MySQL не пытается полностью реализовать стандарты SQL, она не является полностью совместимой с SQL. Из-за этого могут возникнуть проблемы при интеграции с другими РСУБД.
- **Конкурентность:** хотя MySQL неплохо справляется с операциями чтения, одновременные операции чтения-записи могут вызвать проблемы.
- **Недостаток функций:** в зависимости от выбора движка MySQL может неоставать некоторых функций.

# Инструменты для работы с SQL в Python

Для работы с SQLite будем использовать библиотеку [sqlite3](#)

Для работы с MySQL - библиотеку [SQLAlchemy](#)



# Заключение

- MySQL и SQLite самые универсальные и гибкие среди остальных СУБД
- Для «локальной» работы используйте SQLite, для удаленного подключения к БД - MySQL