



Краткое описание

На курсе вы познакомитесь с продуктом Tarantool и его функциональными возможностями. Обучение состоит из 14 лекций, 14 практических заданий, выполняющихся на виртуальных машинах VK Tech, и тестирования.

Целевая аудитория

Курс предназначен для специалистов, собирающихся использовать Tarantool в своих проектах.

Предварительные требования

- Базовые знания linux
- Базовые знания программирования
- Базовое понимание работы с сетью
- Базовое понимание принципов работы с данными
- Знакомство с реляционными базами данных

Формат курса: асинхронное обучение

Продолжительность курса: ~ 7 недель

Чистое время прохождения: 30-35 часов

Количество лекций: 14

Версия Tarantool: 2.10

Необходимое оборудование: ноутбук или компьютер, подключенный к интернету, устройство воспроизведения звука, микрофон.



Tarantool

Какие темы разберем:

- Архитектура Tarantool
- Топологии Tarantool
- Программирование на Lua
- Создание приложений
- Tarantool Cartridge
- Tarantool Data Grid

Что вы будете знать после окончания курса:

- Что такое Tarantool и в каких кейсах его использовать
- Как устроено ядро Tarantool
- Как настраивать репликацию и шардинг
- Как описывать бизнес-логику с помощью языка Lua в Tarantool
- Как создать приложение на Tarantool
- Как обслуживать Tarantool, настраивать его и справляться с критическими ситуациями
- Что такое Tarantool Cartridge и Tarantool Data Grid, как с помощью них масштабировать Tarantool

Все презентации, конспекты, практические задания останутся на руках после обучения

Сертификат о прохождении обучения выдается при условии просмотра всех лекций и успешном завершении минимум 5 тестов из 7



Содержание курса

Архитектура Tarantool

Часть первая — База данных

- История развития
- Tarantool сегодня — Open Source и Enterprise
- Устройство ядра
- Сравнения с другими БД
- Кейсы использования
- Функциональность БД

Часть вторая — Функциональность сервера приложений

- Кооперативная многозадачность
 - Файберы
 - Каналы
 - Асинхронная работа с сетью
- Экосистема
 - Lua модули
 - Коннекторы
- Масштабирование

Топологии Tarantool

Часть первая — Репликация

- Виды репликация
- Устройство репликация в Tarantool
- Настройка репликации
- Master-master репликация
- Master-slave репликация
- Параметры и свойства репликации

Часть вторая — Шардинг

- Шардирование
- vshard, виртуальное шардирование в Tarantool
- Ребалансировка данных
- Масштабирование
- Балансировка запросов
- Мониторинг
- Расчет ресурсов

Программирование на Lua

Часть первая — Основные возможности

- Базовый синтаксис
- Типизация
- Итераторы
 - Проектирование итератора
- Метапрограммирование
 - Метатаблицы
 - Перегрузка операторов
- Ошибки и отладка
- Изоляция

Часть вторая — luafun, FFI

- Luafun
- LuaJIT
- Собрщик мусора
- Доступ к библиотекам - FFI
- Разработка приложений
- Библиотеки

Создание приложений

Часть первая — Создание приложения

- Создание очереди сообщений
- Работа с luarocks
- Шаблон приложения
- Tarantoolctl
- Работа с конфигурационным файлом
- Модуль приложения

Часть вторая — Метрики и тестирование

- Фоновая обработка отложенных задач
- Сбор статистики
- Триггеры
- Горячая перезагрузка кода
- Тестирование

Обслуживание и диагностика

Часть первая — Настройка

- Интерфейс командной строки
- Опции командной строки
- Расширенные опции `box.cfg`
- Диагностические функции
- Резервное копирование

Часть вторая — Мониторинг

- Мониторинг
 - Метрики по умолчанию
 - Важные маркеры состояния
- Критические ситуации
- Пересоздание реплики
- Переключение мастера
- Обновление Tarantool
- Расчет ресурсов
- Тюнинг производительности
- Prometheus
- Grafana

Tarantool Cartridge

Часть первая — Архитектура и мониторинг

- Общие сведения — история развития и сценарии использования
- Разработка кластерного приложения
- Архитектура Cartridge
- Мониторинг

Часть вторая — Тестирование, деплой и эксплуатация

- Тестирование
 - Пример кластерных тестов
 - Статический анализ кода
 - Нагрузочное тестирование
- Деплой
 - Упаковка приложения
 - Ansible
 - Установка плагинов
 - Запуск ролей
 - Описание топологии кластера
- Эксплуатация



Tarantool Data Grid

Часть первая — Хранилище данных

- Обзор Tarantool Data Grid
- Задание модели данных — Avroschema
- Доступ к данным — GraphQL
- Работа с историческими данными
- Конкурентный доступ — оптимистические блокировки
- Лимитирование запроса

Часть вторая — Пользовательские скрипты обработки данных

- Коннекторы
- Ремонтная очередь
- Хранимые процедуры
- Фоновые задачи
- API для хранимых процедур
- Вспомогательные системы