

Домашнее задание к занятию

"Базы данных, их типы"

Акимов Александр

Инструкция по выполнению домашнего задания

1. Сделайте fork [репозитория с шаблоном решения](#) к себе в Github и переименуйте его по названию или номеру занятия, например, <https://github.com/имя-вашего-репозитория/gitlab-hw> или <https://github.com/имя-вашего-репозитория/8-03-hw>).
2. Выполните клонирование этого репозитория к себе на ПК с помощью команды `git clone`.
3. Выполните домашнее задание и заполните у себя локально этот файл README.md:
 - впишите вверху название занятия и ваши фамилию и имя;
 - в каждом задании добавьте решение в требуемом виде: текст/код/скриншоты/ссылка;
 - для корректного добавления скриншотов воспользуйтесь инструкцией [«Как вставить скриншот в шаблон с решением»](#);
 - при оформлении используйте возможности языка разметки md. Коротко об этом можно посмотреть в [инструкции по Markdown](#).
4. После завершения работы над домашним заданием сделайте коммит (`git commit -m "comment"`) и отправьте его на Github (`git push origin`).
5. Для проверки домашнего задания преподавателем в личном кабинете прикрепите и отправьте ссылку на решение в виде md-файла в вашем Github.
6. Любые вопросы задавайте в чате учебной группы и/или в разделе «Вопросы по заданию» в личном кабинете.

Желаем успехов в выполнении домашнего задания.

Задание 1. СУБД

Кейс

Крупная строительная компания, которая также занимается проектированием и девелопментом, решила создать правильную архитектуру для работы с данными. Ниже представлены задачи, которые необходимо решить для каждой предметной области.

Какие типы СУБД, на ваш взгляд, лучше всего подойдут для решения этих задач и почему?

1.1. Бюджетирование проектов с дальнейшим формированием финансовых аналитических отчётов и прогнозированию рисков. СУБД должна гарантировать целостность и чёткую структуру данных.

ОТВЕТ:

Выбор - реляционная СУБД для работы с данными в табличной форме с информацией по объекту для составления отчетов и прогнозирования на основе накопления данных.

1.1.* Хеширование стало занимать длительно время, какое API можно использовать для ускорения работы?

1.2. Под каждый девелоперский проект создаётся отдельный лендинг, и все данные по лидам стекаются в CRM к маркетологам и менеджерам по продажам. Какой тип СУБД лучше использовать для лендингов и для CRM? СУБД должны быть гибкими и быстрыми.

ОТВЕТ:

Для CRM - реляционная СУБД (неизменяемые данные в табличной форме);
Для Лендингов - тоже реляционная СУБД.

1.2.* Можно ли эту задачу закрыть одной СУБД? И если да, то какой именно СУБД и какой реализацией?

1.3. Отдел контроля качества решил создать базу по корпоративным нормам и правилам, обучающему материалу и так далее, сформированную согласно структуре компании. СУБД должна иметь простую и понятную структуру.

ОТВЕТ:

СУБД типа ключ-значение. Запросил документ по ключу и получил необходимую инфу.

1.3.* Можно ли под эту задачу использовать уже существующую СУБД из задач выше и если да, то как лучше это реализовать?

1.4. Департамент логистики нуждается в решении задач по быстрому формированию маршрутов доставки материалов по объектам и распределению курьеров по маршрутам с доставкой документов. СУБД должна уметь быстро работать со связями.

ОТВЕТ:

Графовые СУБД - специфичный тип, предназначены для работы с графами, с их узлами, свойствами, и произвольными отношениями между узлами.

1.4.* Можно ли к этой СУБД подключить отдел закупок или для них лучше сформировать свою СУБД в связке с СУБД логистов?

1.5.* Можно ли все перечисленные выше задачи решить, используя одну СУБД? Если да, то какую именно?

Приведите ответ в свободной форме.

Задание 2. Транзакции

2.1. Пользователь пополняет баланс счёта телефона, распишите пошагово, какие действия должны произойти для того, чтобы транзакция завершилась успешно. Ориентируйтесь на шесть действий.

ОТВЕТ:

1. Открытие транзакции. Открытие соединения с БД Банка;
2. Обращение к счету в банке и проверка наличия необходимых к переводу средств;
3. Блокировка для списания со счета в банке необходимой суммы;
4. Открытие соединения с БД Оператора сотовой связи;
5. Зачисление на лицевой счет суммы, равной списанной суммы со счета в банке;
6. Коммит изменений. Закрытие транзакции.

2.1.* Какие действия должны произойти, если пополнение счёта телефона происходило бы через автоплатёж?

ОТВЕТ:

Те же самые действия. Автоплатеж активирует выполнение транзакции в определенное время без участия пользователя.

Приведите ответ в свободной форме.

Задание 3. SQL vs NoSQL

3.1. Напишите пять преимуществ SQL-систем по отношению к NoSQL.

ОТВЕТ:

1. SQL-базы данных имеют лучшую поддержку транзакционности, что позволяет автоматически откатывать изменения при обнаружении проблемных транзакций и, таким образом, нивелировать возможные проблемы безопасности;
2. В SQL можно легко установить ограничения на доступ к данным для разных пользователей, а также применять различные аутентификационные механизмы для обеспечения безопасности данных;
3. Кэширование в SQL-базах данных позволяет улучшить производительность и более точно отслеживать изменения, что может помочь в оптимизации обработки запросов. SQL-базы данных имеют большую структурированность и лучшее соответствие этикету ACID;
4. SQL имеет мощный язык запросов, который позволяет обрабатывать сложные запросы;
5. SQL-базы данных имеют жёстко определённый формат хранения данных, что делает их наиболее подходящим выбором для представления сложных связанных данных.

3.1.* Какие, на ваш взгляд, преимущества у NewSQL систем перед SQL и NoSQL.

Приведите ответ в свободной форме.

ОТВЕТ:

NewSQL (англ. новый SQL) — класс реляционных СУБД, возникших на рубеже 2000-х и 2010-х годов, стремящихся совместить в себе преимущества NoSQL и транзакционные требования классических систем управления базами данных. Термин предложен в 2011 году аналитиком 451 Group Мэтью Аслетом, отметившим потребность в данных таких системах у организаций, работающих с критическими данными (например, финансового сектора), которым требуются масштабируемые решения, в то время как решения NoSQL не могли

обеспечить транзакционные механизмы и не отвечали требованиям надёжности данных.

Судя из описания из Вики, это попытка получения лучшего от SQL (транзакционные механизмы) и NoSQL (масштабируемость).

Задание 4. Кластеры

Необходимо производить большое количество вычислений при работе с огромным количеством данных, под эту задачу выделено 1000 машин.

На основе какого критерия будете выбирать тип СУБД и какая модель *распределённых вычислений* здесь справится лучше всего и почему?

Приведите ответ в свободной форме.

ОТВЕТ:

Масштабируемость, надёжность, синхронизованность, ребалансировка!

Популярность языка SQL и развитие “железа” породили новое движение — распределённые базы данных с языком запросов SQL. Среди них выделяется Google Spanner, которая гарантирует linearizability — глобальный порядок записи всех транзакций. Чтобы решить такую задачу в масштабах планеты, нужно синхронизировать время на серверах БД по всему миру. Компания Google использует для этого атомные часы, а для резерва — GPS-приёмники.

Однако для простых смертных атомные часы пока остаются роскошью, поэтому авторы Spanner построили аналогичную БД с несколько меньшими гарантиями на порядок транзакций, но достаточными для большинства приложений. Эта БД называется CockroachDB (от англ. “таракан”) и своим названием олицетворяет живучесть кластера при сбоях железа или связях между дата-центрами. CockroachDB предоставляет полноценные распределённые транзакции и автоматическую ребалансировку кластера при потере ноды, что, вкупе с привычным языком запросов SQL, выгодно отличает ее от Cassandra. Из недостатков стоит отметить отсутствие полнотекстовых индексов и сравнительную молодость решения.