1. **Поясните понятие GraphQL.**

GraphQL — это язык запросов, используемый клиентскими приложениями для работы с данными.

GraphQL - это язык запросов и манипулирования данными с открытым исходным кодом для API, а также среда выполнения для выполнения запросов с существующими данными.

1. **Поясните понятие схема GraphQL.**

Схема содержит в себе описания всех типов, полей и методов получения данных. Все типы в рамках GraphQL-схемы должны иметь уникальные имена. Не должно быть двух разных типов с одним именем.

1. **Расшифруйте аббревиатуру SDL GraphQL.**

Для создания схем в GraphQL используется собственный язык Schema Definition Language (SDL). SDL обладает интуитивно понятным синтаксисом и универсален для любой используемой технологии.

1. **Поясните понятие resolver GraphQL.**

**Resolver или распознаватель** — функция, которая возвращает данные для определённого поля. Resolver'ы возвращают данные того типа, который определён в схеме. Распознаватели могут быть асинхронными. С их помощью можно получать данные из REST API, базы данных или другого источника.

1. **Поясните понятие query GraphQL.**

С помощью запросов GraphQL получает необходимые данные с сервера. Тип запроса Query в GraphQL — аналог GET в REST. ... Query описывает данные, которые необходимо получить с сервера.

1. **Поясните понятие mutation GraphQL**

В GraphQL изменения — способ модифицировать данные на сервере и получить обработанную информацию. Этот процесс можно рассматривать как аналогичный концепции CUD (Create, Update, Delete) в стандарте REST.

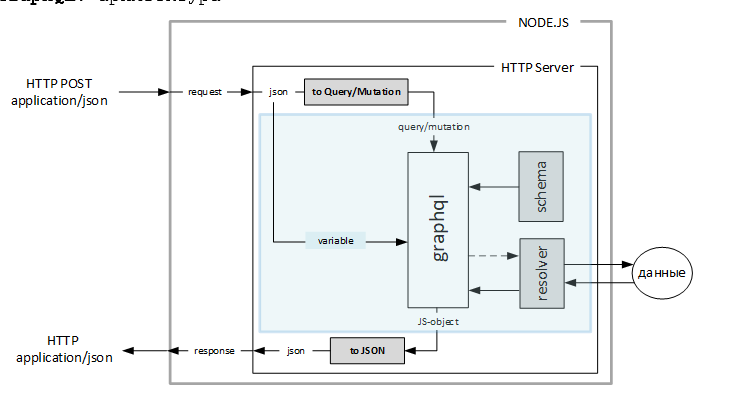
1. **Поясните понятие subscription GraphQL.**

**Подписки GraphQL** - это способ передачи данных с сервера клиентам, которые выбирают прослушивание сообщений с сервера в режиме реального времени.

1. **Поясните понятие context GraphQL.**

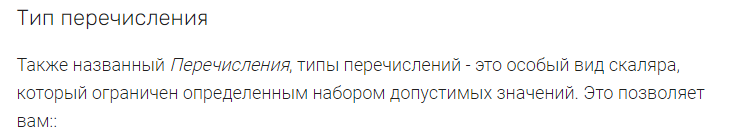
Контекст представляет из себя объект, хранящий информацию о подключении пользователя к базе данных.

1. **Поясните схему работы модуля graphql**

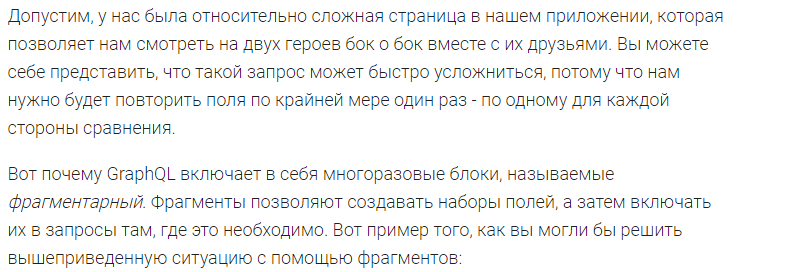


1. Поясните следующие компоненты **subscription, interface, enum, fragment, union схемы GraphQL.**

Как и многие системы типов, GraphQL поддерживает интерфейсы. Интерфейс-это абстрактный тип, который включает в себя определенный набор полей, которые тип должен включать в себя для реализации интерфейса.

****

**fragment**

****

**Union**

Типы объединения очень похожи на интерфейсы, но они не задают никаких общих полей между типами.