1. Поясните понятие **GraphQL**.

**синтаксис, который описывает как запрашивать данные**, и, в основном, используется клиентом для загрузки данных с сервера.

1. Поясните понятие схема **GraphQL**.

это то, что позволяет организовывать создание, чтение, обновление и удаление данных в вашем приложении (то есть — перед нами четыре базовые функции, используемые при работе с хранилищами данных, которые обычно обозначают акронимом CRUD — create, read, update, delete).

1. Расшифруйте аббревиатуру **SDL GraphQL**.

Для создания схем в GraphQL используется собственный язык Schema Definition Language (SDL).

1. Поясните понятие **resolver** **GraphQL**.

Действие при запросе

1. Поясните понятие **query GraphQL**.

Выборка данных

1. Поясните понятие **mutation** **GraphQL**.

Изменение бд

1. Поясните понятие **subscription GraphQL**.

Вызов при изменении в бд

1. Поясните понятие **context GraphQL**.

Общий объект для распознавания resolve

1. Поясните схему работы модуля **graphql**

Единичный запрос данных, мутация или подписка, которые интерпретирует исполняемый модуль GraphQL.

1. Область где происходит описание типов

В query

1. Поясните следующие компоненты **subscription,**

**Interface -** способом построения и использования схем GraphQL посредством использования абстрактных типов

**Enum -** специальным типом, который мы можем использовать для перечисления всех возможных значений в поле. Используя Enums, мы добавляем другой вид проверки в существующую схему GraphQL. Указанные значения типа Enum являются единственно возможными вариантами, которые принимаются.

**fragment** позволяют создавать наборы полей, а затем включать их в запросы, где это необходимо. Вот пример того, как вы можете решить вышеуказанную ситуацию, используя фрагменты

**union** - указывает на то, что поле может возвращать более одного типа объекта