1. **Поясните аббревиатуру MIME, какой организаций поддерживается допустимые MIME, в каких компонентах запросов и ответов используется MIME.**

1) MIME – Multipurpose Internet Mail Extensions — многоцелевые расширения интернет-почты – стандарт или спецификация для кодирования и форматирования информации для пересылки её по Интернету. В формат входят как текстовые данные любых кодировок, так и файлы множества форматов (картинки, видео, музыка, JSON, XML).

2) За MIME ответственна IANA (Internet Assigned Numbers Authority), все описано в RFC (конкретно в версии 6838).

3) MIME используется в заголовках, например Content-Type (со стороны сервера: то, какие данные пересылаются в ответе, чтобы клиенту было понятно, как их обрабатывать) и Accept (со стороны браузера, чтобы указать, в каком формате они предпочитают принимать данные).

Заголовок Content-Type пишется следующим образом: тип/подтип, например text/html. Тип представляет общую категорию типа данных: video, image, audio, application (неинтерпретируемый двоичный код какого-то приложения).

Подтип же определяет конкретный тип данных, определенный MIME: html, png, plain, xml, json.

И в типе, и в подтипе можно указать \* — то есть принимаются все форматы типа или подтипа. Например, \*/\* значит что браузеру ваще похуй принимаю всё что скормят.

**Основные типы и их подтипы:**

1. text – текстовые данные: html, plain, css.
2. image – графические типы: jpeg, png, gif, bmp.
3. audio – звуковые типы: mpeg (это mp3), wav, midi.
4. video – видео типы: avi, mpeg (это значит mpeg), mp4.
5. application – данные какого-то приложения: msword, pdf, zip, rtf.
6. model – данные 3D-моделей: vml, 3mf.
7. font – шрифты: ttf, otf, woff.
8. example – для написания примеров. Используется *только* в документации или примерах кода.
9. message - сообщение, включающее в себя другое сообщение.
10. multipart – данные состоят из нескольких частей разных типов: mixed, alternative (используется в e-mail).
11. Перечислите теги HTML, интерпретация которых приводит к HTTP-запросам.

хуй знает пока что завтра спрошу у кого нибудь

1. **Перечислите способы выполнения HTTP-запросов из JS-сценария.**

fetch, XmlHttpRequest.

1. **Поясните понятие «параметризованный модуль».**

Это модуль, при импорте которого можно передать в него некоторые параметры. Два варианта, как это сделать:

1) Экспортировать из модуля одну функцию, которая будет иметь параметр, и экспортировать её следующим образом:

module.exports = (parm) => new ModFunc(parm);

2) Реализовать в модуле конструктор через prototype и экспортировать не *экземпляр*, а сам *конструктор*:

Func.prototype.verifyRoles = parm = > {}

module.exports = Func;