Учреждение образования

«БЕЛОРУССКИЙ ГОСУДАРСТВЕННЫЙ ТЕХНОЛОГИЧЕСКИЙ УНИВЕРСИТЕТ»

Исследование протокола HTTP

Студент: Трубач Д. С.

ФИТ 3 курс 5 группа

Преподаватель: Бернацкий П. В.

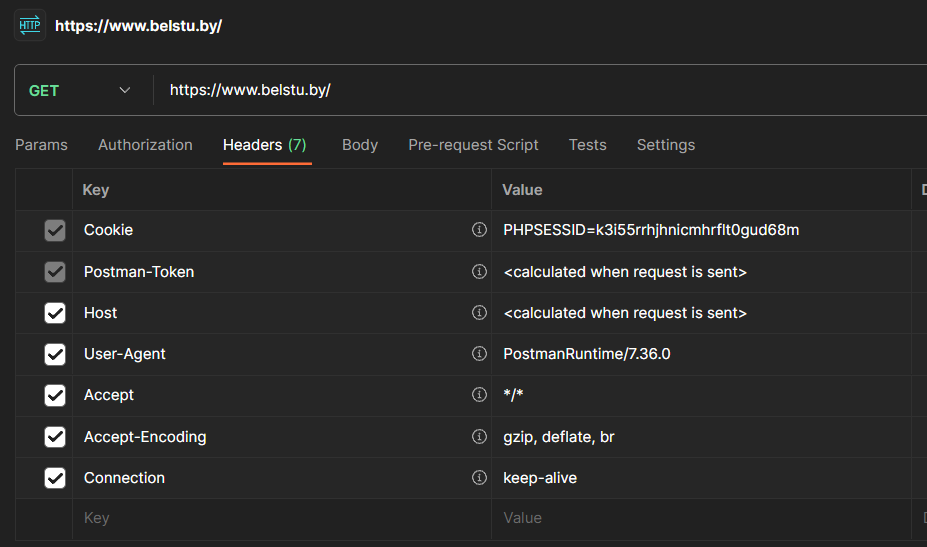
Минск 2024

**Задание 1**

3. POSTMAN: выполните GET-запрос к сайту БГТУ: https://www.belstu.by/

База, вроде было легко

4. Определите заголовки запроса. Поясните их назначение.



**General заголовки:**

Connection – запрашивает открытый или закрытый тип соединения (Кроме keep-alive еще может быть close (закрыть соединение), update (переключение на другой протокол)).

**Request заголовки:**

Cookie – содержит сохраненную информацию о куках. Это позволяет серверу узнать, что несколько запросов пришли от одного и того же браузера, что полезно, например, для аутентификации пользователя.

Postman-Token – пользовательский заголовок (custom), содержит случайный UUID для каждого запроса (помогает в отладке разработчикам). Значение этого заголовка - это уникальный идентификатор, который генерируется для каждого запроса.

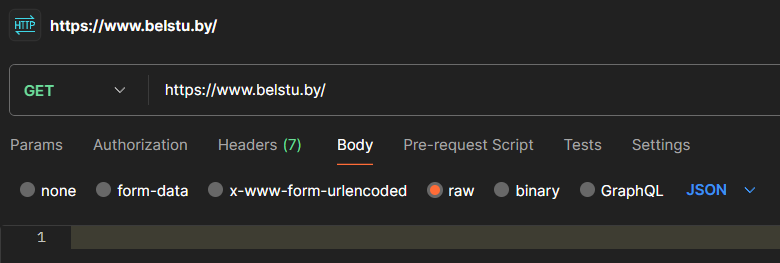
Host – указывает хост и номер порта.

User-Agent – содержит информацию о клиенте, который отправляет запрос. Это может включать в себя детали о браузере, операционной системе и другую информацию, которая может быть полезна для сервера при формировании ответа.

Accept – список допустимых форматов ресурсов, которые клиент может принять (Если \*/\* - то клиент, у нас это браузер, может принять любой тип контента (html, json, png …).

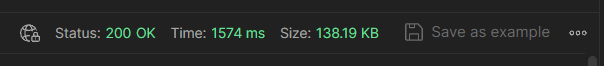
Accept-Encoding – список допустимых кодировок понятные клиенту (Если gzip, deflate, br – то это просто алгоритмы сжатия, инфа по каждому думаю излишне).

5. Определите содержимое тела запроса. Поясните.

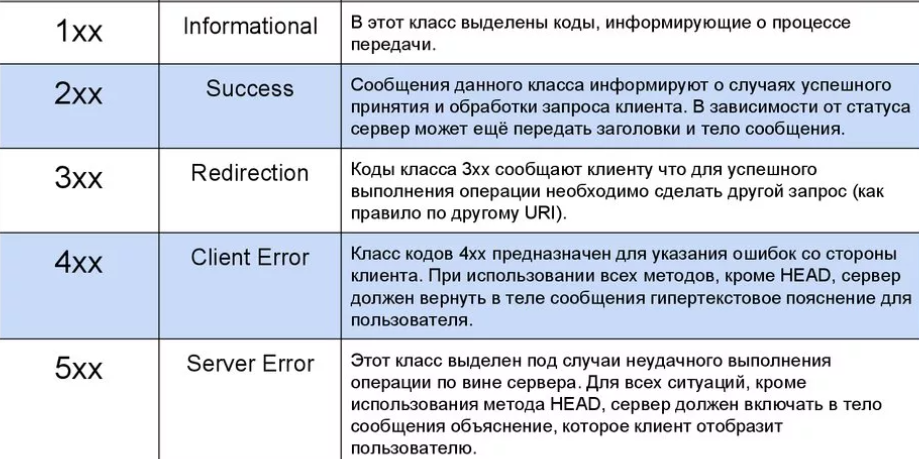


Тело пустое

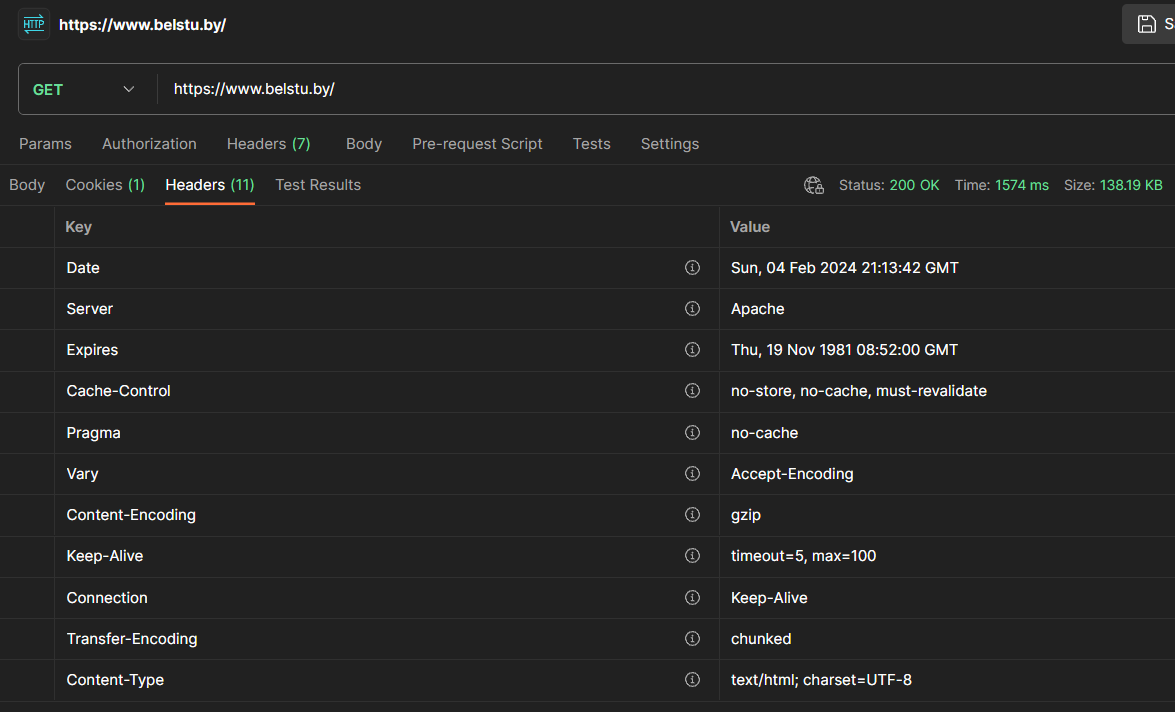
6. Определите статус ответа. Поясните значение.



200 OK – запрос успешно выполнен.



7. Определите заголовки ответа. Поясните их назначение.



**General заголовки:**

Date – дата и время генерации отклика.

Cache-Control – используется для задания инструкций кеширования как для запросов, так и для ответов (Может содержать: no-store (запрещает кеширование), no-cache (необходимость отправить запрос на сервер для валидации ресурса перед использованием закешированных данных), must-revalidate (кешированный ответ должен быть проверен на актуальность перед использованием)).

Pragma – устаревший заголовок, который используется для обратной совместимости с HTTP/1.0, означает то же, что и Cache-Control: no-cache (указывает на необходимость отправить запрос на сервер для валидации ресурса перед использованием закешированных данных).

Keep-Alive – пользовательский заголовок (custom), работающий вместе с Connection. Указывает сколько по времени или количество запросов проживет соединение с сервером. (Не поддерживается в HTTP/2 и HTTP/3)

Connection – определяет останется ли соединение открытым (keep-alive) или закрытым (close) после завершения текущей транзакции. (Не поддерживается в HTTP/2 и HTTP/3)

Transfer-Encoding – определяет форму кодирования, используемую для безопасной передачи payload body пользователю.

**Response заголовки:**

Server – содержит информацию как (с помощью чего) сервер обрабатывает запросы (В данном случае, у меня на сервере работает веб-сервер Apache, еще может быть nginx, GWS, Microsoft IIS).

Set-Cookie – используется для отправки куки от сервера к пользователю (при повторном запросе заголовок отсутствует).

Expires – определяет дату и время, когда клиенту следует считать копию ресурса устаревшей. Прошедшая или невалидная дата, например 0, обозначает, что ресурс уже устарел.

Vary – заголовок используется для отображения запрашиваемого содержимого страницы при ее кэшировании. Наиболее актуально использование данного заголовка при наличии нескольких версий динамических или статических веб-страниц (мобильных, десктопных, мультиязычных версий) под одним URL.

Процесс кэширования призван сократить время и ресурсы при передаче данных от браузера к первоначальному серверу.

Vary позволяет отдавать на запрос одного и того же URL разные ответы в зависимости от различных факторов и условий:

* Версии сайта (для десктопа или мобильных платформ);
* Географии или языка пользователя;
* Статуса пользователя (регистрация на сайте, оплата подписки, наличие товаров в корзине) и др.

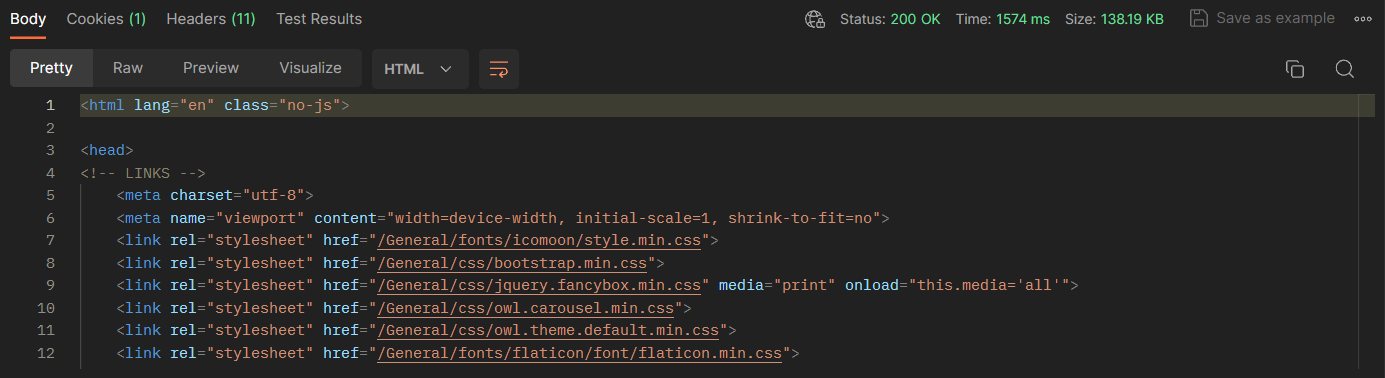
(Если установлен Accept-Encoding, то сервер может отправлять разные версии ресурса в зависимости от того, какие кодировки сжатия поддерживаются клиентом, может еще быть User-Agent (разные версии в зависимости от, к примеру, браузера), Accept-Language (разные версии в зависимости от языка пользователя), Origin (в зависимости от источника запроса).

**Entity заголовки:**

Content-Encoding – определяет способ кодирования содержимого сущности при передачи.

Content-Type – указывает на тип мультимедийного ресурса, передаваемого в теле сообщения

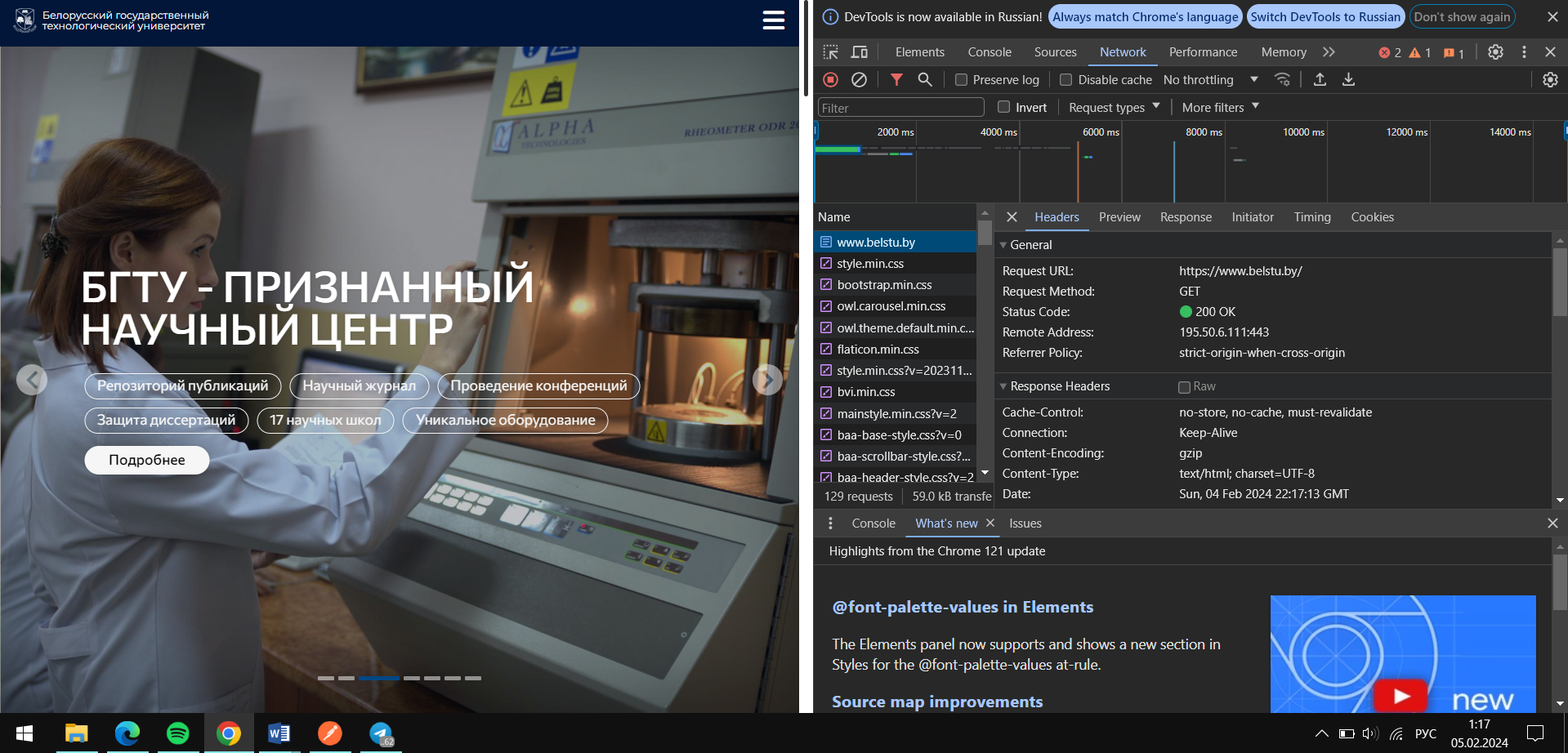
8. Определите содержимое тела ответа. Поясните.



В теле ответа будет HTML-код сайта belstu.by

**Задание 2**

1. **WEB BROWSER**: выполните GET-запрос к сайту БГТУ: <https://www.belstu.by/>
2. Используйте «**Инструмент разработчи**ка» браузера



1. Определите заголовки запроса. Определите содержимое тела запроса.

**Request заголовки:**

Accept-Language – список языков поддерживаемых клиентом.

Sec-Ch-Ua – отправляет информацию о браузере.

Sec-Ch-Ua-Mobile – указывает браузер мобильной версии или нет.

Sec-Ch-Ua-Platform – указывает ОС клиента.

Sec-Fetch-Dest – указывает, как и где будут использоваться полученные данные (тип).

Sec-Fetch-Mode – указывает режим запроса (в общих чертах это позволяет серверу различать запросы, исходящие от пользователя, перемещающегося между HTML-страницами, и запросы на загрузку изображений и других ресурсов).

Sec-Fetch-Site – указывает на связь между источником инициатора запроса и источником запрошенного ресурса.

Sec-Fetch-User – указывает, что запрос инициирован пользователем.

Upgrade-Insecure-Requests – указывает, что браузер желает перенаправить запрос с HTTP на HTTPS.

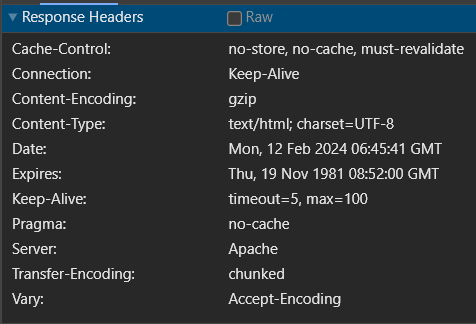
1. Определите содержимое тела запроса.

Тело пустое

1. Определите статус ответа.



1. Определите заголовки ответа.



1. Определите содержимое тела ответа.

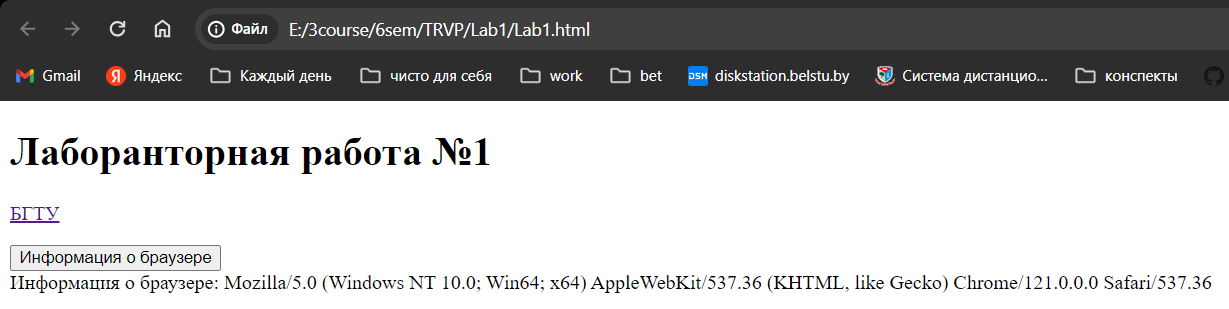
Ну по сути тоже результат наша веб страница

**Задание 3**

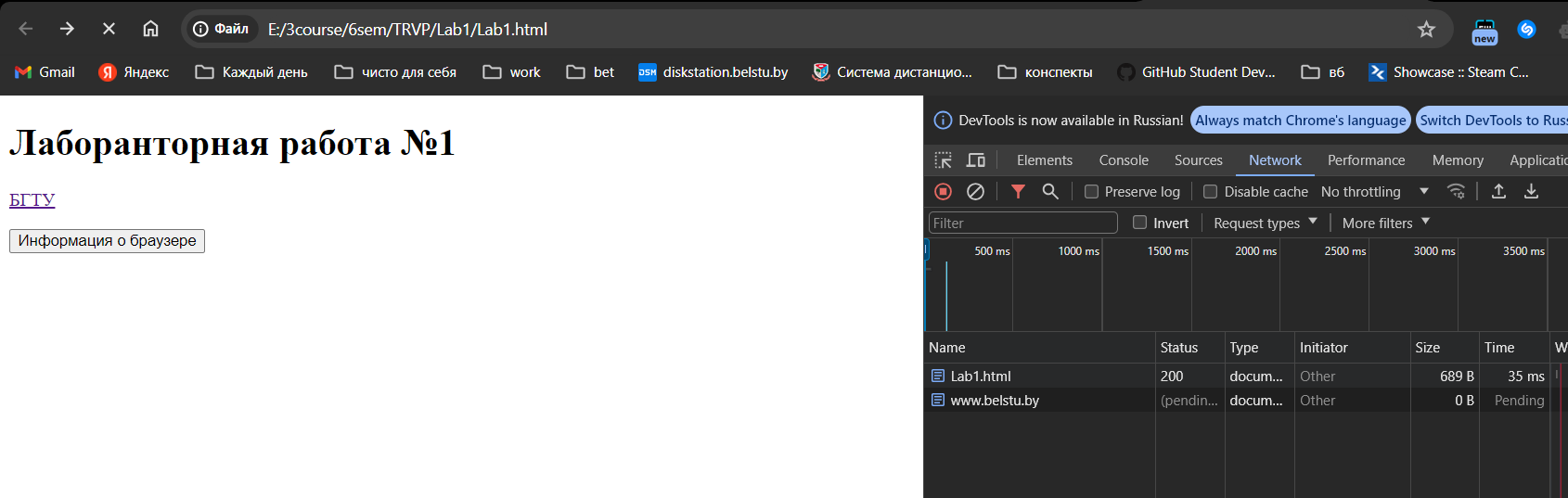
1. Разработайте html-страницу с помощью Блокнота, которая в браузере отображается следующим образом. Страница по ссылке БГТУ должна переходить на сайт БГТУ, при нажатии кнопки высвечивается информация о браузере, а в консоль разработчика должно выводиться Hello World.

Код:

|  |
| --- |
| <!DOCTYPE html>  <html>  <head>  <title>Lab1</title>  </head>  <body>  <h1>Лабораторная работа №1</h1>  <p><a href="https://www.belstu.by/">БГТУ</a></p>  <button onclick="showBrowserInfo()">Информация о браузере</button>  </body>  <script>  function showBrowserInfo() {  var info = document.createTextNode("Информация о браузере: " + navigator.userAgent);  var appCodeName = document.createTextNode("appCodeName: " + navigator.appCodeName);  var appName = document.createTextNode("appName: " + navigator.appName);  var appVersion = document.createTextNode("appVersion: " + navigator.appVersion);  var newLine = document.createElement('br');  document.body.appendChild(newLine);  document.body.appendChild(info);  document.body.appendChild(newLine.cloneNode());  document.body.appendChild(appCodeName);  document.body.appendChild(newLine.cloneNode());  document.body.appendChild(appName);  document.body.appendChild(newLine.cloneNode());  document.body.appendChild(appVersion);  console.log("Hello World");  }  </script>  </html> |









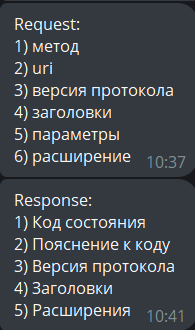
Изменилось только значение заголовка Sec-Fetch-Site. Было none (означает, что операция была инициирована пользователем, например: ввод URL-адреса в адресную строку, открытие закладки или перетаскивание файла в окно браузера), стало cross-site (инициатор запроса и сервер, на котором размещен ресурс, имеют разные сайты).

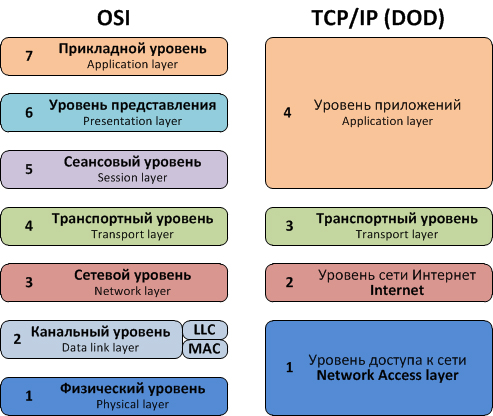
**Задание 4. Ответьте на вопросы**

1. **В каком документе описывает протокол HTTP?**

RFC 7230-7235(Request for Comments)

1. **Какого уровня HTTP-протокол в моделях OSI/ISO, TCP/IP?**



На прикладном уровне модели OSI/ISO и на уровне приложений в модели TCP/IP



1. **Что такое web-приложение?**

Веб-приложение – приложение, которое работает по протоколу HTTP и имеет клиент-серверную архитектуру.

1. **Какие порты использует HTTP-протокол?**

80 – HTTP

443 - HTTPS

1. **Что такое RFC?**

RFC (Request for Comments – заявка на отзывы = тема для обсуждения) – документ из серии пронумерованных информационных документов Интернета, содержащих технические спецификации и стандарты, широко применяемые во всемирной сети. Публикацией занимается IETF (Internet Engineering Task Force – рабочая группа проектирования Internet)

1. **Почему HTTP-протокол называют ассиметричным?**

HTTP-протокол называют ассиметричным, потому что:

По Смелову: сообщения, которые идут от клиента к серверу, отличаются от сообщений, которые идут от сервера к клиенту

Как сказал Бернацкий: роли клиента и сервера РАЗЛИЧНЫ!!!

1. **Что такое MIME?**

Multipurpose Internet Mail Extensions – многоцелевые расширения Internet-почты. Используется и как стандарт кодирования Internet-сообщений. IANA (Internet Assigned Numbers Authority – Администрация адресного пространства Internet) регистрирует типы данных MIME. Стандарт указывающий характер и формат документа, файла или набора байтов.