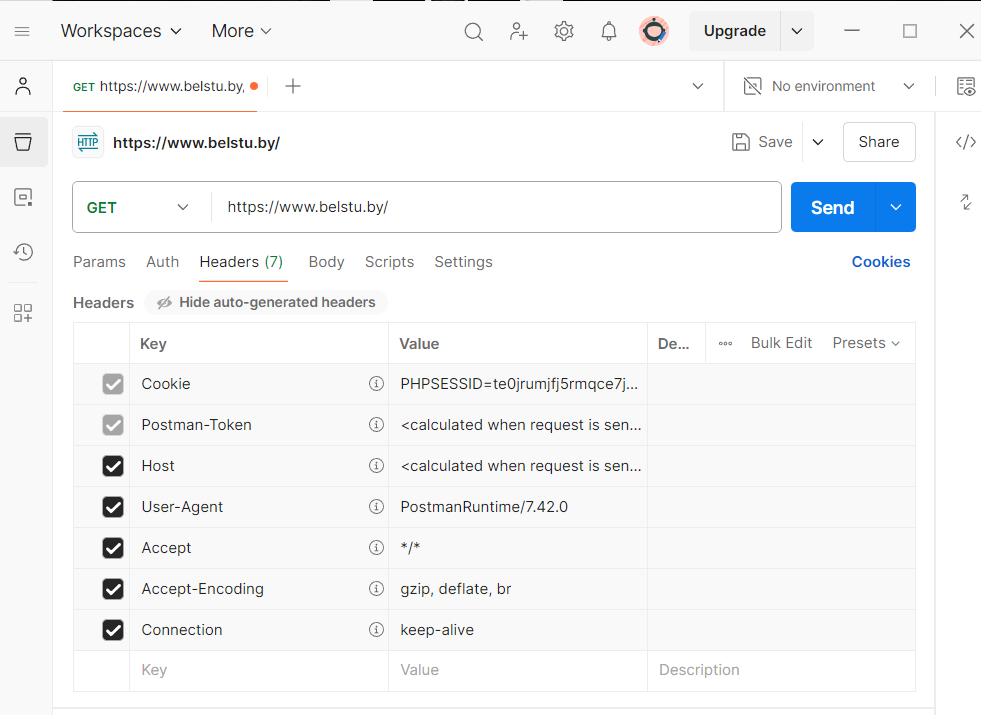
**ТРWП**

**ПОИТ-3**

**Лабораторная 1. Исследование протокола HTTP (2 часа)**

**Задание1**

1. **POSTMAN:** выполните GET-запрос к сайту БГТУ: <https://www.belstu.by/>



1. Определите заголовки запроса. Поясните их назначение.

**Cookie** – содержит данные, которые были сохранены на стороне клиента.

**Postman-Token** – это уникальный идентификатор, который генерируется для каждого запроса, позволяет избежать повторной аутентификации.

**Host** – указывает хост (адрес устройства в сети) и номер порта (что именно обрабатывает входящие соединения).

**User-Agent** – содержит информацию о клиенте, который отправляет запрос.

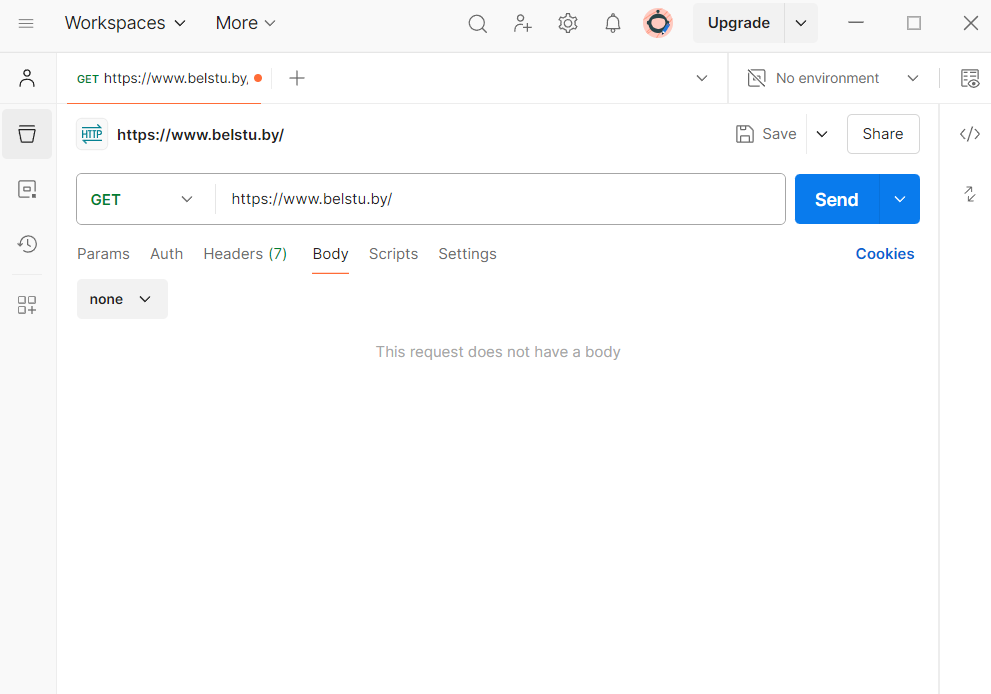
**Accept** – список допустимых форматов ресурсов, которые клиент может принять (\*/\* - может принять любой тип).

**Accept-Encoding** – список допустимых кодировок понятные клиенту (gzip, deflate, br – то это просто алгоритмы сжатия).

**Connection** – запрашивает открытый или закрытый тип соединения.

1. Определите содержимое тела запроса. Поясните.

Там ничего нет 🥀 🥀 🥀 🥀 🥀 🥀 🥀 🥀 🥀 🥀 🥀 🥀 🥀 🥀 🥀 🥀 🥀 🥀



1. Определите статус ответа. Поясните значение.



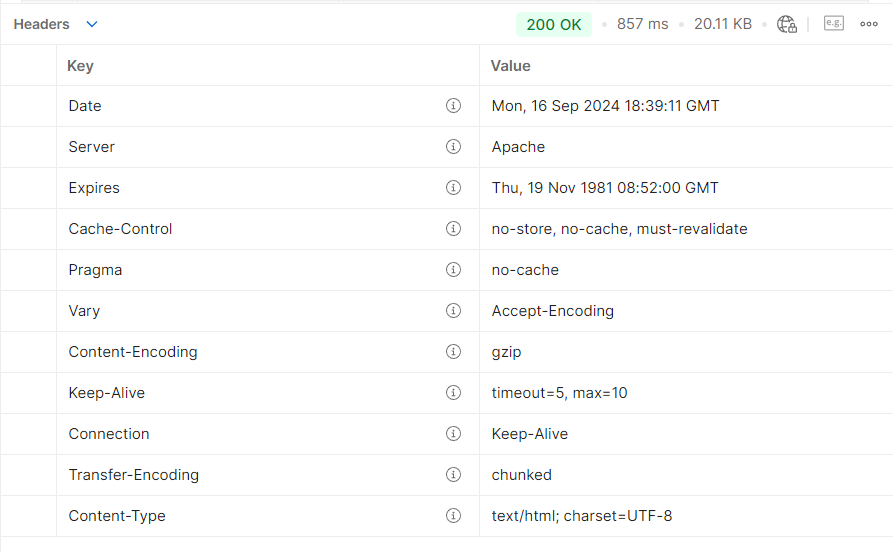
Код 200 означает, что сервер успешно обработал запрос.

"OK" подтверждает, что запрос выполнен успешно.

857ms: Это время от отправки запроса до получения ответа от сервера.

2 0.11KB: Объем данных в ответе.

1. Определите заголовки ответа. Поясните их назначение.



**Date** – дата и время генерации отклика.

**Server** – как сервер обрабатывает запросы.

**Expires** – определяет дату и время, когда клиенту следует считать копию ресурса устаревшей.

**Cache-Control** – для задания инструкций кеширования запросов и для ответов.

**Pragma** – указывает на необходимость отправить запрос на сервер для валидации ресурса перед использованием закешированных данных.

**Vary** – заголовок используется для отображения запрашиваемого содержимого страницы при ее кэшировании.

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

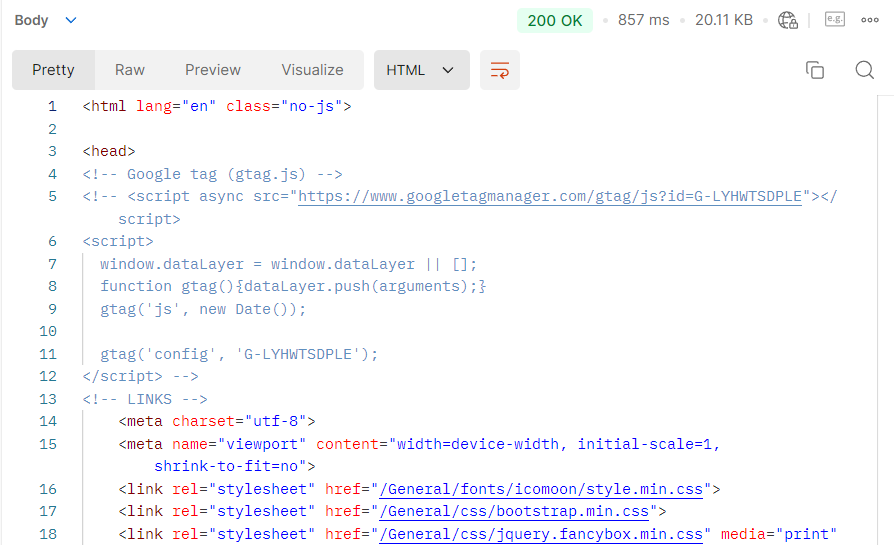
**Keep-Alive** – указывает сколько по времени или количество запросов проживет соединение с сервером.

**Connection** – определяет останется соединение открытым или закрытым

**Transfer-Encoding** – определяет форму кодирования, используемую для безопасной передачи

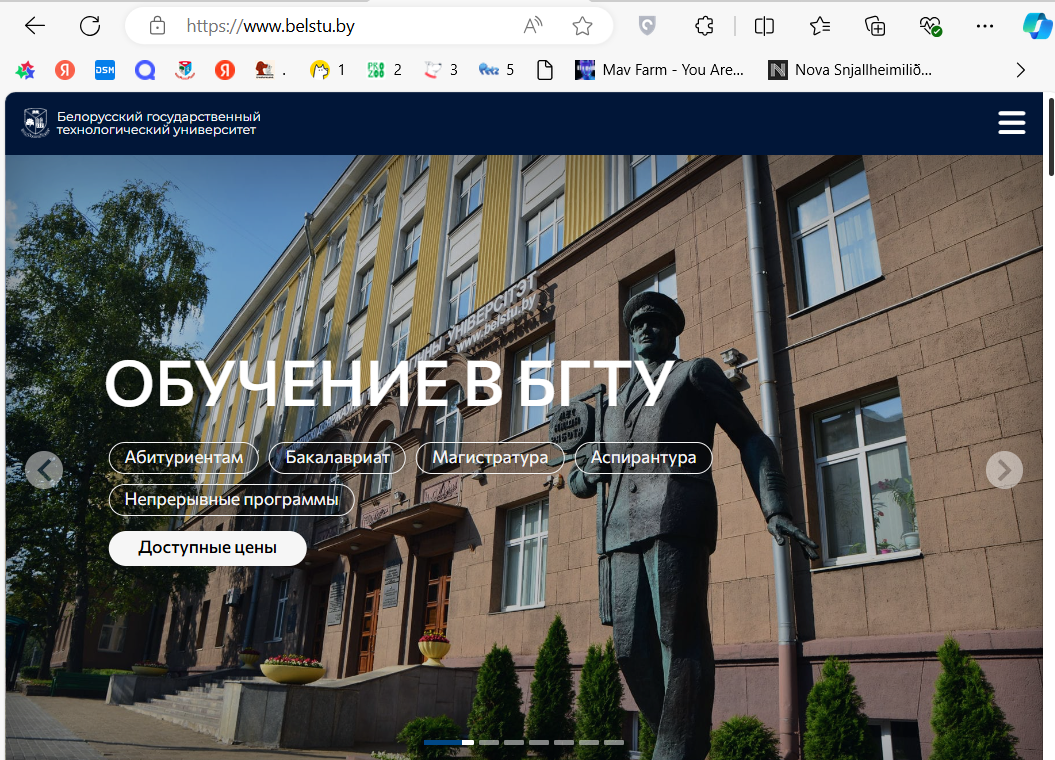
**Content-Type** – указывает на тип мультимедийного ресурса, передаваемого в теле сообщения

1. Определите содержимое тела ответа. Поясните.

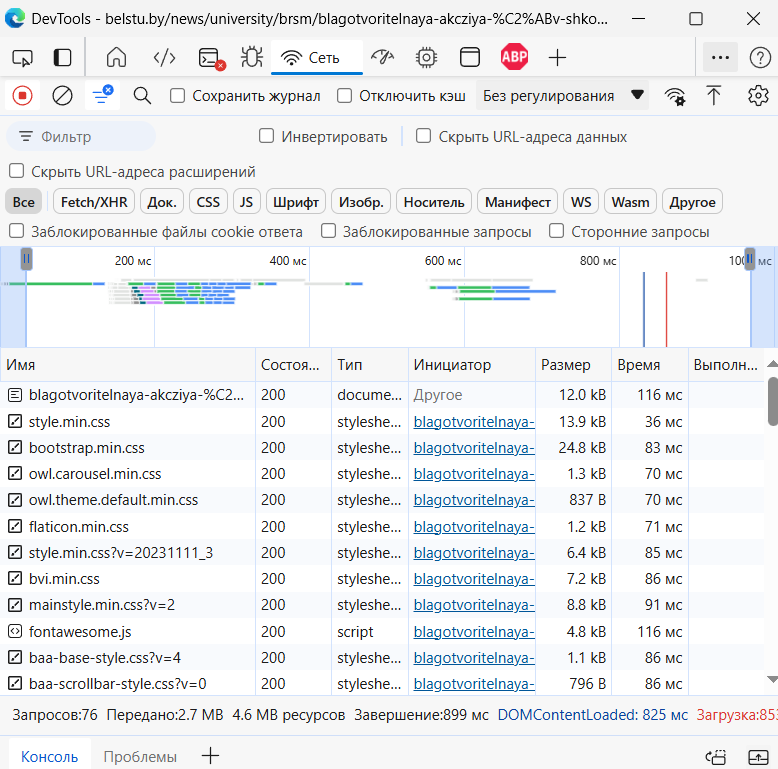


**Задание2**

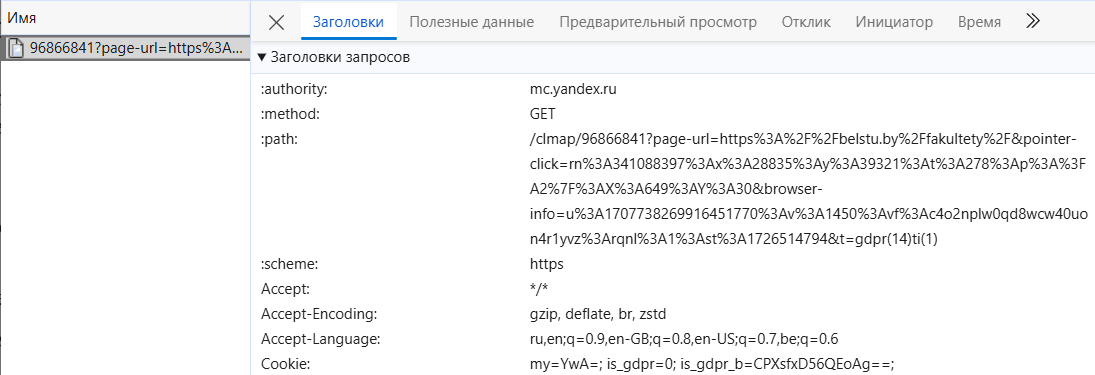
1. **WEB BROWSER**: выполните GET-запрос к сайту БГТУ: <https://www.belstu.by/>

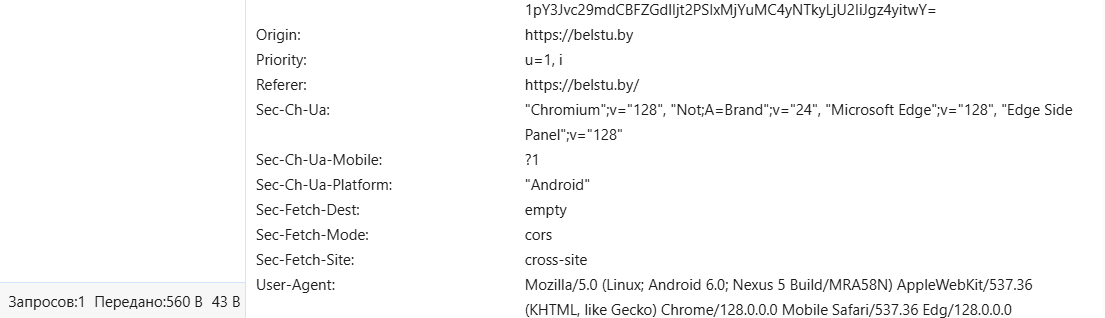


1. Используйте «**Инструмент разработчи**ка» браузера



1. Определите заголовки запроса. Определите содержимое тела запроса.





**Accept** – Указывает, какие типы контента клиент готов принять.

**accept-encoding** – Указывает, какие кодировки сжатия поддерживает клиент.

**accept-language** – Указывает предпочтения языков клиента для ответа.

**origin** – Указывает источник запроса (в данном случае, сайт belstu.by).

**priority** – Определяет приоритет запроса (например, `u=1` может означать высокий приоритет).

**referrer** – Указывает, с какого URL пришел запрос (здесь это belstu.by).

**sec-ch-ua** – Информация о браузере и его версии.

**sec-ch-ua-mobile** – Указывает, используется ли мобильное устройство

**sec-ch-ua-platform** – Платформа, на которой работает браузер (в данном случае, Android).

**sec-fetch-dest** – Определяет цель запроса (пустое значение указывает на отсутствие конкретной цели).

**sec-fetch-mode** – Указывает режим запроса (например, `cors` для кросс-доменных запросов).

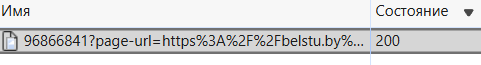
**sec-fetch-site** – Тип сайта, с которого был сделан запрос (в данном случае, `cross-site`).

**user-agent** – Информация о браузере и операционной системе пользователя.

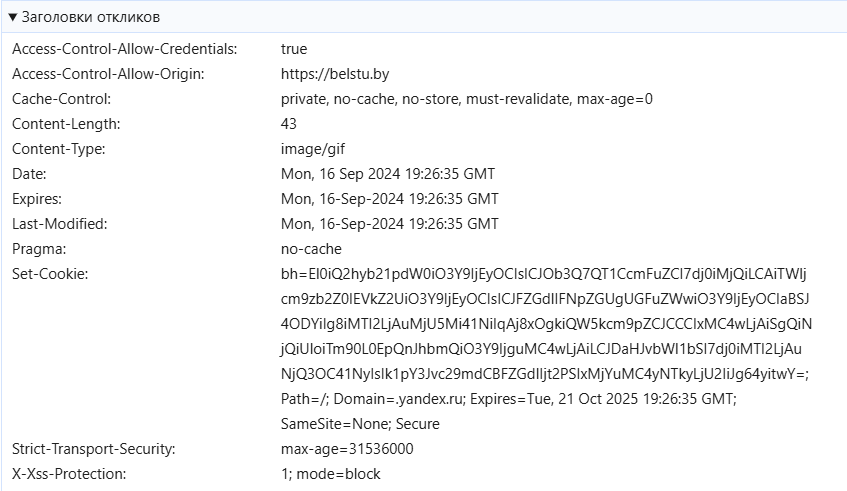
1. Определите содержимое тела запроса.

пустое

1. Определите статус ответа.



1. Определите заголовки.

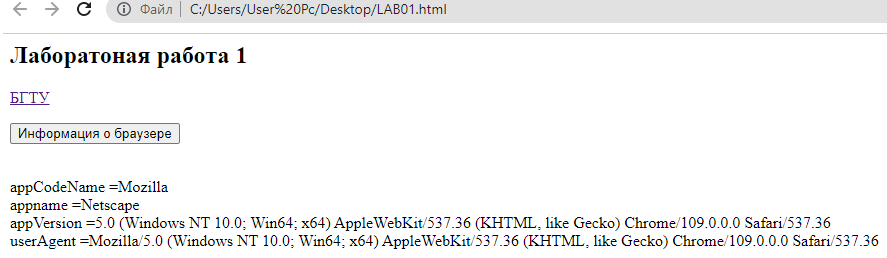


1. Определите содержимое тела ответа.

Это наша веб-страничка

**Задание3**

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.



1. Загрузите разработанную страницу в браузер и выполните запрос к сайту БГТУ с помощью ссылки.
2. Выполните анализа запроса и ответа с помощью «**Инструмент разработчи**ка» браузера

Sec-Fetch-Site. Было none (операция была инициирована пользователем), 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?

относится к прикладному уровню (7-й уровень)

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: стандартный порт для HTTP.

- 443: стандартный порт для HTTPS (HTTP Secure).

1. Что такое RFC?

RFC (Request for Comments) – содержит технические спецификации и стандарты, широко применяемые во всемирной сети.

1. Почему HTTP-протокол называют ассиметричным?

Cообщения, которые идут от клиента к серверу, отличаются от сообщений, которые идут от сервера к клиенту

1. Что такое MIME?

Multipurpose Internet Mail Extensions — стандарт, который позволяет передавать различные типы данных через интернет, добавляя поддержку текстов, изображений, аудио и видео, а также определяя заголовки для их обработки.