

# Лабораторная работа № 1

## Изучение протокола HTTP

### Задание

Данная работа является вводной в курсе и призвана ознакомить студента с базовым протоколом web – протоколом HTTP.

\*Примечание: под заданием вы найдете подсказки и необходимый теоретический материал.

**Результатом работы является отчет в формате MD и код сервера, выложенные в репозиторий.**

#### 1. Базовая часть работы

- 1.1. Цель данной работы – ознакомиться с применением протокола HTTP на практике, в реальных системах. Каждый из рассмотренных типов запросов предлагается отправить на несколько известных интернет-сервисов. Впрочем, сервисы указаны лишь как примеры и при желании вы можете выбрать другие (социальные сети, почта, облака, новостные сайты и т.д.).
- 1.2. С помощью специального ПО (Postman, либо многочисленные аналоги, например, Restlet Client - расширение для Chrome) вручную отправить следующие запросы и ответить на предлагаемые вопросы.

- 1.2.1. Запрос OPTIONS. Отправьте запрос на <http://mail.ru>, <http://ya.ru>, [www.rambler.ru](http://www.rambler.ru), <https://www.google.ru>, <https://github.com/>, [www.apple.com/](http://www.apple.com/).

Для чего используется запрос OPTIONS? Какие коды ответов приходят при этом запросе? Какие сайты правильно обработали запрос и вернули ожидаемые данные?

- 1.2.2. Запрос HEAD. [vk.com](http://vk.com), [www.apple.com](http://www.apple.com), [www.msn.com](http://www.msn.com).

Для чего нужен запрос HEAD? Какой сайт прислал ожидаемый ответ?

- 1.2.3. Запросы GET и POST. Отправьте по запросу на [yandex.ru](http://yandex.ru), [google.com](http://google.com) и [apple.com](http://apple.com). Что они вернули? Что содержится в теле ответа?

- 1.3. Работа с API сайта. Многие крупные сервисы предоставляют открытое API. Как правило, оно реализовано на подходе REST, но это необязательно. Такое API используется сторонними сервисами и приложениями, которые хотят воспользоваться услугами предоставляющего такое API сервиса. Рассмотрим такое API на примере сайта [vk.com](http://vk.com) (при желании можно выбрать другой подходящий сервис).

- 1.3.1. Зайдите на [https://vk.com/dev/api\\_requests](https://vk.com/dev/api_requests) и посмотрите структуру запросов к данному API.

- 1.3.2. Используя документацию (<https://vk.com/dev/methods>) выполните следующие задания (обратите внимание, запросы нужно отправлять не из предложенной на сайте формы, а как в предыдущем задании):
  - 1.3.2.1. Получите список всех факультетов МГТУ им. Н.Э.Баумана.
  - 1.3.2.2. Получите свою аватарку.
  - 1.3.2.3. Ответьте на вопросы: какой код ответа присылается от api? Что содержит тело ответа? В каком формате и какой кодировке содержатся данные? Какой веб-сервер отвечает на запросы? Какая версия протокола HTTP используется?
- 1.3.3. POST запросы проще отправлять с формы, встроенной в документацию api. Чтобы посмотреть, как выглядит запрос, можно воспользоваться панелью разработчика браузера (F12 в Chrome -> вкладка Network).
  - 1.3.3.1. Отправьте запись на стену любому пользователю/группе и убедитесь, что она пришла.
  - 1.3.3.2. Ответьте на вопрос: каким образом передаются данные от пользователя к серверу в POST-запросах?
2. Реализуйте небольшое серверное приложение, с использованием любого фреймворка. Лучшего всего для этой цели подойдет NodeJS: решение получится очень компактным и простым.

Сервер должен содержать предоставлять некоторое API с поддержкой (GET, POST, DELETE, PUT, OPTION). Данные отправлять в формате json. Конкретное содержание запросов - на ваше усмотрение. Подключите фантазию. (Можно сделать простейший CRUD-сервис с хранением данных в RAM).