Лабораторная работа 02

ПвИ

**Задание 01**

1. Разработайте серверное приложение **02-01**, которое на запрос [**http://localhost:5000**](http://localhost:5000)**/html** возвращает ответ, в теле которого содержится содержимое файла **index.html** (расположен в той же папке, что и скрипт **02-01.js**).
2. Файл **index.html** содержит статическую HTML-разметку с данными студента (факультет, курс, группа, фамилия, имя и отчество).
3. С помощью браузера проверьте работоспособность приложения.

**Метод fs.readFileSync()** — это встроенный интерфейс прикладного программирования модуля fs, который используется для чтения файла и возврата его содержимого.

В методе fs.readFile() мы можем читать файл неблокирующим асинхронным способом, но в методе fs.readFileSync() мы можем читать файлы синхронно, т. е. мы сообщаем node.js, чтобы заблокировать другой параллельный процесс и выполнить текущий процесс чтения файла. То есть, когда вызывается метод fs.readFileSync(), исходная программа узла прекращает выполнение, и узел ожидает выполнения функции fs.readFileSync(), после получения результата метода выполняется оставшаяся программа узла.

**Задание 02**

1. Разработайте серверное приложение **02-02**, которое на GET-запрос [**http://localhost:5000**](http://localhost:5000)**/png** возвращает ответ, в теле которого содержится содержимое файла **pic.png** (расположен в той же папке, что и скрипт **02-02.js**).
2. Файл **pic.png** содержит любое изображение в формате png.
3. С помощью браузера проверьте работоспособность приложения.

**Задание 03**

1. Разработайте серверное приложение **02-03**, которое на GET-запрос [**http://localhost:5000/api/name**](http://localhost:5000/api/name)возвращает ответ, в теле которого содержится ваша фамилия имя и отчество в текстовом виде (text/plain).
2. C помощью браузера проверьте работоспособность приложения.
3. Проверьте работоспособность приложения с помощью **POSTMAN**.

**Задание 04**

1. Разработайте серверное приложение **02-04**, которое на запрос к **<http://localhost:5000/xmlhttprequest>** возвращает ответ, в теле которого содержится содержимое файла **xmlhttprequest.html** (расположен в той же папке, что и скрипт **02-04.js**).
2. Файл **xmlhttprequest.html** содержит HTML-разметку и JS.
3. С помощью объекта **XMLHTTPRequest** выполняется асинхронный GET-запрос к [**http://localhost:5000/api/name**](http://localhost:5000/api/name) иполученный ответ (фамилия, имя и отчество) динамически встраивается в HTML-разметку страницы.

**Задание 05**

1. Разработайте серверное приложение **02-05**, которое на запрос к [**http://localhost:5000/fetch**](http://localhost:5000/fetch)возвращает ответ, в теле которого содержится содержимое файла **fetch.html** (расположен в той же папке, что и скрипт **02-05.js**).
2. Файл **fetch.html** содержит HTML-разметку и JS.
3. С помощью объекта **fetch** выполняется асинхронный GET-запрос к [**http://localhost:5000/api/name**](http://localhost:5000/api/name)иполученный ответ (фамилия, имя и отчество) динамически встраивается в HTML-разметку страницы.

**Задание 06**

1. Разработайте серверное приложение **02-06**, которое на запрос к [**http://localhost:5000/jquery**](http://localhost:5000/jquery)возвращает ответ, в теле которого содержится содержимое файла **jquery.html** (расположен в той же папке, что и скрипт **02-06.js**).
2. Файл **jquery.html** содержит HTML-разметку и JS.
3. С помощью **jQuery.Ajax** (функция библиотеки jQuery, обеспечьте в **jquery.html** соответствующий <script> c jQuery CDN: [https://code.jquery.com](https://code.jquery.com/)) выполняется асинхронный GET-запрос к [**http://localhost:5000/api/name**](http://localhost:5000/api/name)иполученный ответ (фамилия, имя и отчество) динамически встраивается в HTML-разметку страницы.

**Задание 07.Ответьте на следующие вопросы**

1. Перечислите основные свойства **NODE.JS**
2. Что такое **npm**?
3. Поясните назначение HTTP-заголовка **Content-Type**.
4. Поясните назначение функции **require**.
5. Поясните понятие «**Модуль Node.js**».
6. Поясните понятие «**Node.js built-in modules**» («**Node.js Core modules**»).
7. Какой модуль **NODE.JS** обеспечивает работу с протоколом HTTP?
8. Какой модуль **NODE.JS** обеспечивает работу с файловой системой?
9. **Перечислите основные свойства NODE.JS**

* основан на **Chrome V8**;
* **среда (контейнер) исполнения** приложений на JavaScript;
* поддерживает механизм **асинхронности**;
* ориентирован на **события**;
* **однопоточный** (код приложения исполняется только в одном потоке, один стек вызовов); обычно в серверах для каждого соединения создается свой поток, в Node.js все соединения обрабатываются в одном JS-потоке;
* **не блокирует** выполнение кода при вводе/выводе (в файловой системе до 4х одновременно);
* в состав Node.js входят инструменты: **npm** – пакетный менеджер; **gyp** - Python-генератор проектов; **gtest** – Google фреймворк для тестирования С++ приложений;

использует библиотеки: **V8** – библиотека V8 Engine, **libuv** – библиотека для абстрагирования неблокирующих операций ввода/вывода (представляет собой обертку над epoll, kqueue, IOCP); **llhttp** – легковесный парсер http-сообщений (написан на C и не выполняет никаких системных вызовов); **c-ares** -библиотека для работы с DNS; **OpenSSL**

1. **Что такое npm?**

**node** **package** **manager** — это стандартный менеджер пакетов, автоматически устанавливающийся вместе с **Node**.**js**. пакетом в **Node**.**js** называется один или несколько **JavaScript**-файлов, представляющих собой какую-то библиотеку или инструмент.

1. **Поясните назначение HTTP-заголовка Content-Type.**

В ответах сервера заголовок Content-Type сообщает клиенту, какой будет тип передаваемого контента.

1. **Поясните назначение функции require.**

require () — это **встроенная функция для включения внешних модулей, которые существуют в отдельных файлах**. Оператор require () читает файл JavaScript, выполняет его, а затем возвращает экспортированный объект.

1. **Поясните понятие «Модуль Node.js».**

это **набор функций и объектов JavaScript, который могут использовать внешние приложения**

1. **Поясните понятие «Node.js built-in modules» («Node.js Core modules»).**

Node.js имеет набор встроенных модулей, которые вы можете использовать без какой-либо дальнейшей установки.

1. **Какой модуль NODE.JS обеспечивает работу с протоколом HTTP?**

http

1. **Какой модуль NODE.JS обеспечивает работу с файловой системой?**

fs