

Кафедра информационных систем и технологий (№ 42)

Дисциплина «Web-программирование»
Автор – ст. преподаватель каф. 42 В.А. Ушаков

Лабораторная работа № 3

РАБОТА С ФАЙЛАМИ В NODE.JS

Цель работы – Изучить функции для работы с файлами (файловые функции) в среде Node.JS.

Краткие методические сведения

Некоторые правила, которым нужно следовать для работы с файлами в HTML-форме (Без этих требований загрузка файла работать не будет):

- убедиться, что HTML-форма использует метод POST (method="post");
- форма должна содержать следующий атрибут enctype="multipart/form-data", который указывает, какой тип контента использовать при отправке формы;
- атрибут type="file" тега <input> показывает поле ввода, как элемент управления выбора файла в браузере.

Задание

Используя web-страницу, полученную в ходе выполнения лабораторной работы 2, дополнить ее следующими возможностями.

- 1 Написать JS-скрипт в среде Node.JS, позволяющий загружать файл на сервер, который содержит дополнительную информацию по предметной области лабораторной работы 1. Скрипт должен выводить сообщение с результатом загрузки (успех/неуспех). *Необходимо реализовать*

дополнительным полем в HTML-форме (интегрировать в форму).

- 2 Дополнить в JS-скрипт из пункта 1 возможностью проверки типа и размера загружаемого файла. Тип файла определяется по номеру варианта из таблицы 1, а размер файла – из таблицы 2.
- 3 Добавить в получившуюся в пункте 1 web-страницу кнопку, которая позволяет записать всю текстовую информацию из полей формы с названиями полей формы в текстовый файл (можно *.txt). Для работы кнопки необходимо дополнить JS-скрипт из пункта 2. После сохранения информации в файл форма должна предложить просмотреть/скачать получившийся файл. Когда скачивание будет завершено, необходимо выдать сообщение согласно номеру варианта из таблицы 3. *Текст необходимо структурировать.*

Таблица 1 – Перечень типов файлов

№ варианта	Тип файла	№ варианта	Тип файла
1	.doc	6	.rtf
2	.mp3	7	.wav
3	.flv	8	.avi
4	.bmp	9	.jpg
5	.7z	10	.zip

Таблица 2 – Перечень результирующих сообщений

№ варианта	Размер файла
1, 6	50 – 100 кб
2, 7	1 – 5 мб
3, 8	10 – 20 мб
4, 9	100 – 200 кб
5, 10	5 – 10 мб

Таблица 3 – Перечень результирующих сообщений

№ варианта	Содержание сообщения
1, 6	Размер файла
2, 7	Время последнего изменения файла
3, 8	Тип файла
4, 9	Время последнего доступа к файлу
5, 10	Права доступа к файлу

Примечания к выполнению лабораторных работ:

В ходе выполнения ЛР следует использовать HTML5, CSS3 (*в частности, блочную верстку или FlexBox*) и среда Node.JS 17 или новее.

В ходе выполнения ЛР следует выносить код каскадной таблицы стилей и сценариев в файлы *.css и *.js соответственно. Код скриптов должен располагаться в файлах *.js. *Не следует использовать Javascript, JQuery и другие технологии для тех задач, которые можно решить в среде Node.JS.*

Рекомендуется использовать относительные пути при создании *.html-, *.css-, *.js- файлов для обращения к рисункам, файлам *.css и т.д.

Содержание отчета

- 1 Титульный лист
- 2 Цель работы
- 3 Задание к лабораторной работе
- 4 Код web-страниц (*.html и *.css) и/или скриптов (*.js)

Листинг кода должен содержать комментарии

- 5 Примеры web-страниц
- 6 Выводы по лабораторной работе

Требования к оформлению отчета о лабораторной работе

При оформлении отчета о лабораторной работе следует пользоваться ГОСТ 7.32-2017 издания 2017 года. Правила оформления текстовых документов по ГОСТ 7.32-2017, а также титульные листы лабораторных работ представлены на сайте ГУАП (<https://guap.ru/regdocs/docs/uch#rules>).

Рекомендуемые источники информации:

- 1 <https://nodejsdev.ru/>
- 2 <https://nodejs.org/ru/docs/guides>
- 3 Сухов К.К. Node.js. Путеводитель по технологии. – М.: ДМК Пресс, 2015. – 416 с.
- 4 Кантелон М., Хантер М., Головайчук Т., Райлих Н. Node.js в действии. – СПб.: Питер, 2014. – 548 с.
- 5 Пауэрс Ш. Изучаем Node. Переходим на сторону сервера. СПб.: Питер, 2017. – 304 с.