

Лабораторная работа №6 и №7 Фреймворки JavaScript и средства сборки

Node.js

Программная платформа для исполнения JavaScript кода, позволяющая использовать JS-скрипты как нативные приложения операционной системы.

Официальное руководство

<https://nodejs.org/>

Скринкасты по установке и использованию

<https://learn.javascript.ru/screencast/nodejs>

Средства сборки Frontend

Большое количество фреймворков, LESS/SASS, coffeescript требуют возможности их соединить, унифицировать и привести к js/css файлам, доступным любому браузеру. Большинство всех средств сборки требуют наличия Node.js.

Grunt

GruntJS — это инструмент командной строки для интерфейсных разработчиков для выполнения predetermined повторяющихся задач.

Используется для автоматизации рабочих процессов JavaScript, таких как объединение и минимизация JS-файлов, проведение тестов, имеет большую базу плагинов для расширения возможностей.

Официальное руководство

<https://gruntjs.com/getting-started>

Хабрастатья с обзором

<https://habr.com/post/177395/>

Как использовать в PhpStorm / WebStorm

<https://www.jetbrains.com/help/phpstorm/using-grunt-task-runner.html>

Как использовать в Visual Studio

<https://webtooling.visualstudio.com/task-runners/grunt/>

Gulp

GulpJS — инструмент, аналогичный Grunt, но использует другой подход. Вместо описания конфигурации сборки используются функции-сборщики для каждой задачи. Также обладает большим количеством плагинов.

Официальное руководство

<https://gulpjs.com/>

Скринкасты по использованию

<https://learn.javascript.ru/screencast/gulp>

Настройка в PhpStorm / WebStorm

<https://sawtech.ru/tehnoblog/nastrojka-gulp-dlya-front-end-razrabotki-v-phpstorm/>

Webpack

Webpack — модульная система сборки приложений, реализует не только сборку файлов, но

анализирует синтаксис и учитывает зависимости различных библиотек.

Официальное руководство

<https://webpack.js.org/>

Хабрапособие по webpack

<https://habr.com/post/309306/>

Скринкасты по установке и использованию

<https://learn.javascript.ru/screencast/webpack>

Фреймворки JavaScript

Angular.js

Предназначен для разработки одностраничных приложений, реализует концепцию MVC-шаблонов.

Официальное руководство

<https://angularjs.org/>

Учебный курс от W3C

https://www.w3schools.com/angular/angular_intro.asp

Скринкасты по настройке и использованию

https://www.youtube.com/playlist?list=PLY4rE9dstrJxWEX3fCPjFpmcnoU_3GRWW

React.js

Предназначен для создания интерактивных пользовательских интерфейсов.

Официальное руководство

<https://reactjs.org/>

Скринкасты по использованию

<https://learn.javascript.ru/screencast/react>

Vue.js

Предназначен для быстрого создания интерактивных пользовательских интерфейсов

Официальное руководство

<https://vuejs.org/>

Хабраобзор

<https://habr.com/post/329452/>

Скринкасты по использованию

<https://learn.javascript.ru/screencast/webpack>

Варианты заданий к лабораторной работе

По порядковому номеру в списке на сайте asb.im:

- 1-7 — реализуют сборку на grunt.js
- 8-14 — реализуют сборку на gulp.js
- 15-21 — реализуют сборку на webpack

По номеру варианта в списка на сайте asb.im:

- 1-9 — используют Angular.js
- 10-19 — используют React.js
- 20-29 — используют Vue.js

Вариант задания каждый выбирает сам.

«Кубок мира»

Разработайте одностраничный сайт с результатами чемпионата мира по футболу 2014. Оформите сайт в официальные цвета чемпионата, используйте флаги стран-участниц для создания спрайта.

Источник данных: https://ru.wikipedia.org/wiki/%D0%A7%D0%B5%D0%BC%D0%BF%D0%B8%D0%BE%D0%BD%D0%B0%D1%82_%D0%BC%D0%B8%D1%80%D0%B0%D0%BF%D0%BE_%D1%84%D1%83%D1%82%D0%B1%D0%BE%D0%BB%D1%83_2014

На сайте должна быть одна таблица со всеми групповыми зачетами и следующими возможностями:

- Поиск по названию страны-участники. По мере набора в поле поиска скрываются все ячейки, не попадающие под условие. Критерий поиска — вхождение подстроки. Например, для ввода «ина» подходят «Аргентина» и «Босния и Герцеговина».
- Выполнить сортировку по количеству очков, выигрывшей, мячей, ничьих и побед.
- Разница в мячах должна вычисляться на основе результатов игр.
- Возможность редактировать все записи и данные в них.

«Демография России»

Разработайте одностраничный сайт с данными о рождаемости и смертности в регионах России в 2016 и 2017 годах. Оформите в любой цветовой схеме, используйте значки «больше», «меньше» «без изменений» для создания спрайта.

Источник данных: https://ru.wikipedia.org/wiki/%D0%9D%D0%B0%D1%81%D0%B5%D0%BB%D0%B5%D0%BD%D0%B8%D0%B5_%D0%A0%D0%BE%D1%81%D1%81%D0%B8%D0%B8

На сайте должна быть таблица со следующими функциями:

- Поиск по названию субъекта РФ. По мере набора в поле поиска скрываются все ячейки, не попадающие под условие. Критерий поиска — вхождение подстроки. Например, для ввода «область» подходят все области РФ.
- Выполнить сортировку по рождаемости и смертности, при этом группировки таблицы исчезают. Возможность сбросить сортировку.
- Среднее по федеральным округам и итог по таблице должны вычисляться.

- Возможность добавлять новые года (2015, 2018) и заполнять данные.

«Чемпионы формулы 1»

Разработайте одностраничный сайт с данными о чемпионах. В оформлении используйте оформление сайта <https://www.formula1.com/> Для спрайта используйте флаги команд и чемпионов.

Источник данных: https://ru.wikipedia.org/wiki/%D0%A1%D0%BF%D0%B8%D1%81%D0%BE%D0%BA_%D1%87%D0%B5%D0%BC%D0%BF%D0%B8%D0%BE%D0%BD%D0%BE%D0%B2_%C2%AB%D0%A4%D0%BE%D1%80%D0%BC%D1%83%D0%BB%D1%8B-1%C2%BB

На сайте должна быть таблица со следующими возможностями:

- Фильтр по производителю шин.
- Фильтр по стране. При выборе флага скрываются все записи, где флаг не является частью команды или чемпиона.
- Сортировка по поулам, победам, подиумам, быстрым кругам и очкам.
- Возможность добавлять новые записи.

Критерии сдачи лабораторной работы:

ЛР№6. Сборка ресурсов

- ✓ Настроек сборщик согласно варианту.
- ✓ Генерируется один сжатый js-файл и один сжатый css-файл из всех ресурсов.
- ✓ Генерируется png-картинка со спрайтом на основе папки с картинками.

ЛР№7. JS-фреймворки

- ✓ Используется фреймворк согласно варианту.
- ✓ Поиск, сортировка и фильтры работают корректно, в том числе после изменения таблиц.
- ✓ Таблица является отдельным компонентом, отрисовка и рендер находится на стороне JS-фреймворка, а шаблоне страницы находится только контейнер.
- ✓ Данные для таблицы находятся в JSON-массиве, который загружается асинхронно и хранится в LocalStorage браузера