МИНИСТЕРСТВО НАУКИ И ВЫСШЕГО ОБРАЗОВАНИЯ РФ ФЕДЕРАЛЬНОЕ ГОСУДАРСТВЕННОЕ БЮДЖЕТНОЕ ОБРАЗОВАТЕЛЬНОЕ УЧРЕЖДЕНИЕ ВЫСШЕГО ОБРАЗОВАНИЯ **«БЕЛГОРОДСКИЙ ГОСУДАРСТВЕННЫЙ ТЕХНОЛОГИЧЕСКИЙ УНИВЕРСИТЕТ им. В. Г. ШУХОВА» (БГТУ им. В.Г. Шухова)**

Кафедра программного обеспечения вычислительной техники и автоматизированных систем

Лабораторная работа №2  
по дисциплине: Технологии Web-программирования

Выполнил: Ляхов Степан ПВ-42

Проверил: Картамышев С. В.

Белгород 2020г.

**Цель**: познакомиться с основами разработки на языке JavaScript. Познакомится с основами разработки frontend приложения, в частности с использованием клиентских фреймворков (React). Научиться переносить вёрстку в отдельные компоненты.

**Задание к лабораторной работе:**

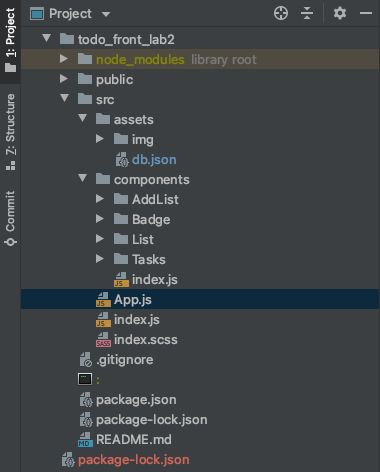
1. Изучить основы разработки на языке JavaScript.
2. Изучить основы разработки frontend-приложения.
3. Развернуть базовое приложение на фреймворке React JS.
4. Добавить необходимые компоненты и перенести в них вёрстку, сделанную в прошлой лабораторной работе.
5. Продемонстрировать работу web-приложения.

**Выполнение**

В данной лабораторной работе мы будем реализовывать ToDoIst сайт, используя технологию React JS

Стек технологий:  
HTML, CSS, JS, REACT, NPM

Создаём новый проект с такой структурой:



README.md – текстовый файл, в котором находится описание и инструкция для развёртывания.

public – директория, в которой будут храниться ресурсы или статические данные.

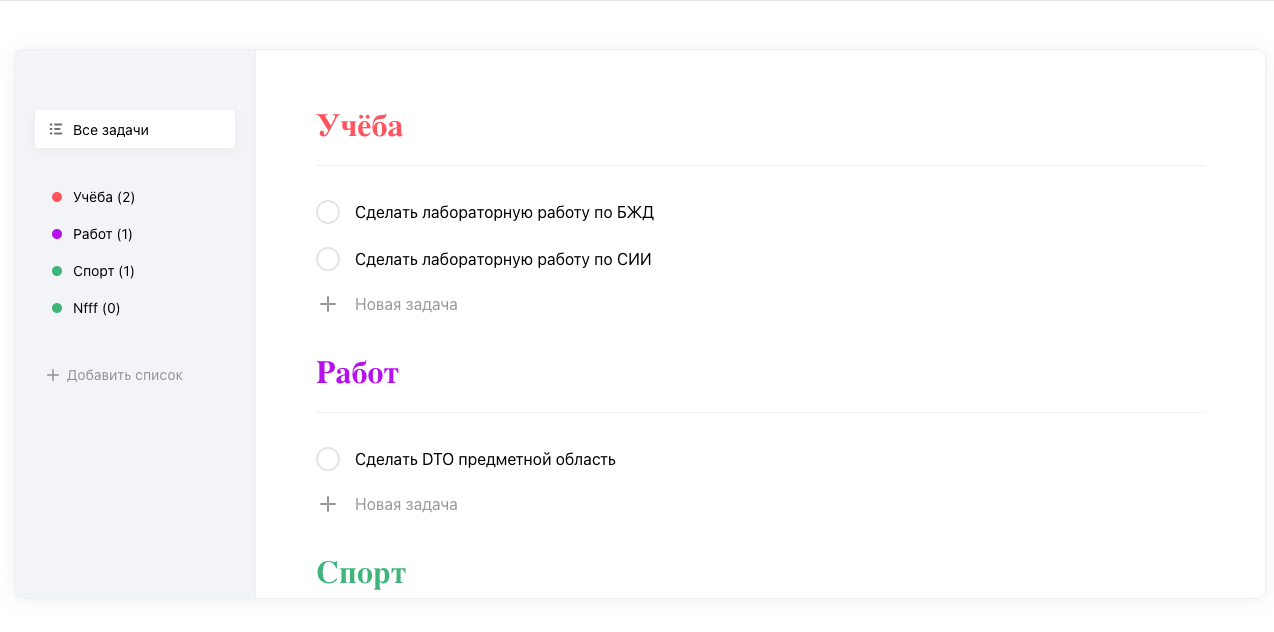
package.json – файл, в котором прописаны скрипты для запуска, конфигурации и необходимые зависимости.

assets – статические данные, необходимые для генерации, внедрения в приложения и.т.д

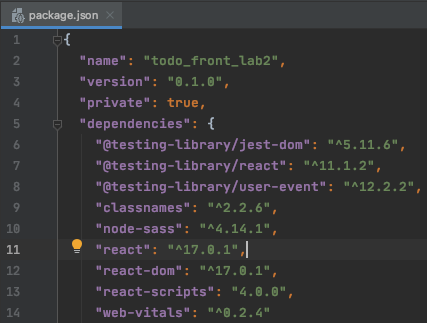
db.json – наша локальная база данных в формате JSON

components – директория в которой будут хранится наши созданные компоненты.

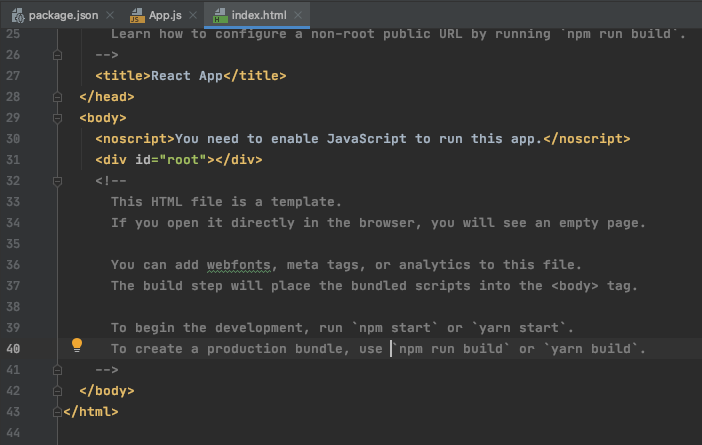
**Внешний вид сайта**

****

**Подключение необходимых библиотека**



**Корень проекта**



React так устроен, что ему нужен базовый шаблон в котором будет root dev элемент, внутрь которого уже будет внедрятся сгенерированный React-ом код

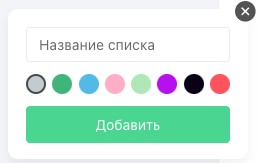
**Главный компонент**



Генерация кода в данном примере начинается с нашего главного компонента, который, вызывается генерацию других компонентов, или обращение к серверу через REST API по http.

В этом же компоненте, обрабатываются события, например создание нового списка или изменения заголовка.

**Примерно компонента**

****

Круги на этой форме – это компонент, код которого показан ниже.

В этот компонент передаётся цвет, нажат не нажат, и имя класса

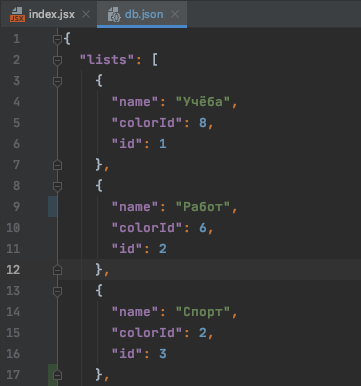
****

Вернётся уже полностью сгенерированный компонент.

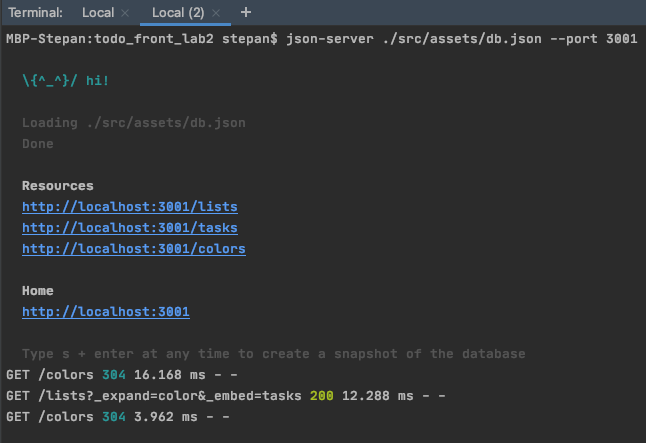
**Фейкавая БД**

Так как в данной лабораторной не было задачи связаться с беком, мы будем использовать утилиту json-server, которая на вход принимает файл в формате json, а через http даёт нам API для получения этих данных.

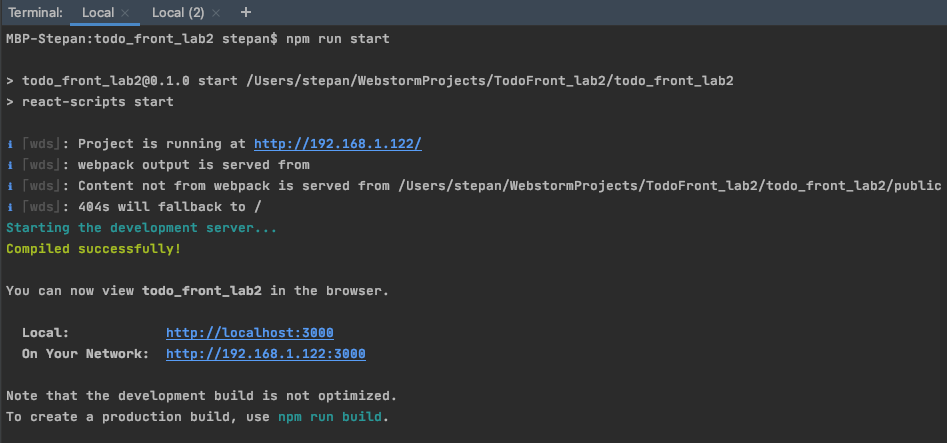
Наша бд



Запуск json-server



**Запуск приложения**

****