

# Контрольная работа: Минимальное To-Do приложение на React

## Цель:

Создать простое одностраничное приложение для управления списком задач: добавление и у

## Функциональные требования:

### 1. Добавление задач

- Пользователь может ввести текст задачи в текстовое поле.
- По нажатию на кнопку "Добавить" задача добавляется в список ниже.
- После добавления задача очищается из поля ввода.
- Пустые задачи (только пробелы) добавлять нельзя — при попытке должна выводиться ошиб

### 2. Удаление задач

- У каждой задачи в списке есть кнопка "Удалить".
- По нажатию на кнопку задача удаляется из списка.

### 3. Сохранение данных

- Все задачи должны сохраняться в localStorage.
- При перезагрузке страницы список задач должен загружаться из localStorage.

## Технические требования:

### 1. Используемые технологии

- React (только функциональные компоненты)
- Стилизация через CSS или inline-стили

### 2. Структура проекта

```
src/  
├── components/  
│   ├── TodoInput.jsx  
│   ├── TodoList.jsx  
│   └── TodoItem.jsx  
├── App.jsx  
└── index.js
```

## Этапы разработки:

### Этап 1: Создание проекта

- Использовать create-react-app.
- Удалить лишние файлы и настроить чистую структуру.

### Этап 2: Реализация ввода задачи (TodoInput.jsx)

- Input + кнопка "Добавить".
- Состояние для ввода (useState).
- Валидация: запрет на пустой ввод.
- После успешного добавления — очистка поля.

### Этап 3: Отображение задач (TodoList.jsx)

- Получает список задач и отображает через TodoItem.

### Этап 4: Компонент задачи (TodoItem.jsx)

- Отображает текст задачи и кнопку "Удалить".

### Этап 5: Логика в App.jsx

- useState для задач.
- useEffect для загрузки/сохранения в localStorage.
- Функции addTodo, removeTodo.

### Стилизация (CSS):

#### Общие

- Центрировать приложение.
- Максимальная ширина — 600px.
- Использовать Flexbox для размещения элементов.

#### Инпут и кнопка

- На одной строке.
- Отступы, рамка, скругления.
- Кнопка с цветом и эффектом наведения.

#### Список задач

- Каждая задача в отдельной карточке (с тенью или рамкой).
- Красная кнопка "Удалить".

#### Сообщения

- Показывать сообщение об ошибке при пустом вводе.
- Сообщение "Нет задач" при пустом списке.

#### Дополнительные бонусы (необязательно)

- Добавление по нажатию Enter.
- Анимации при добавлении/удалении.
- Счётчик задач.
- Фильтрация (все / выполненные / невыполненные — если есть время на доработку).

#### Что сдавать

- Ссылка на GitHub репозиторий.
- Скриншоты работающего приложения.
- Желательно — ссылка на развернутое приложение (например, Vercel, Netlify).