# ЛАБОРАТОРНАЯ РАБОТА №12

1. **Установим зависимости**

**npm install --save-dev @testing-library/react @testing-library/jest-dom jest**

1. **Напишем все Тесты по данному списку ниже**
2. Проверка отображения содержимого домашнего страницы:
   * заголовок (текст)
   * кнопки (количество + содержимое)
   * поля ввода (количество + содержимое)
   * заглушки для случая пустого списка и пр (текст)
3. Проверка правильной работы поля ввода:
   * в поле вводятся данные
   * кликается кнопка сабмита
   * в списке появляется тудушка
4. Проверка работы компонента, отображающего список:
   * отображение пустого массива тудушек
   * отображение заполненного массива (текст + кнопки действия, если они есть для каждого элемента списка)
5. Проверка работы кнопок фильтрации и сортировки
   * отображение полного набора тудушек
   * отображение отфильтрованного набора
6. **Листинг скриптов App.test.js и TodoList.test.js**

import React from 'react';

import { render, screen } from '@testing-library/react';

import App from './App';

test('renders home page content', () => {

render(<App />);

// Проверка заголовка

const header = screen.getByText(/Список дел/i);

expect(header).toBeInTheDocument();

// Проверка кнопок

const addButton = screen.getByText('Добавить');

const clearButton = screen.getByText('Очистить всё');

expect(addButton).toBeInTheDocument();

expect(clearButton).toBeInTheDocument();

// Проверка полей ввода

const input = screen.getByPlaceholderText('Новый элемент списка');

expect(input).toBeInTheDocument();

// Проверка заглушки для пустого списка

const emptyStub = screen.queryByText(/Не найдено ни одного дела/i);

if (emptyStub) {

expect(emptyStub).toBeInTheDocument();

}

});

**TodoList.test.js**

import React from 'react';

import { render, screen, fireEvent } from '@testing-library/react';

import TodoList from './App.js';

test('adds a new todo', () => {

render(<TodoList />);

const input = screen.getByPlaceholderText('Новый элемент списка');

const addButton = screen.getByText('Добавить');

fireEvent.change(input, { target: { value: 'Новая задача' } });

fireEvent.click(addButton);

const newTodo = screen.getByText('Новая задача');

expect(newTodo).toBeInTheDocument();

});

test('renders empty todo list', () => {

render(<TodoList />);

const clearButton = screen.getByText('Очистить всё');

fireEvent.click(clearButton);

const emptyStub = screen.queryByText(/Не найдено ни одного дела/i);

expect(emptyStub).toBeInTheDocument();

});

test('renders filled todo list', () => {

render(<TodoList />);

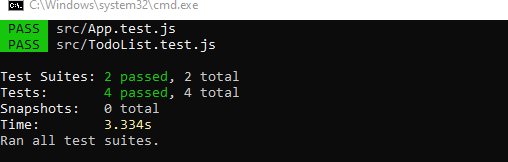
const todoItems = screen.getAllByRole('listitem');

expect(todoItems).toHaveLength(5); // начальное количество задач

expect(todoItems[0]).toHaveTextContent('Сделать дело');

expect(todoItems[1]).toHaveTextContent('Погладить кошку');

});



1. **При подключении всех библиотек и при запуске могут возникать ошибки с зависимости**

Листинг для помощи если такие ошибки возникают

**del package-lock.json**

**npm yarn lock**

**yarn install**

**npm install -g yarn**

1. **Проверка Тестов**

