**ФЕДЕРАЛЬНОЕ ГОСУДАРСТВЕННОЕ АВТОНОМНОЕ ОБРАЗОВАТЕЛЬНОЕ УЧРЕЖДЕНИЕ ВЫСШЕГО ОБРАЗОВАНИЯ**



**МОСКОВСКИЙ ПОЛИТЕХНИЧЕСКИЙ УНИВЕРСИТЕТ**

***Факультет Информационных технологий***

***Кафедра Информатики и информационных технологий***

**направление подготовки**

**09.03.02 «Информационные системы и технологии»**

**ЛАБОРАТОРНАЯ (ПРАКТИЧЕСКАЯ РАБОТА) №** \_2\_

**Дисциплина: Функциональное программирование**

**Тема:** \_\_\_\_ Применение функционального программирования в JS \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

**Вариант**

**Выполнил(а): студент(ка) группы 221-374**

**Максимов Юрий Сергеевич**

(Фамилия И.О.)

**Дата, подпись** \_\_10.02.2025 \_\_\_\_\_\_\_\_\_\_

(Дата)(Подпись)

**Проверил:** \_\_*\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ \_\_\_\_\_\_\_\_\_\_\_*

(Фамилия И.О., степень, звание) (Оценка)

**Дата, подпись** \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ *\_\_\_\_\_\_\_\_\_\_\_*

(Дата)(Подпись)

**Москва**

**2025**

**Задание**

Разработайте веб-приложение "Список задач", которое позволяет пользователю:

* Добавлять новые задачи.
* Отмечать задачи как выполненные.
* Удалять задачи.
* Фильтровать задачи по статусу (выполненные/невыполненные).

**Код**

**index.html**

|  |
| --- |
| **<!DOCTYPE html>**  **<html lang=”ru”>**  **<head>**  **<meta charset=”UTF-8”>**  **<meta name=”viewport” content=”width=device-width, initial-scale=1.0”>**  **<title>Список задач</title>**  **<link rel=”stylesheet” href=”styles.css”>**  **</head>**  **<body>**  **<h1>Список задач</h1>**    **<form id=”task-form”>**  **<input type=”text” id=”task-input”>**  **<button type=”submit” id=”task-submit”>Добавить</button>**  **</form>**    **<div id=”task-list”></div>**    **<div class=”filter-btns”>**  **<button id=”all-btn”>Показать все</button>**  **<button id=”completed-btn”>Выполнено</button>**  **<button id=”in-progress-btn”>Невыполнено</button>**  **</div>**    **<script src=”app.js”></script>**  **</body>**  **</html>** |

**app.js**

|  |
| --- |
| **// Определение кнопок**  **const task\_input = document.getElementById('task-input')**  **const task\_list = document.getElementById('task-list')**  **const all\_tasks\_btn = document.getElementById('all-btn')**  **const completed\_tasks\_btn = document.getElementById('completed-btn')**  **const in\_progress\_tasks\_btn = document.getElementById('in-progress-btn')**  **// Все задачи**  **let tasks = []**  **// Основные методы**  **const add\_task = (title) => {**  **const new\_task = { id: tasks.length + 1, title: title, status: false }**  **tasks = [...tasks, new\_task]**  **render\_tasks()**  **}**  **const delete\_task = (id) => {**  **tasks = tasks.filter(task => task.id !== id)**  **render\_tasks()**  **}**  **const change\_task\_status = (id) => {**  **tasks = tasks.map(task => {**  **if(task.id === id){**  **return { ...task, status: !task.status }**  **}**  **return task**  **})**  **render\_tasks()**  **}**  **const render\_tasks = (predicate = () => true) => {**  **task\_list.innerHTML = ''**  **tasks.forEach(task => {**  **if(predicate(task)){**  **const taskItem = document.createElement('li')**  **taskItem.className = 'task'**    **const checkbox = document.createElement('input')**  **checkbox.type = 'checkbox'**  **checkbox.checked = task.status**  **checkbox.addEventListener('change', () => change\_task\_status(task.id))**    **const label = document.createElement('label')**  **label.textContent = task.title**    **const deleteButton = document.createElement('button')**  **deleteButton.textContent = 'X'**  **deleteButton.addEventListener('click', () => delete\_task(task.id))**    **taskItem.appendChild(checkbox)**  **taskItem.appendChild(label)**  **taskItem.appendChild(deleteButton)**    **task\_list.appendChild(taskItem)**  **}**  **})**  **}**  **// Прослушиватели**  **document.getElementById('task-form').addEventListener('submit', event => {**  **event.preventDefault()**  **if(task\_input.value.trim() !== ''){**  **add\_task(task\_input.value)**  **task\_input.value = ''**  **}**  **})**  **all\_tasks\_btn.addEventListener('click', event => {**  **render\_tasks()**  **})**  **completed\_tasks\_btn.addEventListener('click', event => {**  **render\_tasks(task => task.status === true)**  **})**  **in\_progress\_tasks\_btn.addEventListener('click', event => {**  **render\_tasks(task => task.status === false)**  **})** |

**styles.css**

|  |
| --- |
| **body {**  **display: flex;**  **flex-direction: column;**  **align-items: center;**  **justify-content: center;**  **background-color: blanchedalmond;**  **gap: 15px;**  **}**  **.filter-btns button {**  **padding: 10px 15px;**  **background-color:cadetblue;**  **border: none;**  **border-radius: 15px;**  **}**  **#task-input {**  **border: none;**  **padding: 10px;**  **border-radius: 15px;**  **}**  **.task {**  **font-size: 20px;**  **justify-content: space-between;**  **align-items: center;**  **display: flex;**  **padding: 10px;**  **}**  **.task button {**  **padding: 10px 15px;**  **background-color:crimson;**  **border: none;**  **border-radius: 15px;**  **}**  **.task input[type="checkbox"] {**  **transform: scale(2);**  **}**  **#task-submit {**  **padding: 10px 15px;**  **background-color:rgb(4, 159, 107);**  **border: none;**  **border-radius: 15px;**  **}**  **#task-list {**  **width: 400px;**  **padding: 10px;**  **}** |

**Ссылка на код**

<https://github.com/MaksimovYuriy/FunctionalProgramming/tree/main/Лаб%202>

**Скриншоты выполнения**

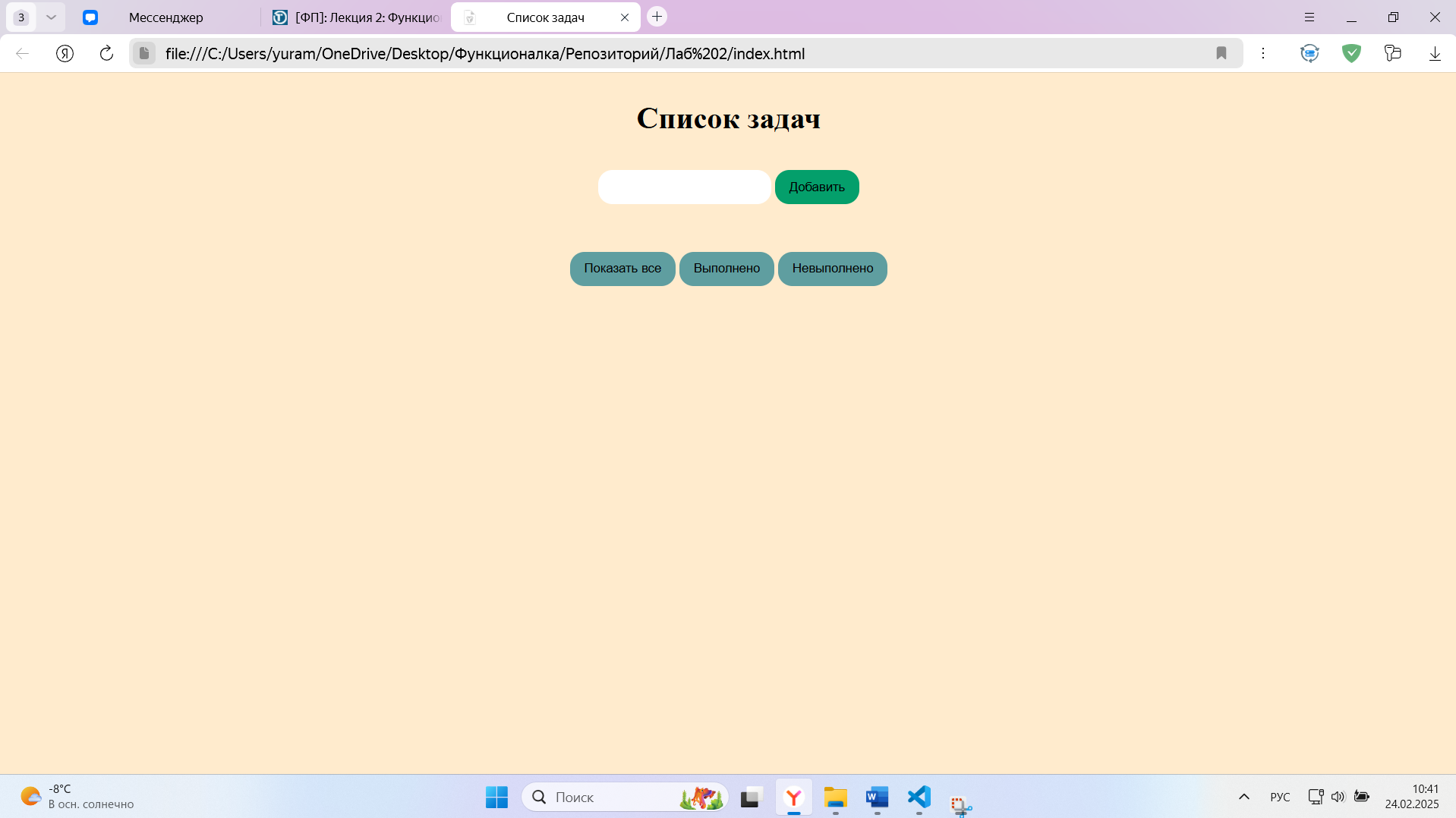
****

Рис. 1 – Выполнение программы

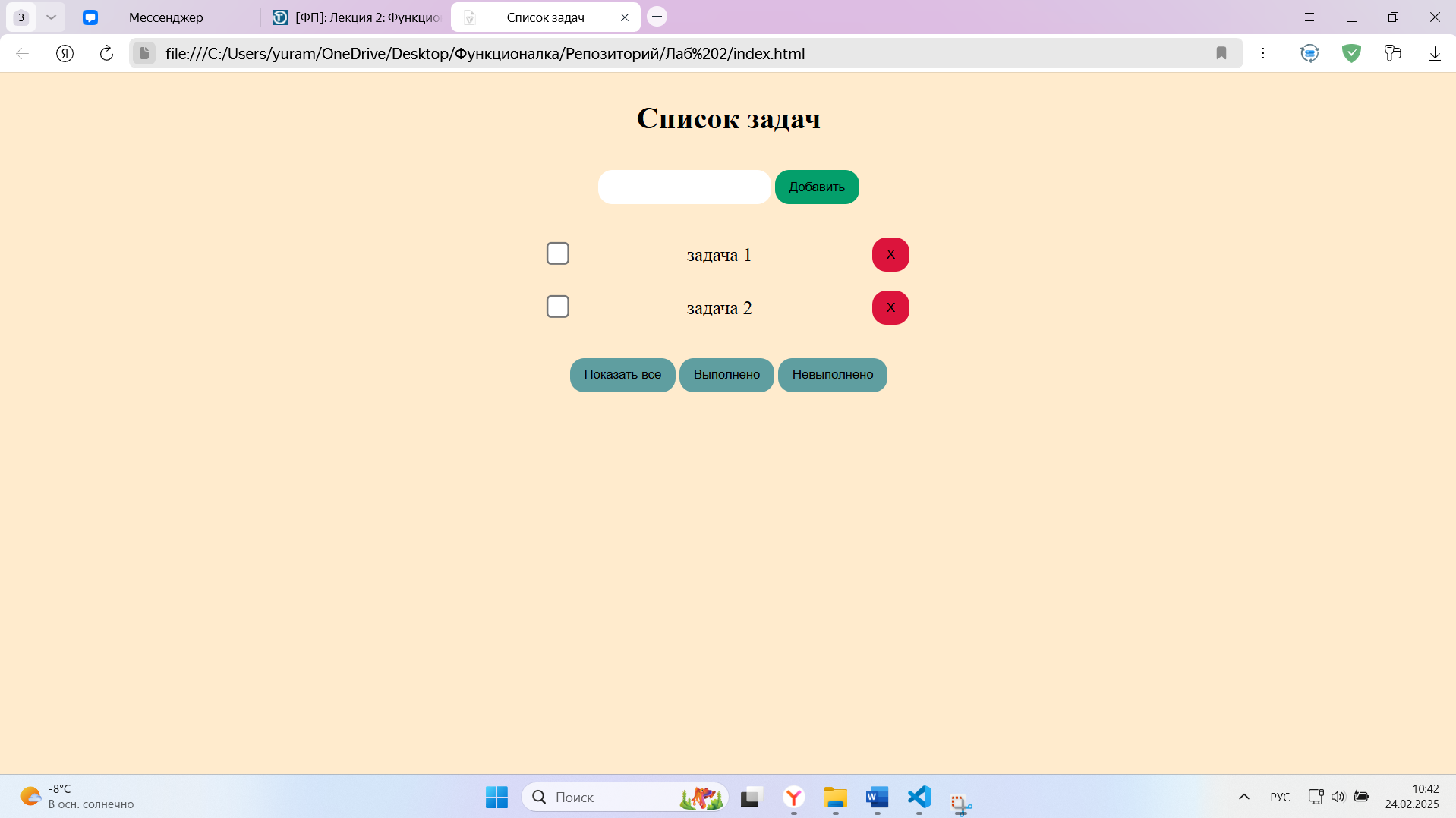
****

Рис. 2 – Выполнение программы