



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

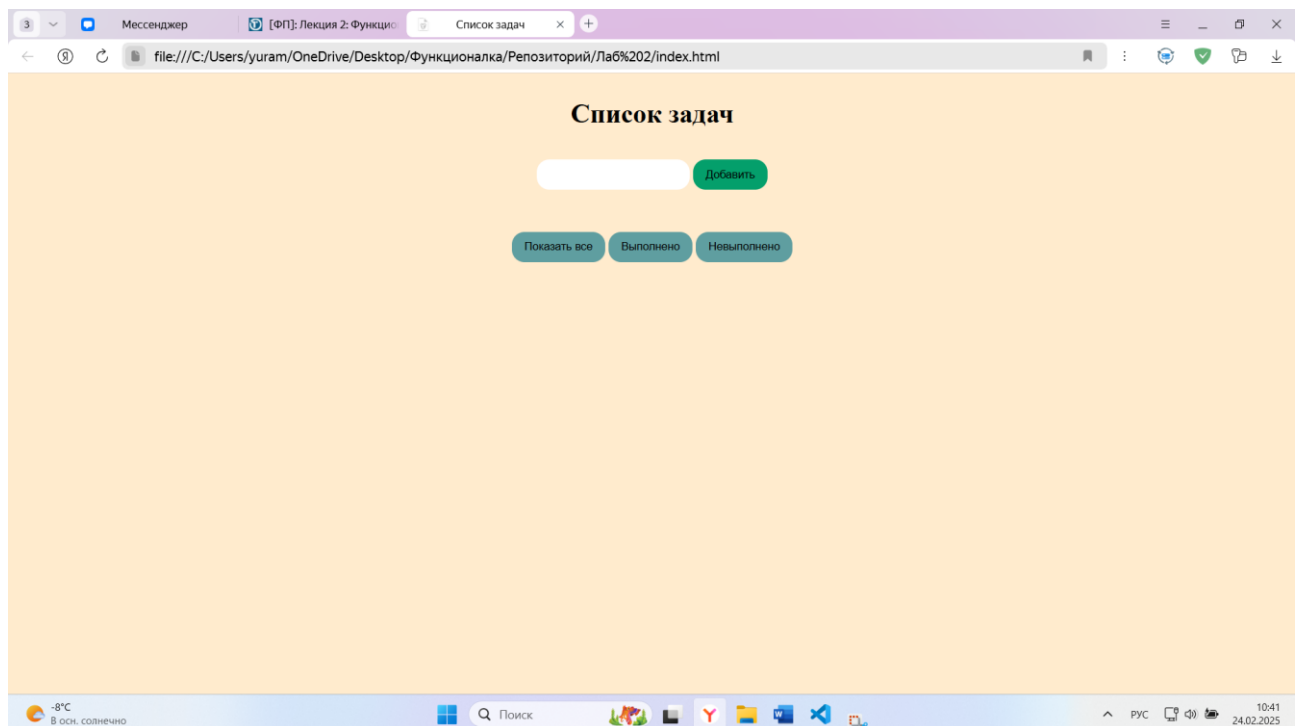


Рис. 1 – Выполнение программы

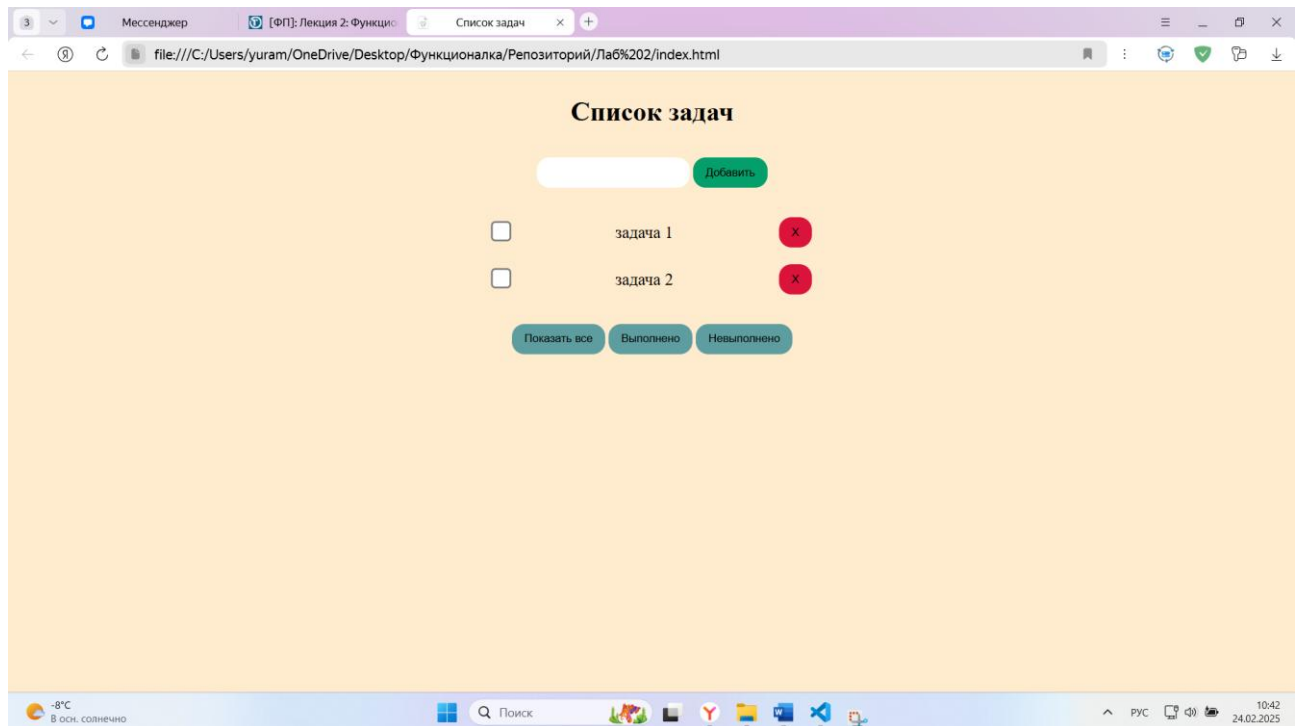


Рис. 2 – Выполнение программы