

Лабораторная работа № 17

Тема: Хранение и загрузка данных. Сохранение состояния интерфейса

Общее требование для всех вариантов:

Сохранение состояния интерфейса:

- Приложение должно сохранять данные в `localStorage` так, чтобы после перезагрузки страницы (`$F5$`) все внесенные изменения оставались на экране.
- При загрузке страницы должна быть проверка: если данных в хранилище нет, приложение должно отображать пустой список или значения по умолчанию.
- Использовать `JSON.stringify()` и `JSON.parse()`.

Разделение логики и отрисовки (UI):

- Создать отдельную функцию `render()`, которая отвечает только за создание HTML-элементов на основе текущего массива данных.
- Нельзя изменять DOM напрямую внутри функций обработки данных. Сначала меняем массив -> потом вызываем `render()`.

Использование функций:

- Логика работы с `localStorage` должна быть вынесена в отдельную функцию (например, `saveData()` или `syncStorage()`).
- Обработчики событий должны быть аккуратными и вызывать соответствующие функции.

Работа с массивом объектов:

- Данные должны храниться в виде массива объектов, например: `[{ id: 1, name: 'Книга', price: 500 }]`. Это приучает студентов к работе с реальными структурами данных.

Оформление (CSS):

- Приложение не должно выглядеть как голый текст.
- Обязательно наличие стилей для: формы ввода, карточек списка и кнопок удаления.
- Использовать `Flexbox` или `Grid` для выравнивания элементов.
- Визуальное выделение (например, через прозрачность или цвет), если элемент считается «выполненным» или «архивным».

Валидация (защита от дурака):

- Проверять, что пользователь не пытается добавить пустую строку.
- При парсинге данных из localStorage обеспечить проверку на null.

Вариант 1: «Менеджер личных закладок»

Задача: Создать страницу, куда пользователь может сохранять ссылки на полезные ресурсы.

Функционал:

- Форма с двумя полями: «Название сайта» и «Ссылка (URL)».
- Кнопка «Добавить».
- Список карточек с добавленными ссылками. Каждая ссылка должна открываться в новой вкладке.
- Кнопка «Удалить» у каждой закладки.

Сценарий проверки:

- Добавить 3 закладки.
- Перезагрузить страницу — закладки должны быть на месте.
- Удалить одну — после перезагрузки её быть не должно.

Вариант 2: «Трекер расходов (Simple Budget)»

Задача: Создать приложение для учета трат за день.

Функционал:

- Форма с двумя полями: «Название расхода» (например, «Обед») и «Сумма» (число).
- Кнопка «Добавить».
- Список всех записей под формой.
- Кнопка «Очистить всё» для удаления истории расходов.

Сценарий проверки:

- Ввести три расхода: 100, 200 и 500.
- Убедиться, что «Итого» показывает 800.
- Перезагрузить страницу — список и сумма 800 должны сохраниться.

Дополнительное задание: «Добавить кнопку переключения Темной/Светлой темы, выбор которой также сохраняется в localStorage».

 **Отчет должен содержать (см. образец):**

- номер и тему лабораторной работы;
- фамилию, номер группы студента и вариант задания;
- скриншоты окна VSC с исходным кодом программ;
- скриншоты с результатами выполнения программ;
- пояснения, если необходимо;
- выводы.

Отчеты в формате **pdf** отправлять на email:
colledge20education23@gmail.com