**Отчет по лабораторной работе №3**

**Тема:** Учет личных финансов

**Цель:** Ознакомиться с основами взаимодействия JS с DOM-деревом на основе веб-приложения для учета личных финансов.

**Инструкция по запуску проекта**

**1. Скачайте все три файла: index.html, script.js, style.css.**

**2. Поместите их в одну папку.**

**2. Откройте файл index.html в любом современном веб-браузере.**

Будьте внимательны, не все браузеры разрешаю подгружать JS скрипты в напрямую открытых html файлах.

**Описание лабораторной работы**

Данная лабораторная работа направлена на создание простого приложения для управления транзакциями. Приложение предоставляет возможность добавления новых транзакций, их отображение в таблице с возможностью удаления, а также подсчет общей суммы транзакций и отображение подробной информации о каждой транзакции.

**Краткая документация к проекту**

* Добавление транзакции: Заполните форму суммы, выберите категорию и введите краткое описание, затем нажмите кнопку "Добавить транзакцию".
* Удаление транзакции: Нажмите кнопку "Удалить" в строке таблицы, чтобы удалить соответствующую транзакцию.
* Отображение подробной информации: Нажмите на строку таблицы с транзакцией, чтобы увидеть полную информацию о ней в блоке подробного описания.
* Подсчет общей суммы: Общая сумма транзакций отображается в верхней части страницы.

**Примеры использования проекта с приложением скриншотов или фрагментов кода**

В данной работе примеры не нужны, т.к. достаточно посомтреть index.html в браузере.

**Ответы на контрольные вопросы**

**1. Каким образом можно получить доступ к элементу на веб-странице с помощью JavaScript?**

Самые популярные методы:

* document.getElementById()
* document.querySelector()
* document.querySelectorAll()

**2. Что такое делегирование событий и как оно используется для эффективного управления событиями на элементах DOM?**

Делегирование событий - это прием программирования, при котором обработчик события устанавливается на общем предке элементов, а не на каждом отдельном элементе. Когда событие происходит на потомке, оно всплывает до его предка, где и обрабатывается. Это позволяет эффективно управлять событиями на большом количестве элементов DOM, особенно если они создаются или удаляются динамически.

**3. Как можно изменить содержимое элемента DOM с помощью JavaScript после его выборки?**

Можно использовать свойство innerHTML или textContent.

**4. Как можно добавить новый элемент в DOM дерево с помощью JavaScript?**

Для добавления нового элемента в DOM дерево с помощью JavaScript используйте метод createElement() для создания нового элемента, затем используйте методы appendChild() или insertBefore() у родительского элемента, чтобы добавить его в нужное место в DOM.

**Список использованных источников**

* JavaScript Docs: <https://developer.mozilla.org/ru/docs/Web/JavaScript>
* GitHub Markdown Docs: <https://docs.github.com/en/get-started/writing-on-github/getting-started-with-writing-and-formatting-on-github/basic-writing-and-formatting-syntax>