Индивидуальная работа

Предмет: Java Script

Тема: DOM-дерево

Подготовила: Доцен Анна

Группа: IA2303

Проверил: преподаватель Nartea Nichita

Кишинев, 2024

Оглавление

**Теоретическая часть 3**

*Цель* 3

*Условия*3

**Практическая часть 4**

*Краткое описание реализации*4

**Ответы на контрольные вопросы** 6

**Источники** 6

**Теоретическая часть**

**Цель**

Ознакомиться с основами взаимодействия JS с DOM-деревом на основе веб-приложений для рассмотрения личных финансовых.

**Условия**

- *Настройка и структурирование проекта*

1. создать файлindex.html
2. создать файлscript.js
3. создать файлstyle.css
4. Подключить файл со стилями и js-кодом к HTML-странице.

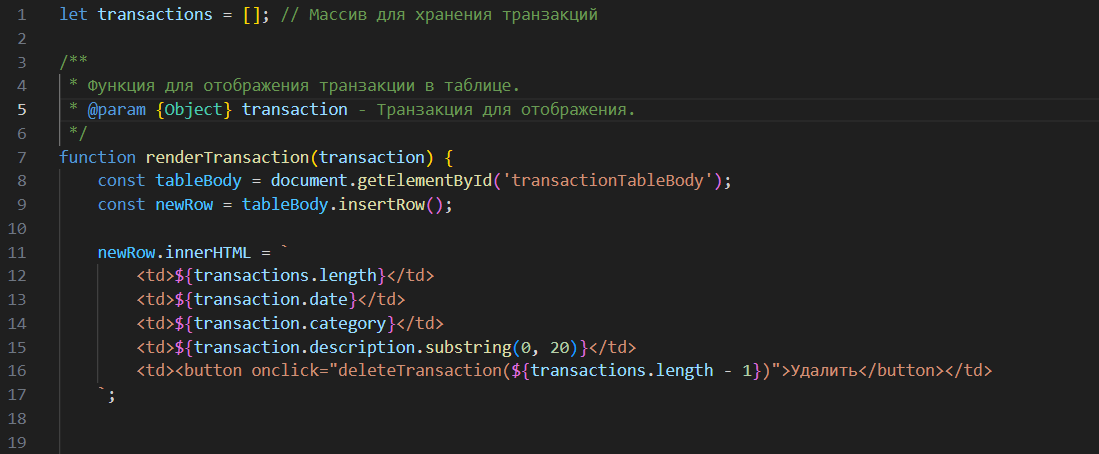
<link rel="stylesheet" href="style.css">

<script src="script.js"></script>

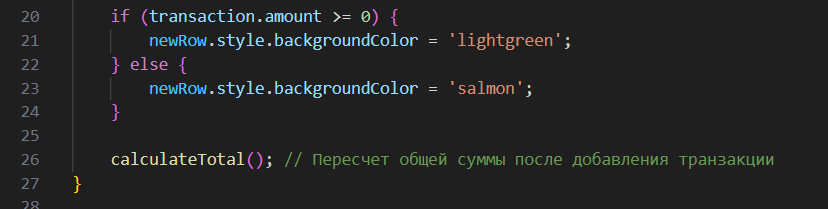
- *Отображение транзакций*

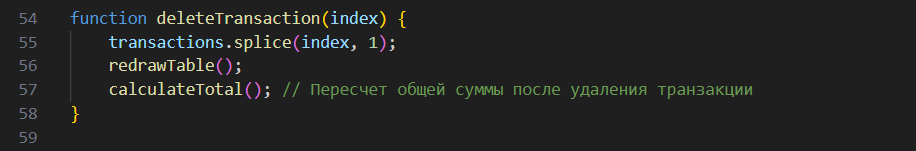
1. Создаем пустую таблицу, куда в перспективе будем записывать транзакций.
2. Таблица должна содержать следующие столбцы:
3. Идентификатор
4. Дата и время
5. Категория обращения
6. Краткое описание прохождения
7. Действие (кнопка удаления остатков)

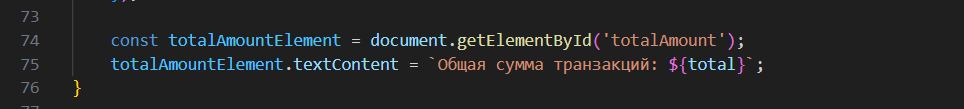




*- Подсчет суммы транзакции*







**Практическая часть**

**Краткое описание реализации**

Проект "Управление транзакциями» реализует простую веб-страницу для управления финансовыми операциями. Пользователи могут добавлять новые транзакции по дате, сумме, категории (доходы или расходы) и описанию. Добавленные транзакции отображаются на странице в виде таблицы, что позволяет пользователям просматривать и управлять своими финансовыми операциями.

**Основные функции.**

- Добавление транзакций.

1. Пользователь заполняет форму с данными о новой транзакции (дата, сумма, категория и описание) и отправляет ее.
2. Новая транзакция добавляется в массив транзакций.
3. Данные о новой транзакции отображаются в виде строк в таблице на странице.
4. Если сумма транзакции положительная, строка таблицы подсвечивается зеленым цветом, в противном случае - красным.
5. Удаление транзакций:

К каждой строке таблицы добавлена кнопка "Удалить".

При нажатии на кнопку "Удалить" соответствующая транзакция удаляется из массива транзакций, а строка таблицы удаляется со страницы.

1. После удаления общая сумма транзакции пересчитывается и обновляется.
2. Отображение итоговых сумм

На странице отображается общая сумма всех транзакций.

Общая сумма автоматически пересчитывается и обновляется при добавлении или удалении транзакции.

**Техническая реализация.**

1. HTML (index.html):
2. Структура страницы включает в себя форму для добавления новых транзакций, таблицу с транзакциями и контейнер, отображающий общую сумму.
3. Подключен файл script.js для логики JavaScript.
4. CSS (style.css):
5. Определяет стиль элементов страницы, таких как формы, таблицы, кнопки и контейнеры итогов.

**JavaScript (script.js):**

1. Создает массив транзакций для хранения объектов транзакций.

Реализуйте функции:

1. renderTransaction(): добавляет новую транзакцию в таблицу и выделяет строки цветом в соответствии со знаком суммы.
2. addTransaction(): обрабатывает отправку формы и добавляет новую транзакцию в массив транзакций.
3. deleteTransaction(): удаляет выбранную транзакцию из массива: удаляет выбранную транзакцию из массива и обновляет отображение таблицы.
4. redrawTable(): удаляет выбранные транзакции из массива и обновляет отображение таблицы: перерисовывает таблицу, содержащую удаленные транзакции.
5. calculateTotal(): пересчитать и отобразить общую сумму транзакций: пересчитывает и отображает общую сумму всех транзакций на странице.

**Использование.**

1. Откройте файл index.html в веб-браузере.
2. Заполните форму новой транзакции (укажите дату, сумму, выберите категорию и добавьте описание).
3. Нажмите на кнопку "Добавить транзакцию".
4. Проверьте добавленные транзакции в таблице.
5. Если вам нужно удалить транзакцию, нажмите кнопку "Удалить" в соответствующей строке таблицы.

Этот проект демонстрирует базовые навыки работы с HTML, CSS и JavaScript и создает интерактивную веб-страницу для управления финансовыми операциями.

**Ответы на контрольные вопросы**

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

Для доступа к элементам веб-страницы с помощью JavaScript используются методы DOM (Document Object Model):

- Через идентификаторы (ID): document.getElementById('elementId') позволяет получить элемент по его уникальному идентификатору.

- Через класс: document.getElementsByClassName('className') возвращает коллекцию элементов с определенным классом.

- Через теги: document.getElementsByTagName('tagName') возвращает коллекцию элементов с определенным тегом.

- Через CSS-селекторы (querySelector и querySelectorAll): document.querySelector('cssSelector') и document.querySelectorAll('cssSelector') позволяют выбирать элементы с помощью селекторов CSS.

1. Что такое управление событиями и как оно используется для эффективного управления событиями на элементах DOM?

Управление событиями в JavaScript позволяет присваивать функции обработчики определенных событий, происходящих с элементами DOM. Это позволяет элементам реагировать на действия пользователя, такие как клики, наведение курсора, ввод формы и т. д. Обработка событий придает интерактивность веб-страницам. Для назначения обработчика события следует использовать метод addEventListener() для указания типа события и функции обработчика.

1. Как можно изменить элементы элементов DOM с помощью JavaScript после его выбора?

Выбрав элемент DOM с помощью методов доступа (таких как getElementById или querySelector), его свойства и содержимое можно изменить с помощью JavaScript. Некоторые способы изменения элемента

Изменение текста: textContent = “new text”;

Изменение атрибутов: setAttribute(“attribute”, “value”);

Изменение стилей CSS: property = “value”;

Добавление и удаление классов: и element.

Изменение содержимого HTML: innerHTML = “новое содержимое HTML”;

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

Для добавления нового элемента в дерево DOM с помощью JavaScript используется комбинация методов:

Создание нового элемента: createElement(“tagName”) создает новый элемент с указанным тегом.

Назначение атрибутов и содержимого новому элементу при необходимости.

Добавление нового элемента в DOM с помощью методов:

Добавление в конец родительского элемента:

Добавление перед другим элементом: insertBefore(newElement, referenceElement);

Замена существующего элемента: replaceChild(newElement, oldElement);

**Источники**

1) <https://github.com/MSU-Courses/javascript_typescript/blob/main/lab/LI3/JS03.md>

2) <https://ravesli.com/put-javascript-in-html/>

3) <https://mixelte.github.io/pages/html-css-js/>