**Отчет по Лабораторной работе 1: HTML и CSS Верстка Калькулятора**

**Выполнил студент:** Ованнисян Петрос Абгарович

**Группа:** СГН3-64Б

**Обзор:**

Данный отчет посвящен разбору HTML-структуры и CSS-стилей, используемых для создания визуального представления калькулятора в файле calculator.html. Целью является анализ структуры элементов, используемых классов и способов стилизации, применяемых для достижения желаемого внешнего вида.

**HTML Структура:**

HTML-структура калькулятора состоит из следующих основных элементов:

• <div class="calculator">: Основной контейнер, оборачивающий весь калькулятор. Он задает ширину, центрирование и базовые стили.

• <input type="text" id="display" ...>: Текстовое поле, предназначенное для отображения введенных чисел, операторов и результата вычислений. Атрибут readonly предотвращает ручной ввод данных в это поле.

• <div class="calculator-grid">: Контейнер для кнопок калькулятора. Используется CSS Grid Layout для организации кнопок в сетку.

• <button onclick="...">: Каждая кнопка калькулятора (цифры, операторы, AC, =). Атрибут onclick используется для вызова JavaScript-функций при нажатии на кнопку. Кнопки с классом span-two занимают две колонки в сетке.

**CSS Стилизация:**

Для стилизации элементов калькулятора используются следующие CSS-классы и свойства:

• .calculator:

\* width: 320px;: Устанавливает ширину калькулятора.

\* margin: 20px auto;: Центрирует калькулятор по горизонтали.

\* border: 1px solid #ddd;: Добавляет тонкую серую рамку.

\* border-radius: 12px;: Закругляет углы калькулятора.

\* overflow: hidden;: Обрезает содержимое, выходящее за границы контейнера.

\* box-shadow: 0 4px 8px rgba(0, 0, 0, 0.15);: Добавляет мягкую тень.

\* background-color: #f8f8f8;: Устанавливает светло-серый фон.

• .calculator input[type="text"]:

\* width: 100%;: Занимает всю ширину контейнера.

\* padding: 20px;: Добавляет внутренние отступы.

\* font-size: 32px;: Устанавливает размер шрифта.

\* text-align: right;: Выравнивает текст по правому краю.

\* border: none;: Убирает рамку.

\* background-color: #fff;: Устанавливает белый фон.

\* color: #222;: Устанавливает темный цвет текста.

\* box-sizing: border-box;: Учитывает padding и border в ширине.

\* outline: none;: Убирает обводку при фокусе.

• .calculator-grid:

\* display: grid;: Включает CSS Grid Layout.

\* grid-template-columns: repeat(4, 1fr);: Создает 4 колонки одинаковой ширины.

\* gap: 1px;: Добавляет отступ между ячейками сетки.

\* background-color: #ddd;: Устанавливает серый фон для сетки.

• .calculator-grid button:

\* padding: 20px;: Добавляет внутренние отступы.

\* font-size: 24px;: Устанавливает размер шрифта.

\* border: none;: Убирает рамку.

\* background-color: #fff;: Устанавливает белый фон для кнопок.

\* color: #333;: Устанавливает темный цвет текста.

\* cursor: pointer;: Меняет курсор при наведении.

\* transition: background-color 0.15s ease-in-out, transform 0.1s ease;: Добавляет анимацию при наведении и нажатии.

\* outline: none;: Убирает обводку при фокусе.

\* box-shadow: none;: Убирает тень.

• .calculator-grid button:hover:

\* background-color: #f0f0f0;: Меняет цвет фона при наведении.

• .calculator-grid button:active:

\* background-color: #e0e0e0;: Меняет цвет фона при нажатии.

\* transform: scale(0.95);: Уменьшает размер кнопки при нажатии.

• .calculator-grid button:focus:

\* box-shadow: 0 0 0 3px rgba(52, 152, 219, 0.3);: Добавляет подсветку при фокусе.

• .calculator-grid button.span-two:

\* grid-column: span 2;: Занимает две колонки сетки.

• .calculator-grid button:first-child:

\* background-color: #e74c3c;: Устанавливает красный фон для кнопки AC.

\* color: white;: Устанавливает белый цвет текста.

• .calculator-grid button:first-child:hover:

\* background-color: #c0392b;: Меняет цвет фона при наведении.

**Анализ и Выводы:**

• Структура HTML проста и понятна, что облегчает понимание организации элементов калькулятора.

• Использование CSS Grid Layout позволяет легко и эффективно создавать сетку кнопок.

• Применение классов позволяет контролировать стили отдельных элементов и групп элементов.

• Добавление переходов и анимаций улучшает пользовательский опыт, делая взаимодействие с калькулятором более приятным.

• Код хорошо структурирован и отформатирован, что облегчает его чтение и поддержку.

**Рекомендации:**

• Рассмотреть использование CSS переменных для управления цветами и размерами, что упростит изменение внешнего вида калькулятора.

• Вынести CSS в отдельный файл для улучшения организации и переиспользования стилей.

• Добавить медиа-запросы для адаптации калькулятора к разным размерам экранов (responsive design).