|  |  |
| --- | --- |
| Gerb-BMSTU_01 | **Министерство науки и высшего образования Российской Федерации**  **Федеральное государственное бюджетное образовательное учреждение**  **высшего образования**  **«Московский государственный технический университет**  **имени Н.Э. Баумана**  **(национальный исследовательский университет)»**  **(МГТУ им. Н.Э. Баумана)** |

ФАКУЛЬТЕТ **Информатика и системы управления**

КАФЕДРА **Компьютерные системы и сети (ИУ6)**

НАПРАВЛЕНИЕ ПОДГОТОВКИ: **09.03.01 Информатика и Вычислительная техника**

**Отчет**

**по лабораторной работе 4**

**Дисциплина: языки интернет-программирования**

**Вариант №15**

Студент гр. **ИУ6-33Б \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_** В.К. Залыгин

(Подпись, дата) (И.О. Фамилия)

Преподаватель  **\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_** В.Д. Шульман

(Подпись, дата) (И.О. Фамилия)

Москва, 2023

**Цель**

Научиться применять стек html+css+js для выполнения бизнес-задач. В качестве примера разработать страницу магазина.

**Задание**

Разработать страницу для оформления заказа товаров.

1. Реализовать форму для ввода наименования и стоимости. При нажатии на кнопку «Добавить», добавить введенные значения в таблицу.
2. Сделать кнопку расчёта стоимости заказа.
3. Поставить обработчик на изменение фона ячеек таблицы со стоимостью в тот момент, когда над ними находится указатель мыши.
4. Подготовить страницу, содержащую таблицу товаров и их стоимости. На базе этой страницы обеспечить возможность выбора строк и добавления их в таблицу формирования заказа.

Примечание: в качестве усложненного варианта задания допустимо использовать библиотеки jQuery и Dojo.

**Выполнение**

HTML:

<!DOCTYPE html>

<html lang="ru">

    <head>

        <meta charset="utf-8"/>

        <link href="./index.css" rel="stylesheet"/>

    </head>

    <body>

        <div class="container form">

            <h3>Добавление товара</h3>

            <div class="horiz">

                <div class="vert">

                    <span class="el">Наименование</span>

                    <span class="el">Стоимость, ₽</span>

                </div>

                <div class="vert">

                    <input class="el" id="name" type="text" alt="Наименование"/>

                    <input class="el" id="price" type="number" alt="Стоимость" value="0"/>

                </div>

            </div>

            <div class="vert">

                <button class="el button" onclick="addToTable()">Добавить в таблицу</button>

            </div>

        </div>

        <div class="container table horiz">

            <div class="table-column vert" id="column1">

                <div class="table-header">Товар</div>

            </div>

            <div class="table-column vert" id="column2">

                <div class="table-header">Стоимость</div>

            </div>

        </div>

        <div class="list container vert" id="showcase">

            <h2>Каталог растений</h2>

        </div>

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

    </body>

</html>

CSS:

body {

    display: flex;

    background-color: beige;

}

main {

    display: flex;

}

.container {

    background-color: bisque ;

    border-radius: 5px;

    padding: 10px;

    margin: 5px;

    height: fit-content;

}

.form {

    display: flex;

    flex-direction: column;

}

.horiz {

    display: flex;

    flex-direction: row;

}

.vert {

    display: flex;

    flex-direction: column;

}

.el {

    margin: 5px;

}

.button:hover {

    background-color: greenyellow;

    transition-duration: 0.3s;

}

.table {

    border: 2px solid;

    min-width: 400px;

    justify-content: space-around;

    border-radius: 5px;

    background-color: bisque;

}

.table-header {

    padding: 5px;

    font-style: oblique;

}

.table-header:hover {

    background-color: darkgray;

    transition-duration: 0.3s;

}

.table-column {

    width: 50%;

}

.table-element {

    padding: 5px;

    margin: 0px;

    justify-content: center;

}

.table-element:hover {

    background-color: aliceblue;

    transition-duration: 0.3s;

}

.list {

    display: flex;

    flex-direction: row;

    flex-wrap: wrap;

    width: 50%;

}

.card {

    background-color: azure;

    border-radius: 5px;

    margin: 5px;

    width: 150px;

    height: 150px;

    text-align: center;

}

.card:hover {

    background-color: white;

    transition-duration: 0.3s;

}

img {

    height: 70%;

    width: 70%;

    display: block;

    margin-left: auto;

    margin-right: auto;

}

JS:

const makeTableElement = (value) => {

    const el = document.createElement("div");

    el.className = "table-element";

    el.innerText = value;

    return el;

};

const appendToTable = (col1, col2) => {

    const column1 = document.getElementById("column1");

    const column2 = document.getElementById("column2");

    column1.appendChild(col1);

    column2.appendChild(col2);

};

const addToTable = (nameValue, priceValue) => {

    if (nameValue === undefined && priceValue === undefined) {

        nameValue = document.getElementById("name").value;

        priceValue = document.getElementById("price").value;

    }

    const nameColumn = makeTableElement(nameValue);

    const priceColumn = makeTableElement(priceValue);

    appendToTable(nameColumn, priceColumn);

};

const makeCard = (name, price) => {

    const card = document.createElement("div");

    const img = document.createElement("img");

    const h3 = document.createElement("h3");

    card.className = "card";

    card.appendChild(img);

    card.appendChild(h3);

    img.src = `./resources/${name}.webp`;

    h3.textContent = `${name} ${price}₽`

    card.onclick = (e) => {

        addToTable(name, price);

    }

    return card;

};

const clearContent = (element) => {

    while(element.firstChild && element.removeChild(element.firstChild));

};

const renderShowcase = () => {

    const data = [

        {name: "Роза", price: "284"},

        {name: "Хамедорея", price: "455"},

        {name: "Орхидея", price: "730"},

        {name: "Цикламен", price: "288"},

        {name: "Хризантема", price: "160"},

    ];

    const showcase = document.getElementById("showcase");

    clearContent(showcase);

    for (const flower of data) {

        showcase.appendChild(makeCard(flower.name, flower.price));

    }

};

renderShowcase();

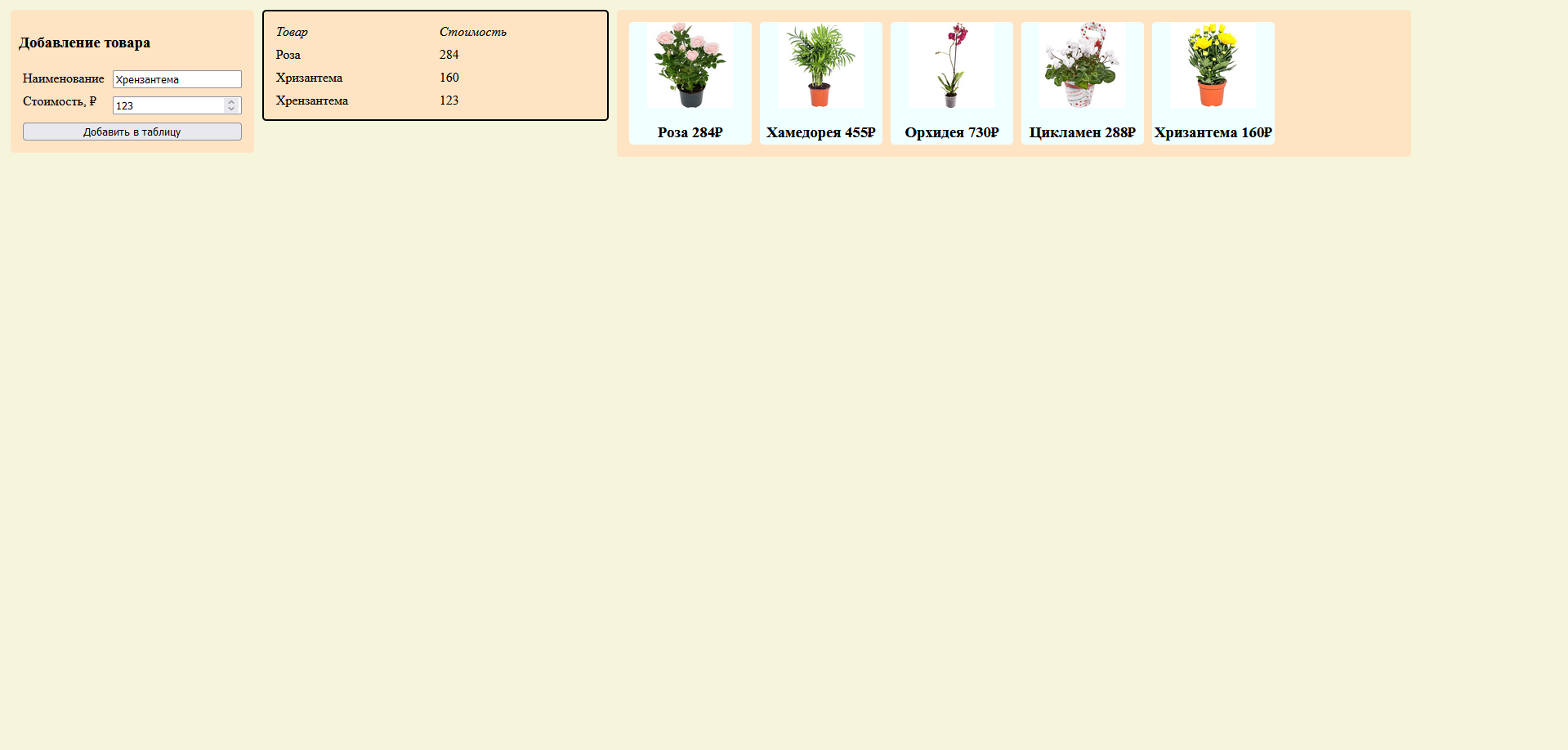


Рисунок 1 - пример работы

**Вывод**

Получен опыт применения стека html+css+js для выполнения бизнес-задач. В качестве примера разработана страница магазина.