Министерство науки и высшего образования Российской Федерации

**Федеральное государственное бюджетное образовательное учреждение высшего образования**

«Владимирский государственный университет

имени Александра Григорьевича и Николая Григорьевича Столетовых»

**(ВлГУ)**

**Кафедра информационных систем и программной инженерии**

Лабораторная работа №1

по дисциплине «Базовые информационные технологии»

Тема: «Язык программирования JavaScript. Настройка окружения, введение и программирование.»

Выполнил:

студент гр. ПРИ-123

А.Ц. Нямаа

Принял:

Ст. пр. кафедры ИСПИ

О.Н. Шамышев

Владимир, 2025 г.

ЦЕЛЬ РАБОТЫ:

Установить необходимые инструменты, плагины и утилиты для работы с JavaScript. Создать динамическую HTML страницу с некоторым функционалом. Решить простейшие задачи на программирование.

ЗАДАНИЕ 1 (вариант 12).

Установить некоторые базовые инструменты для работы с JavaScript (Google Chrome, NodeJS, Visual Studio Code). Добавить несколько плагинов в VS Code для поддержания чистоты написанного кода. Установить утилиту browser-sync и проверить ее работу.

ВЫПОЛНЕНИЕ РАБОТЫ:

В начале выполнения данной работы, я приступил к скачиванию базовых инструментов для работы с JavaScript. Стоит отметить, что инструменты Google Chrome и Visual Studio Code уже были установлены, поэтому осталось поставить только Node.js. Для загрузки использовал установщик, скаченный с официального сайта разработчика.

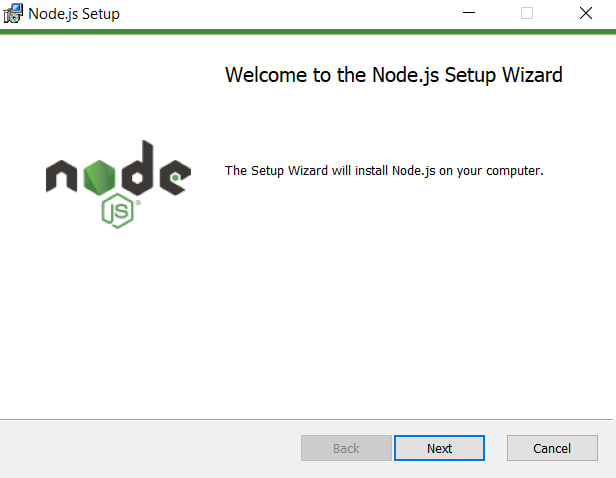


Рисунок 1. Установка Node.js.

После установки проверил версию, введя в командную строку node -v.

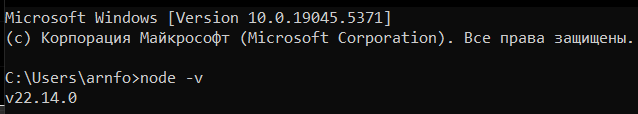


Рисунок 2. Проверка версии Node.js.

Далее приступил к скачиваю плагинов, которые необходимы для поддержания чистоты кода. В Visual Studio Code открыл вкладку Extensions, где нашел плагин Prettier и скачал его. Также изменил некоторые настройки, например, форматирование кода каждый раз, когда происходит сохранение файла.

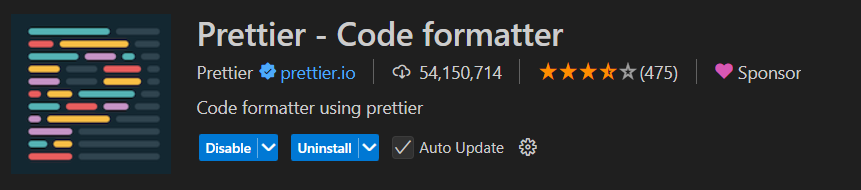


Рисунок 3. Скачивание плагина Prettier.

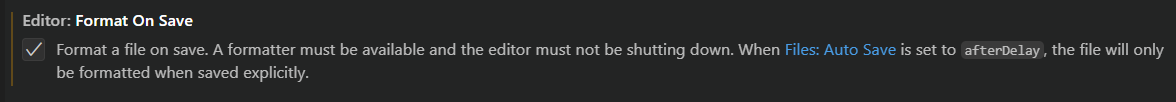


Рисунок 4. Настройка плагина Prettier.

Также тем же методом установил плагин ESLint и для того, чтобы он заработал, изменил некоторые настройки в JSON файле.

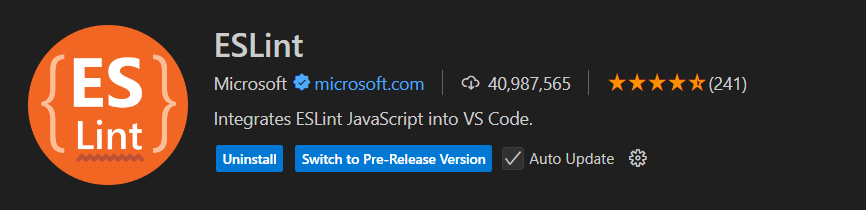


Рисунок 5. Скачивание плагина ESLint.

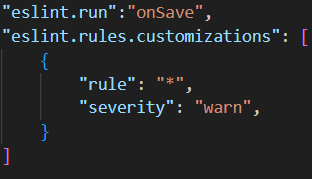


Рисунок 6. Настройка плагина ESLint.

Установил утилиту browser-sync, используя менеджер пакетов Node.js. Для этого открыл командную строку и выполнил следующую команду: *npm install -g browser-sync*.

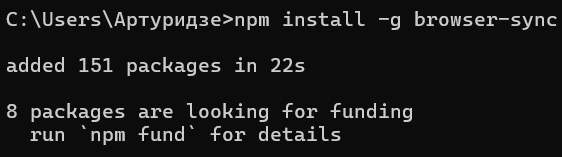


Рисунок 7. Установки утилиты browser-sync.

Так как для тестирование утилиты был использован статический сайт, было необходимо воспользоваться серверным режимом. Для этого я перешел в каталог, где находятся html и css файлы сайта и ввел следующую команду: *browser-sync start --server --files "\*"*.

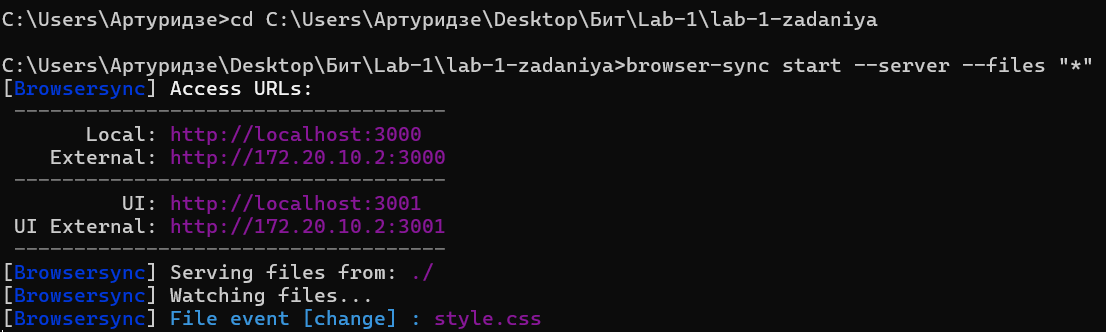


Рисунок 8. Тестирование утилиты browser-sync.

ЗАДАНИЕ 2 (вариант 12).

Выполнить примеры из методички, а также создать одностраничный сайт на тему «Часы». На сайте должны осуществляться следующие действия:

1. Изменение фона при наведении указателя мыши
2. Считываться данные с формы и совершаться простые арифметические действия.

ВЫПОЛНЕНИЕ РАБОТЫ:

Выполняя данную работу, я переписал примеры, данные в методичке, и разобрался в том, как работает код.

Листинг примера 1:

<!DOCTYPE html>

<html>

<head>

    <title>Dynamika</title>

    <style>

    body {

        background-color: white;

    }

    .hover-background {

        background-color: red;

    }

    </style>

</head>

<body>

    <div id="myDiv">Это див</div>

    <script>

    var myDiv = document.getElementById("myDiv");

    myDiv.addEventListener("mouseover", function () {

        myDiv.classList.add("hover-background");

    });

    myDiv.addEventListener("mouseout", function () {

        myDiv.classList.remove("hover-background");

    });

    </script>

</body>

</html>



Рисунок 9. Выполнение кода из примера 1.

Листинг примера 2:

<!DOCTYPE html>

<html>

<head>

    <title>Dynamika</title>

</head>

<body>

    <input type="number" id="num1" placeholder="Число 1">

    <input type="number" id="num2" placeholder="Число 2">

    <button onclick="performCalculation()">Посчитать</button>

    <p id="result"></p>

    <script>

    function performCalculation() {

        var num1 = parseInt(document.getElementById("num1").value);

        var num2 = parseInt(document.getElementById("num2").value);

        var result = num1 + num2;

        document.getElementById("result").innerHTML = "Результат: " + result;

    }

    </script>

</body>

</html>

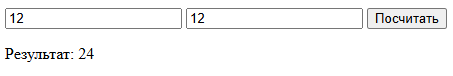


Рисунок 10. Выполнение кода из примера 2.

Листинг index.html:

<!DOCTYPE html>

<html lang="en">

<head>

    <meta charset="UTF-8">

    <meta name="viewport" content="width=device-width, initial-scale=1.0">

    <link rel="preconnect" href="https://fonts.googleapis.com">

    <link rel="preconnect" href="https://fonts.gstatic.com" crossorigin>

    <link href="https://fonts.googleapis.com/css2?family=Overpass:ital,wght@0,100..900;1,100..900&display=swap" rel="stylesheet">

    <title>Clock Shop</title>

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

</head>

<body>

    <header>

        <div class="leftText">

            <h1> ДИЗАЙНЕРСКИЙ ЧАСЫ </h1>

            <p> ПОЛНОСТЬЮ РУЧНОЕ ИЗГОТОВЛЕНИЕ </p>

            <button class="buyHeader buttonToChange" id="openModel"> КУПИТЬ </button>

            <dialog class="dialogMain">

                <form class="orderForm">

                    <h2>Оформление заказа</h2>

                    <p><label for="name">Имя:</label>

                    <input type="text" id="name" name="name" placeholder="Имя" required></p>

                    <p><label for="secondName">Фамилия:</label>

                    <input type="text" id="secondName" name="secondName" placeholder="Фамилия" required></p>

                    <p><label for="thirdName">Отчество:</label>

                    <input type="text" id="thirdName" name="thirdName" placeholder="Отчество"></p>

                    <p><label for="address">Адрес доставки:</label>

                    <input type="text" id="address" name="address" placeholder="Ул. Пушкино, д. 2, кв. 1" required></p>

                    <p><label for="quantity">Количество:</label>

                    <input type="number" id="quantity" name="quantity" placeholder="1" min="1" required></p>

                    <p><label for="promo">Промокод:</label>

                    <input type="text" id="promo" placeholder="\*Промокод\*" name="promo"></p>

                    <p class="priceHeader" data-price="900000">Цена: 900 000$</p>

                    <input type="submit" value="Заказать!">

                </form>

                <button id="closeModel" class="closeModel">&#x2715 </button>

            </dialog>

        </div>

        <div class="rightText">

            <p> JACOB & CO <br>EPIC X TOURBILON BAGUETTE <br>900.000$</p>

            <img src="imgs/mainPhotoWatches.png">

        </div>

    </header>

    <main>

        <div class="textInMain">

            <p>ХИТЫ</p>

            <p>ПОСЛЕДНИЕ ПОСТУПЛЕНИЯ</p>

        </div>

        <div class="productsList">

            <div class="productUnit">

                <img src="imgs/appleulta.png">

                <p>Apple watch ulta 2</p>

                <p>700$</p>

                <button class="buttonUnit"> Купить </button>

            </div>

            <div class="productUnit">

                <img src="imgs/rolexDayDate36.png">

                <p>Rolex Day-Date 36</p>

                <p>117.250$</p>

                <button class="buttonUnit"> Купить </button>

            </div>

            <div class="productUnit">

                <img src="imgs/KingSeiko.png">

                <p>King Seiko Automatic</p>

                <p>2.000$</p>

                <button class="buttonUnit"> Купить </button>

            </div>

            <div class="productUnit">

                <img src="imgs/PatekPhilippe.png">

                <p>Paket Philippe 5270/1R</p>

                <p>230.000$</p>

                <button class="buttonUnit"> Купить </button>

            </div>

            <div class="productUnit">

                <img src="imgs/rolexDateJust31.png">

                <p>Rolex DateJust 31</p>

                <p>51.900$</p>

                <button class="buttonUnit"> Купить </button>

            </div>

            <div class="productUnit">

                <img src="imgs/PatekPhilippeBlackWhite.png">

                <p>Patek Philippe 6300/400G</p>

                <p>9.000.000$</p>

                <button class="buttonUnit"> Купить </button>

            </div>

            <div class="productUnit">

                <img src="imgs/appleseries10.png">

                <p>Apple watch series 10</p>

                <p>450$</p>

                <button class="buttonUnit"> Купить </button>

            </div>

            <div class="productUnit">

                <img src="imgs/Jacob&Co.png">

                <p>Jacob&Co Palatial Classic</p>

                <p>16.500$</p>

                <button class="buttonUnit"> Купить </button>

            </div>

        </div>

    </main>

    <script src="script.js"></script>

</body>

</html>

Листинг style.css:

\* {

    margin: 0;

    padding: 0;

    overflow-x: hidden;

    font-family: "Overpass", serif;

    cursor: default;

}

button {

    cursor: pointer;

}

.changebackground {

    background-color: #f2f2f2 !important;

    color: black !important;

    box-shadow: 0px 5px 20px 8px rgba(34, 60, 80, 0.17) !important;

}

header {

    background-color: #f2f2f2;

    width: 100%;

    display: flex;

    height: 100vh;

}

header div {

    width: 50%;

    display: flex;

    flex-direction: column;

    align-items: center;

    justify-content: center;

    text-align: center;

}

.leftText > \* {

    margin: 0.75% 0;

}

.leftText h1 {

    font-size: 40px;

    font-weight: 700;

}

.leftText p {

    font-size: 20px;

}

.rightText img {

    width: 55%;

}

.rightText p {

    color: #c5c5c5;

    font-weight: 600;

}

.buyHeader {

    padding: 1.5% 4%;

    text-align: center;

    border: none;

    color: white;

    font-size: 20px;

    border-radius: 15px;

    background-color: black;

}

.dialogMain {

    margin: auto;

    top: 0;

    left: 0;

    right: 0;

    bottom: 0;

    width: 70%;

    height: 80%;

    position: absolute;

    border-radius: 20px;

    border: none;

    box-shadow: 0px 5px 20px 15px rgba(34, 60, 80, 0.17);

}

.changehtml {

    overflow: hidden;

}

.closeModel {

    position: absolute;

    right: 2%;

    top: 2%;

    border: none;

    background-color: white;

    font-size: 24px;

}

.orderForm {

    display: flex;

    flex-direction: column;

    gap: 17.5px;

    padding: 20px;

    width: 30%;

    margin: auto;

    font-family: "Overpass", sans-serif;

}

.orderForm h2 {

    text-align: center;

    margin: 5% 0;

}

.orderForm label {

    display: block;

    font-weight: bold;

}

.orderForm input {

    padding: 10px;

    border-radius: 8px;

    border: 1px solid #ccc;

    font-size: 16px;

}

.orderForm input[type="submit"] {

    background-color: black;

    color: white;

    border: none;

    cursor: pointer;

    font-weight: bold;

    margin-top: 10px;

    transition: 0.3s ease;

}

.orderForm input[type="submit"]:hover {

    background-color: #333;

}

input:not([type="submit"]) {

    width: 80%;

    cursor: text;

}

main {

    width: 100%;

    margin-top: 1.5%;

}

.textInMain {

    margin: 0 3%;

    font-weight: bold;

}

.textInMain p:first-of-type {

    font-size: 15px;

}

.textInMain p:last-of-type {

    font-size: 25px;

}

.productsList {

    margin: 0.5% 2%;

    display: grid;

    grid-template-columns: repeat(4, 1fr);

}

.productUnit {

    text-align: center;

    margin: 4% 3%;

    padding: 7% 0;

    box-shadow: 0px 5px 10px 2px rgba(34, 60, 80, 0.17);

    border-radius: 10px;

}

.productUnit \* {

    margin: 1% 0;

}

.productUnit img {

    width: 70%;

}

.buttonUnit {

    border: none;

    color: white;

    padding: 2% 3.5%;

    font-weight: 500;

    border-radius: 7px;

    background-color: black;

}

Листинг script.js:

// Смена цвета кнопки при наведении

const button = document.body.querySelector(".buttonToChange");

button.addEventListener("mouseover", function () {

    button.classList.add("changebackground");

});

button.addEventListener("mouseout", function () {

    button.classList.remove("changebackground");

});

// Модальное окно

const dialog = document.querySelector(".dialogMain");

const html = document.querySelector("html");

document.querySelector("#openModel").onclick = function () {

    dialog.showModal();

    html.classList.add("changehtml");

};

document.querySelector("#closeModel").onclick = function () {

    dialog.close();

    html.classList.remove("changehtml");

};

//Изменения цены

const formElement = document.querySelector(".orderForm");

const quantity = document.querySelector("#quantity");

const promo = document.querySelector("#promo");

const priceHeader = document.querySelector(".priceHeader");

const price = parseInt(

    document.querySelector(".priceHeader").getAttribute("data-price")

);

const promoList = {

    "10procentSaleWOW": 0.9,

    "15yearShop": 0.85,

};

function updatePrice() {

    let quantityVal = 1;

    if (quantity.value > 1) {

        quantityVal = quantity.value;

    }

    let discount = 1;

    if (promo.value in promoList) {

        discount = promoList[promo.value];

    }

    let finalPrice = price \* quantityVal \* discount;

    priceHeader.textContent = `Цена: ${finalPrice.toLocaleString()}$ (${quantityVal} шт, скидка ${Math.round(

        (1 - discount) \* 100

    )}%)`;

}

quantity.addEventListener("input", updatePrice);

promo.addEventListener("input", updatePrice);

// Отправка формы

formElement.addEventListener("submit", () => {

    const formData = new FormData(formElement);

    const name = formData.get("name");

    const surname = formData.get("secondName");

    const thirdName = formData.get("thirdName");

    const address = formData.get("address");

    alert(

        `Спасибо, ${name} ${surname} ${thirdName}!\nВаш заказ на адрес - ${address} в количестве ${quantity.value}шт будет исполнен в ближайшее время!`

    );

});

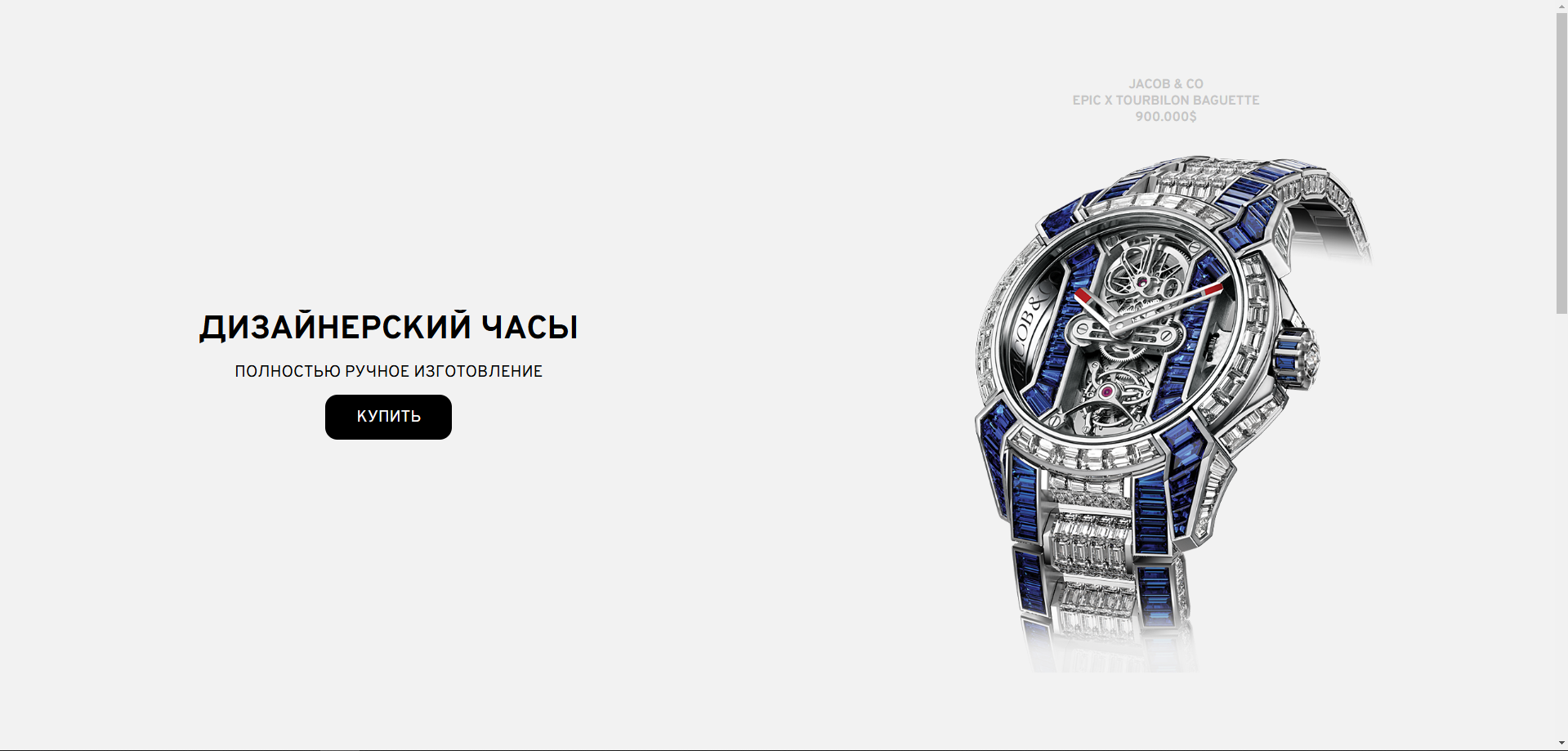


Рисунок 11. Сайт «Часы».

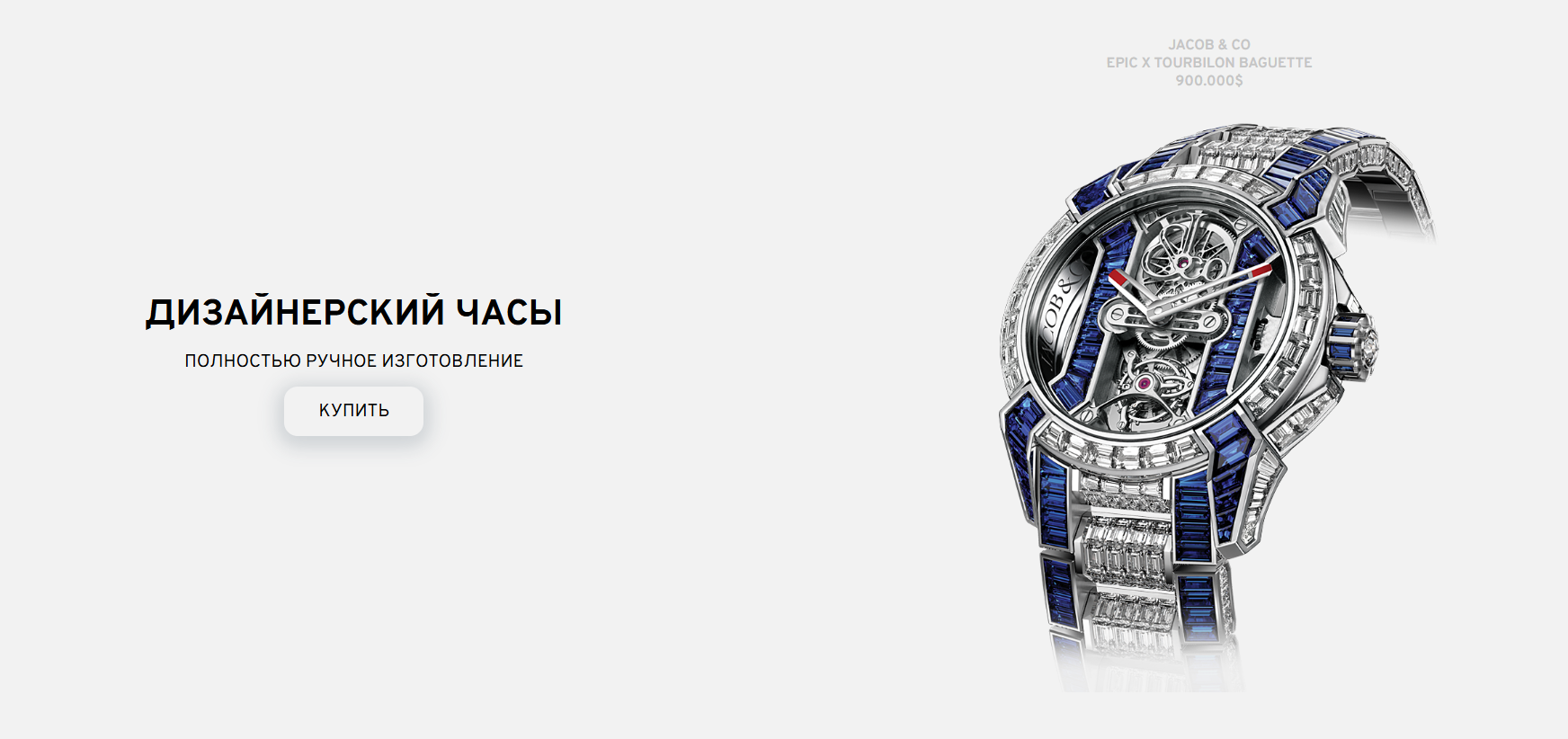


Рисунок 12. Изменение фона кнопки при наведении.

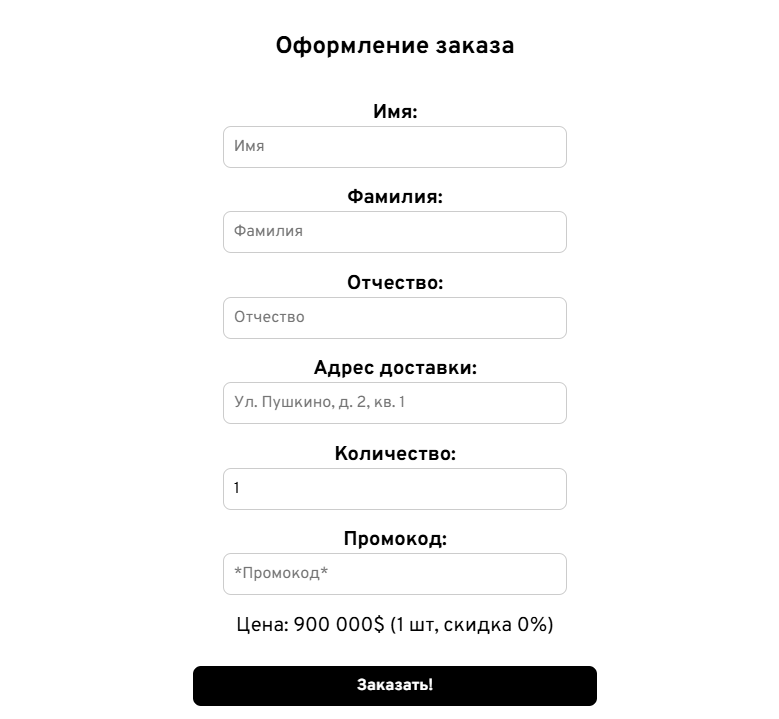


Рисунок 13. Форма оформления заказа.

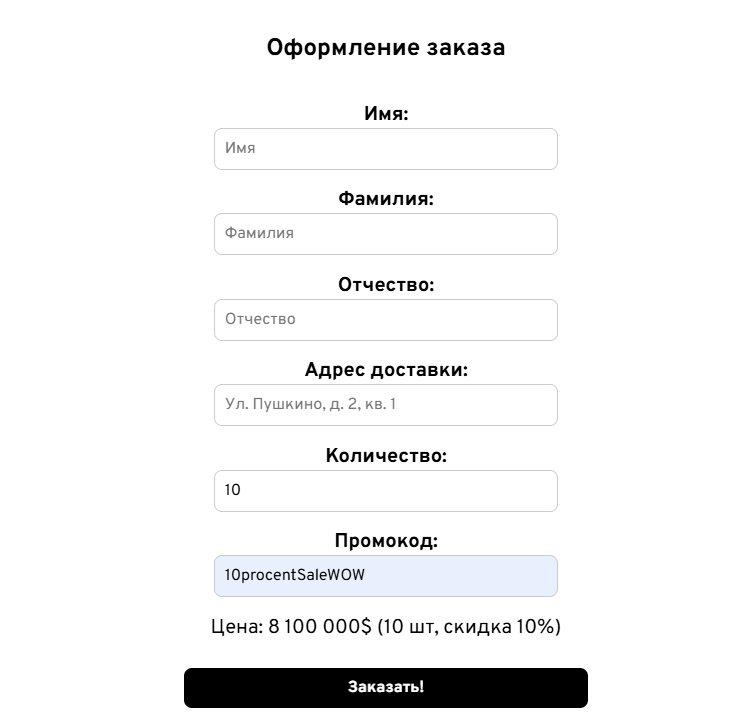


Рисунок 14. Выполнения простых арифметических вычислений.

ЗАДАНИЕ 3 (вариант 12).

Создать одностраничный сайт на тему «Часы». На сайте должны осуществляться следующие действия:

1. Используя переменны, которые вы считаете с формы, и уловные операторы (“if” и “?”), произвести с ними какое-либо действие;
2. Используя циклы (while…do, do…while, for), считать 2 любых интервала и вывести на этом промежутки любые числа равные какому-либо условию;
3. Используя функции, задать переменные (локально и внутренне) и произвести какое-либо действие и вернуть значение.

ВЫПОЛНЕНИЕ РАБОТЫ:

Листинг index.html:

<!DOCTYPE html>

<html lang="en">

<head>

    <meta charset="UTF-8">

    <meta name="viewport" content="width=device-width, initial-scale=1.0">

    <link rel="preconnect" href="https://fonts.googleapis.com">

    <link rel="preconnect" href="https://fonts.gstatic.com" crossorigin>

    <link href="https://fonts.googleapis.com/css2?family=Overpass:ital,wght@0,100..900;1,100..900&display=swap" rel="stylesheet">

    <title>Clock Shop</title>

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

</head>

<body>

    <header>

        <div class="leftText">

            <h1> ДИЗАЙНЕРСКИЙ ЧАСЫ </h1>

            <p> ПОЛНОСТЬЮ РУЧНОЕ ИЗГОТОВЛЕНИЕ </p>

            <button class="buyHeader buttonToChange openModel" data-price="900000"> КУПИТЬ </button>

            <dialog class="dialogMain">

                <form class="orderForm">

                    <h2>Оформление заказа</h2>

                    <p><label for="name">Имя:</label>

                    <input type="text" id="name" name="name" placeholder="Имя" required></p>

                    <p><label for="secondName">Фамилия:</label>

                    <input type="text" id="secondName" name="secondName" placeholder="Фамилия" required></p>

                    <p><label for="thirdName">Отчество:</label>

                    <input type="text" id="thirdName" name="thirdName" placeholder="Отчество"></p>

                    <p><label for="address">Адрес доставки:</label>

                    <input type="text" id="address" name="address" placeholder="Ул. Пушкино, д. 2, кв. 1" required></p>

                    <p><label for="quantity">Количество:</label>

                    <input type="number" id="quantity" name="quantity" placeholder="1" min="1" required></p>

                    <p><label for="promo">Промокод:</label>

                    <input type="text" id="promo" placeholder="\*Промокод\*" name="promo"></p>

                    <p class="priceHeader"></p>

                    <input type="submit" value="Заказать!">

                </form>

                <button id="closeModel" class="closeModel">&#x2715 </button>

            </dialog>

        </div>

        <div class="rightText">

            <p> JACOB & CO <br>EPIC X TOURBILON BAGUETTE <br>900.000$</p>

            <img src="imgs/mainPhotoWatches.png">

        </div>

    </header>

    <main>

        <div class="textInMain">

            <p>ХИТЫ</p>

            <p>ПОСЛЕДНИЕ ПОСТУПЛЕНИЯ</p>

        </div>

        <div class="productsList">

            <div class="productUnit">

                <img src="imgs/appleulta.png">

                <p>Apple watch ulta 2</p>

                <p>700$</p>

                <button class="buttonUnit buttonToChange openModel" data-price="700"> Купить </button>

            </div>

            <div class="productUnit">

                <img src="imgs/rolexDayDate36.png">

                <p>Rolex Day-Date 36</p>

                <p>117.250$</p>

                <button class="buttonUnit buttonToChange openModel" data-price="117250"> Купить </button>

            </div>

            <div class="productUnit">

                <img src="imgs/KingSeiko.png">

                <p>King Seiko Automatic</p>

                <p>2.000$</p>

                <button class="buttonUnit buttonToChange openModel" data-price="2000"> Купить </button>

            </div>

            <div class="productUnit">

                <img src="imgs/PatekPhilippe.png">

                <p>Paket Philippe 5270/1R</p>

                <p>230.000$</p>

                <button class="buttonUnit buttonToChange openModel" data-price="230000"> Купить </button>

            </div>

            <div class="productUnit">

                <img src="imgs/rolexDateJust31.png">

                <p>Rolex DateJust 31</p>

                <p>51.900$</p>

                <button class="buttonUnit buttonToChange openModel" data-price="51900"> Купить </button>

            </div>

            <div class="productUnit">

                <img src="imgs/PatekPhilippeBlackWhite.png">

                <p>Patek Philippe 6300/400G</p>

                <p>9.000.000$</p>

                <button class="buttonUnit buttonToChange openModel" data-price="9000000"> Купить </button>

            </div>

            <div class="productUnit">

                <img src="imgs/appleseries10.png">

                <p>Apple watch series 10</p>

                <p>450$</p>

                <button class="buttonUnit buttonToChange openModel" data-price="450"> Купить </button>

            </div>

            <div class="productUnit">

                <img src="imgs/Jacob&Co.png">

                <p>Jacob&Co Palatial Classic</p>

                <p>16.500$</p>

                <button class="buttonUnit buttonToChange openModel" data-price="16500"> Купить </button>

            </div>

        </div>

    </main>

    <script src="script.js"></script>

</body>

</html>

Листинг style.css:

\* {

    margin: 0;

    padding: 0;

    overflow-x: hidden;

    font-family: "Overpass", serif;

    cursor: default;

}

button {

    cursor: pointer;

}

.changebackground {

    background-color: #f2f2f2 !important;

    color: black !important;

    box-shadow: 0px 5px 20px 8px rgba(34, 60, 80, 0.17) !important;

}

header {

    background-color: #f2f2f2;

    width: 100%;

    display: flex;

    height: 100vh;

}

header div {

    width: 50%;

    display: flex;

    flex-direction: column;

    align-items: center;

    justify-content: center;

    text-align: center;

}

.leftText > \* {

    margin: 0.75% 0;

}

.leftText h1 {

    font-size: 40px;

    font-weight: 700;

}

.leftText p {

    font-size: 20px;

}

.rightText img {

    width: 55%;

}

.rightText p {

    color: #c5c5c5;

    font-weight: 600;

}

.buyHeader {

    padding: 1.5% 4%;

    text-align: center;

    border: none;

    color: white;

    font-size: 20px;

    border-radius: 15px;

    background-color: black;

}

.dialogMain {

    margin: auto;

    top: 0;

    left: 0;

    right: 0;

    bottom: 0;

    width: 70%;

    height: 80%;

    position: absolute;

    border-radius: 20px;

    border: none;

    box-shadow: 0px 5px 20px 15px rgba(34, 60, 80, 0.17);

}

.changehtml {

    overflow: hidden;

}

.closeModel {

    position: absolute;

    right: 2%;

    top: 2%;

    border: none;

    background-color: white;

    font-size: 24px;

}

.orderForm {

    display: flex;

    flex-direction: column;

    gap: 17.5px;

    padding: 20px;

    width: 30%;

    margin: auto;

    font-family: "Overpass", sans-serif;

}

.orderForm h2 {

    text-align: center;

    margin: 5% 0;

}

.orderForm label {

    display: block;

    font-weight: bold;

}

.orderForm input {

    padding: 10px;

    border-radius: 8px;

    border: 1px solid #ccc;

    font-size: 16px;

}

.orderForm input[type="submit"] {

    background-color: black;

    color: white;

    border: none;

    cursor: pointer;

    font-weight: bold;

    margin-top: 10px;

    transition: 0.3s ease;

}

.orderForm input[type="submit"]:hover {

    background-color: #333;

}

input:not([type="submit"]) {

    width: 80%;

    cursor: text;

}

main {

    width: 100%;

    margin-top: 1.5%;

}

.textInMain {

    margin: 0 3%;

    font-weight: bold;

}

.textInMain p:first-of-type {

    font-size: 15px;

}

.textInMain p:last-of-type {

    font-size: 25px;

}

.productsList {

    margin: 0.5% 2%;

    display: grid;

    grid-template-columns: repeat(4, 1fr);

}

.productUnit {

    text-align: center;

    margin: 4% 3%;

    padding: 7% 0;

    box-shadow: 0px 5px 10px 2px rgba(34, 60, 80, 0.17);

    border-radius: 10px;

}

.productUnit \* {

    margin: 1% 0;

}

.productUnit img {

    width: 70%;

}

.buttonUnit {

    border: none;

    color: white;

    padding: 2% 3.5%;

    font-weight: 500;

    border-radius: 7px;

    background-color: black;

}

Листинг stypt.js:

// Смена цвета кнопки при наведении

const button = document.body.querySelectorAll(".buttonToChange");

for (let i = 0; i < button.length; i++) {

    button[i].addEventListener("mouseover", function () {

        button[i].classList.add("changebackground");

    });

    button[i].addEventListener("mouseout", function () {

        button[i].classList.remove("changebackground");

    });

}

// Модальное окно

const dialog = document.querySelector(".dialogMain");

const html = document.querySelector("html");

const openModels = document.querySelectorAll(".openModel");

const priceHeader = document.querySelector(".priceHeader");

let price;

for (let i = 0; i < openModels.length; i++) {

    openModels[i].addEventListener("click", function () {

        dialog.showModal();

        html.classList.add("changehtml");

        if (openModels[i].getAttribute("data-price") > 0) {

            price = parseInt(openModels[i].getAttribute("data-price"));

        }

        priceHeader.textContent = `Цена: ${price.toLocaleString()}$`;

    });

}

document.querySelector("#closeModel").onclick = function () {

    dialog.close();

    html.classList.remove("changehtml");

};

//Изменения цены

const formElement = document.querySelector(".orderForm");

const quantity = document.querySelector("#quantity");

const promo = document.querySelector("#promo");

const promoList = {

    "10procentSaleWOW": 0.9,

    "15yearShop": 0.85,

};

function updatePrice() {

    let quantityVal = 1;

    if (quantity.value > 1) {

        quantityVal = quantity.value;

    }

    let discount = promo.value in promoList ? promoList[promo.value] : 1;

    priceHeader.textContent = `Цена: ${createPrice(price, quantityVal, discount).toLocaleString()}$ (${quantityVal} шт, скидка ${Math.round((1 - discount) \* 100)}%)`;

}

function createPrice(priceFunc, quantityFunc, discountFunc) {

    let finalPrice = priceFunc \* quantityFunc \* discountFunc;

    return finalPrice;

}

quantity.addEventListener("input", updatePrice);

promo.addEventListener("input", updatePrice);

// Отправка формы

formElement.addEventListener("submit", () => {

    const formData = new FormData(formElement);

    const name = formData.get("name");

    const surname = formData.get("secondName");

    const thirdName = formData.get("thirdName");

    const address = formData.get("address");

    alert(`Спасибо, ${name} ${surname} ${thirdName}!\nВаш заказ на адрес - ${address} в количестве ${quantity.value}шт будет исполнен в ближайшее время!`);

});

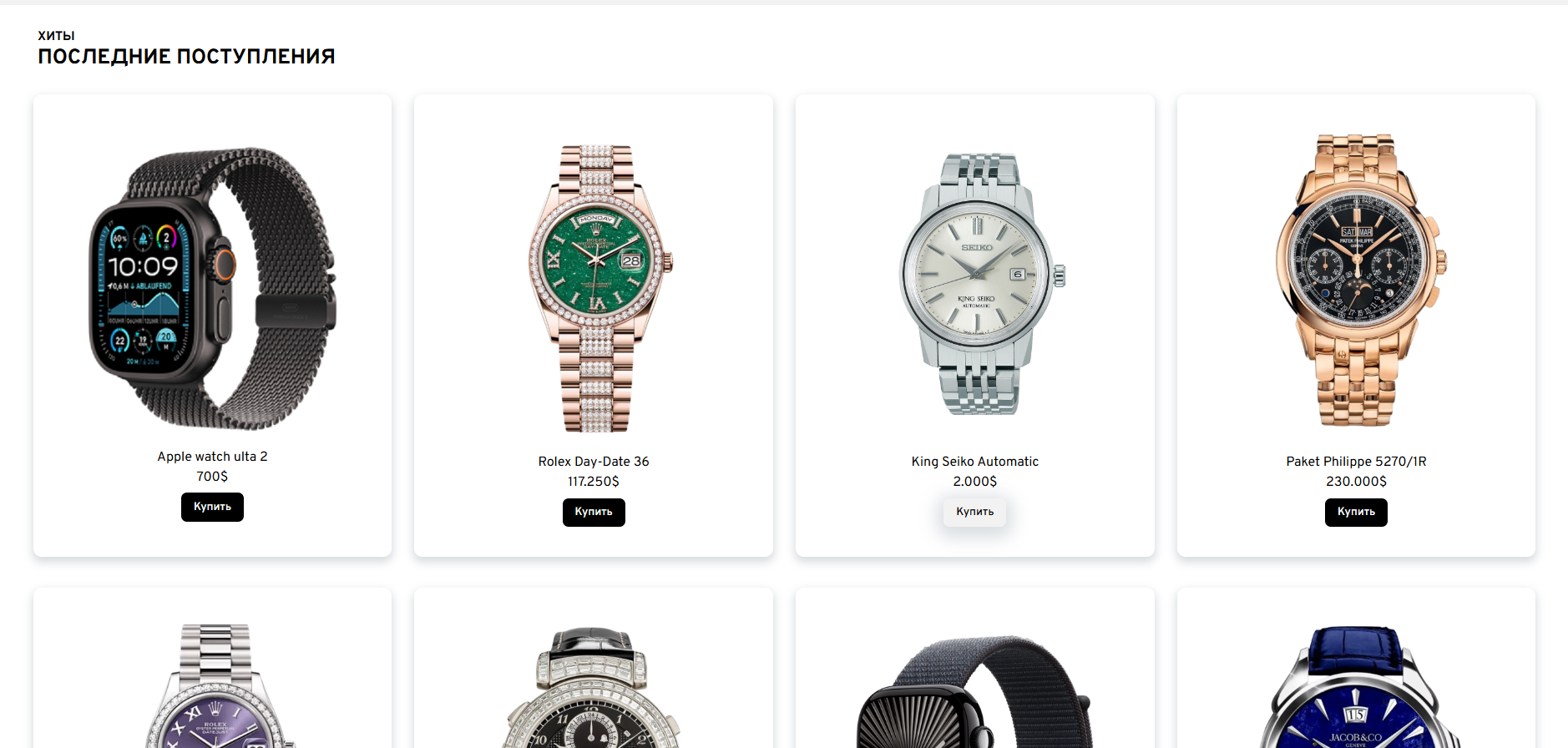


Рисунок 16. Изменение фона кнопки.

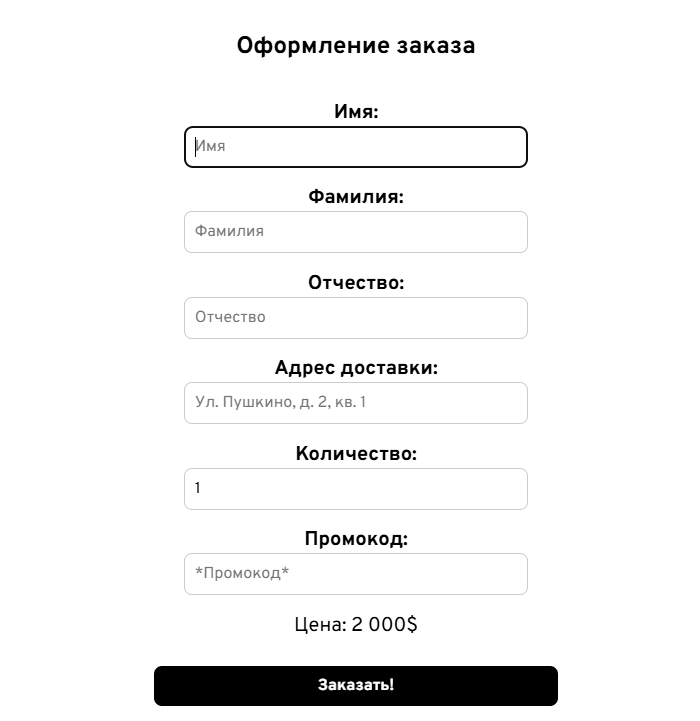


Рисунок 15. Форма оформление заказа.

ВЫВОД К РАБОТЕ:

Я установил необходимые инструменты, плагины и утилиты для работы с JavaScript. Создал динамическую HTML страницу с некоторым функционалом. Решил простейшие задачи на программирование (условия, циклы, функции).