



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

**О Т Ч Е Т**

**по лабораторной работе № 4**

Название: Основы JavaScript

Дисциплина: Языки интернет-программирования

Студент

ИУ6-31Б

(Группа)

\_\_\_\_\_  
(Подпись, дата)

Хан С.Т.

(И.О. Фамилия)

Преподаватель

\_\_\_\_\_  
(Подпись, дата)

В.Д.Шульман

(И.О. Фамилия)

Москва, 2024

**Цель работы:** изучение основных принципов программирования на языке Javascript, изучение способов обхода узлов модели документа DOM, изучение принципов динамического формирования кода страницы HTML, а также получение практических навыков отладки Javascript-приложений.

**Задание:**

1. Ознакомьтесь с разделом "6. JavaScript"
2. Сделайте форк данного репозитория в GitHub, склонируйте получившуюся копию локально, создайте от мастера ветку dev и переключитесь на неё
3. Выполните задания. Ссылки на задания содержатся в README-файлах в директории projects
4. Сделайте отчёт и поместите его в директорию docs
5. Зафиксируйте изменения, сделайте коммит и отправьте полученное состояние ветки dev в ваш удаленный репозиторий GitHub
6. Через интерфейс GitHub создайте Pull Request dev --> master
7. На защите лабораторной работы продемонстрируйте открытый Pull Request.

**Ход работы:**

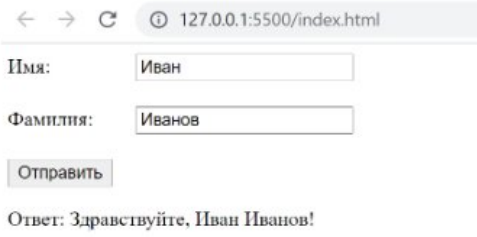
1. Решить задачи в директории projects.

**А) Задача «Form» (рис 1)**

**Задание**

Что должно получиться: пользователь открывает страницу, вводит имя и фамилию, нажимает "Отправить". Внизу появляется текст с приветствием. Затем можно изменить фамилию или имя, вновь нажать "Отправить" - приветствие обновится.

В данном случае не нужно использовать форму `<form>`.



Критерии оценивания:

- вы можете применить к странице любые стили или не применять их вовсе;
- не обязательно, чтобы элементы были расположены как в примере;
- **ОБЯЗАТЕЛЬНО:** в input вводятся имя и фамилия, после нажатия на кнопку Отправить появляется приветственное предложение в формате: Здравствуйте, Имя Фамилия!
- засчитываются варианты решения, если приветствие появляется при введении одного из полей или при отсутствии введенных значений. Здорово, если у вас получится сделать проверку на ввод значений, и только в этом случае показывать приветствие.

Рисунок 1

**Решение:**

**index.html**

```
<!DOCTYPE html>
<html lang="en">
<head>
  <meta charset="UTF-8">
  <meta http-equiv="X-UA-Compatible" content="IE=edge">
  <meta name="viewport" content="width=device-width, initial-scale=1.0">
  <title>Greeting</title>
  <link rel="stylesheet" href="style.css">
</head>
```

```

<body>
  <div class="container">
    <input type="text" id="firstName" placeholder="Имя">
    <input type="text" id="lastName" placeholder="Фамилия">
    <button onclick="showGreeting()">Отправить</button>
    <p id="greeting"></p>
  </div>
  <script src="script.js"></script>
</body>
</html>

```

```

style.css
.container {
  text-align: center;
  margin-top: 100px;
}
input {
  margin: 5px;
  padding: 5px;
}
button {
  padding: 5px 20px;
  background-color: #007bff;
  border: none;
  color: white;
  cursor: pointer;
}
button:hover {
  background-color: #0056b3;
}

```

```

script.js
function showGreeting() {
  const firstName = document.getElementById('firstName').value;
  const lastName = document.getElementById('lastName').value;
  if (firstName || lastName) {
    document.getElementById('greeting').innerText = `Здравствуйтесь, ${firstName} ${lastName}!`;
  } else {
    document.getElementById('greeting').innerText = 'Пожалуйста, введите ваше имя или фамилию.';
  }
}

```

**Тестирование (рис 2-5):**

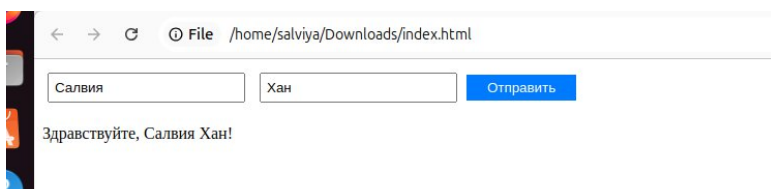


Рисунок 2 – первый пример

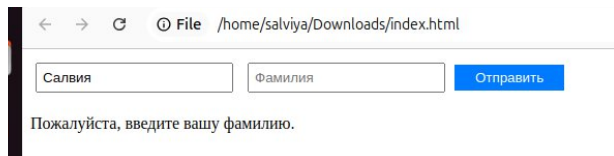


Рисунок 3 – второй пример



Рисунок 4 – третий пример

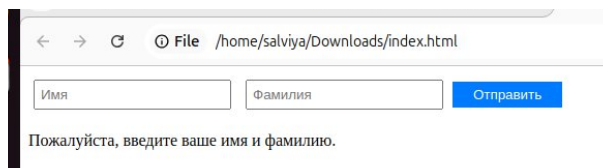
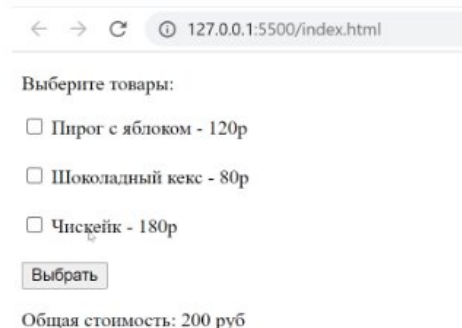


Рисунок 5 – четвертый пример

## В) Задача «Checkbox» (рис 6)

### Задание

**Что должно получиться:** пользователь открывает страницу и выбирает товары, которые хочет приобрести. После этого нажимает «Выбрать» и в строке «Общая стоимость» появляется стоимость покупки. Если пользователь меняет выбранные товары, после нажатия на кнопку «Выбрать» происходит перерасчет «Общей стоимости».



#### Критерии оценивания:

- вы можете применить к странице любые стили;
- не обязательно, чтобы элементы были расположены как в примере;
- **ОБЯЗАТЕЛЬНО:** наличие элементов checkbox, по нажатию на кнопку Выбрать рассчитывается и выводится Общая стоимость выбранных продуктов. Для основы можно взять код из предыдущего шага. У элементов **input** с типом **checkbox** в данном случае в **value** указана стоимость ( `<input type="checkbox" id="Choice3" name="food" value="180">` ).

Рисунок 6

### Решение:

#### index.html

```
<!DOCTYPE html>
<html lang="en">
<head>
  <meta charset="UTF-8">
  <meta http-equiv="X-UA-Compatible" content="IE=edge">
  <meta name="viewport" content="width=device-width, initial-scale=1.0">
  <title>Калькулятор стоимости товаров</title>
  <link rel="stylesheet" href="styles.css">
```

```

</head>
<body>
  <div class="container">
    <h1>Выберите товары</h1>
    <form id="productForm">
      <div class="product">
        <label>
          <input type="checkbox" value="120" class="product-checkbox"> Пирог с яблоком- 120 руб.
        </label>
      </div>
      <div class="product">
        <label>
          <input type="checkbox" value="80" class="product-checkbox"> Шоколадный кекс - 80 руб.
        </label>
      </div>
      <div class="product">
        <label>
          <input type="checkbox" value="180" class="product-checkbox"> Чизкейк - 180 руб.
        </label>
      </div>

      <button type="button" onclick="calculateTotal()">Выбрать</button>
    </form>
    <h2>Общая стоимость: <span id="totalCost">0</span> руб.</h2>
  </div>
  <script src="script1.js"></script>
</body>
</html>

```

### styles.css

```

body {
  font-family: Arial, sans-serif;
  background-color: #f0f0f0;
  text-align: center;
  padding: 20px;
}

.container {
  background-color: white;
  border-radius: 8px;
  padding: 20px;
  box-shadow: 0 0 10px rgba(0, 0, 0, 0.1);
  max-width: 400px;
  margin: 0 auto;
}

.product {
  margin: 10px 0;
}

button {
  padding: 10px 20px;
  background-color: #007bff;
  border: none;
  color: white;
  border-radius: 5px;
  cursor: pointer;
}

button:hover {
  background-color: #0056b3;
}

```

```
}
```

```
script.js
```

```
function calculateTotal() {  
  // Получаем все чекбоксы с классом 'product-checkbox'  
  const checkboxes = document.querySelectorAll('.product-checkbox');  
  let total = 0;  
  
  // Проходим по всем чекбоксам и суммируем значения от выбранных  
  checkboxes.forEach((checkbox) => {  
    if (checkbox.checked) {  
      total += parseInt(checkbox.value);  
    }  
  });  
  
  // Обновляем текст с общей стоимостью  
  document.getElementById('totalCost').innerText = total;  
}
```

Тестирование (7-9):

**Выберите товары**

- ☒ Пирог с яблоком- 120 руб.
- ☒ Шоколадный кекс - 80 руб.
- ☒ Чизкейк - 180 руб.

**Выбрать**

**Общая стоимость: 380 руб.**

Рисунок 7 – первый пример

Рисунок 8 – второй пример

Рисунок 9 – третий пример

### С) Задача «Store»

#### 3. Задача «Store»

##### 1) Условие:

Вам нужно с помощью JavaScript сделать страницу "живой".

Перед вами страница оформления заказа на сайте кофейни.

Пользователь отмечает товар, который хочет заказать и указывает кол-во. По умолчанию кол-во равно 0.

Внизу страницы есть строка Итого, которая меняет свое значение в процессе действий пользователя.

По нажатию на кнопку "Оформить заказ" появляется модальное окно, где указаны заказчик (Фамилия Имя) и Итого (общая сумма покупки).

**Что нужно реализовать:**

- Поля ввода для Фамилии и Имени: по кнопке "Оформить заказ" в модальном окне alert выводится строка Заказчик: Фамилия Имя (те, что были введены пользователем, если ничего не введено - пустая строка).
- Чекбоксы с наименованиями товаров. Пользователь нажимает на чекбокс и программа должна понимать, что данный товар выбран, а значит его нужно учитывать в общей сумме, либо наоборот - чекбокс снят, значит данный товар должен исчезнуть из суммы покупки.
- Рядом с каждым товаром находится поле ввода для указания количества товара. По умолчанию значение равно 0. В общей сумме покупки должно учитываться число, которое вводит пользователь в данное поле. Однако, если пользователь ввел число, а чекбокс не выбрал - данное кол-во не должно учитываться.
- Внизу над кнопкой "Оформить заказ" пользователю всегда пересчитывается итоговая сумма покупки исходя из действий пользователя.

### Модификации по усложнению (2 из 3):

- При выборе чекбокса - кол-во товара становится равно 1, при снятии чекбокса обнуляется.
- Проверка на ввод нежелательных чисел, например, запрет ввода отрицательных значений и значений вида 01, 02 и т.п. Например, при вводе нежелательного числа автоматически сменить его на 0, или выводить ошибку (однако неверно введенное число не должно тогда участвовать в вычислениях).
- Вы можете изменить наименования товаров, их цену, вы можете вывести по кнопке "Оформить заказ" более расширенную информацию, например: наименование товара - его кол-во - общую цену за данный товар и т.п. Вы можете добавить дополнительные блоки с товарами, например, размер стаканчика кофе - от его будет зависеть цена, и т.п. вещи.

### Решение:

**index2.html**

```

<!DOCTYPE html>
<html>
<head>
<meta charset="UTF-8">
<meta name="viewport" content="width=device-width, initial-scale=1.0">
<title>Document</title>
<link rel="stylesheet" href="style2.css">
</head>
<body>
<nav class="header">COFFEE HOUSE</nav>
<div class="main">
<div class="form">
<input type="text" id="last_name" placeholder="Введите Фамилию">
<input type="text" id="first_name" placeholder="Введите Имя">
</div>
<div class="market">
<div class="product">
<input type="checkbox" id="espresso" onclick="activateProduct(event)">
<label for="espresso">Эксперсco 80p.</label>

```



```

<input type="number" class="product_count" value="0"
oninput="updateProductCount(event)" value="0"
min="0" data-cost="80" data-in-cart="0">
</div>
<div class="product">
<input type="checkbox" id="latte" onclick="activateProduct(event)">
<label for="latte">Латте 110p.</label>
<input type="number" class="product_count" value="0"
oninput="updateProductCount(event)" value="0"
min="0" data-cost="110" data-in-cart="0">
</div>
<div class="product">
<input type="checkbox" id="kapuchino" onclick="activateProduct(event)">
<label for="kapuchino">Капучино 120p.</label>
<input type="number" class="product_count" value="0"
oninput="updateProductCount(event)" value="0"min="0" data-cost="120" data-in-cart="0">
</div>
<div class="product">
<input type="checkbox" id="americano" onclick="activateProduct(event)">
<label for="americano">Американо 90p.</label>
<input type="number" class="product_count" value="0"
oninput="updateProductCount(event)" value="0"
min="0" data-cost="90" data-in-cart="0">
</div>
<div class="product long">
<input type="checkbox" id="chocolate" onclick="activateProduct(event)">
<label for="chocolate">Шоколадный кекс 80p.</label>
<input type="number" class="product_count" value="0"
oninput="updateProductCount(event)" value="0"
min="0" data-cost="80" data-in-cart="0">
</div>
<div class="product long">
<input type="checkbox" id="chernika" onclick="activateProduct(event)">
<label for="chernika">Черничный кекс 90p.</label>
<input type="number" class="product_count" value="0"
oninput="updateProductCount(event)" value="0"
min="0" data-cost="90" data-in-cart="0">
</div>
<div class="product long">
<input type="checkbox" id="apple" onclick="activateProduct(event)">
<label for="apple">Яблочный тарт 100p.</label>
<input type="number" class="product_count" value="0"
oninput="updateProductCount(event)" value="0"
min="0" data-cost="100" data-in-cart="0">
</div>
</div>
<div class="result">
<div id="result_cost" class="result_cost">Итого: 0 p.</div>
<button class="send" onclick="sendInformation()">ОФОРМИТЬ
ЗАКАЗ</button>
</div></div>
<script src="script2.js"></script>

```

```
</body>
```

```
</html>
```

## style2.css

```
@import url('https://fonts.googleapis.com/css2?family=Oswald:wght@200..700&display=swap');
```

```
html,
```

```
body {
```

```
    margin: 0;
```

```
    font-family: 'Oswald', system-ui;
```

```
    display: flex;
```

```
    flex-direction: column;
```

```
    height: 100%;
```

```
}
```

```
.header {
```

```
    position: sticky;
```

```
    top: 0;
```

```
    left: 0;
```

```
    width: 100%;
```

```
    height: 110px;
```

```
    background: #ff69b4; /* Розовый цвет */
```

```
    display: flex;
```

```
    justify-content: center;
```

```
    align-items: center;
```

```
    font-size: 36px;
```

```
    color: white; /* Цвет текста в заголовке */
```

```
}
```

```
.main {
```

```
    padding-top: 55px;
```

```
    background: #ffc0cb; /* Светло-розовый фон */
```

```
    flex: 1 1 auto;
```

```
    display: flex;
```

```
    flex-direction: column;
```

```
    align-items: center;
```

```
}
```

```
.form {
```

```
    display: flex;
```

```
    column-gap: 64px;
```

```
}
```

```
.form input {
```

```
    background: transparent;
```

```
    border: 3px solid white;
```

```
    outline: none;
```

```
    font-size: 20px;
```

```
    font-weight: 300;
```

```
    text-align: center;
```

```
    width: 300px;
```

```
    height: 55px;
```

```
    color: white;
```

```
}
```

```
.form input::placeholder {
```

```
    color: white; /* Цвет плейсхолдера */
```

```
    font-family: "Oswald";
```

```

}
.market {
    max-height: 350px;
    display: flex;
    flex-flow: column wrap;
    column-gap: 132px;
    row-gap: 17px;
    margin-top: 84px;
}
.product {
    display: flex;
    align-items: center;
}
.product input[type="checkbox"] {
    appearance: none;
    width: 23px;
    height: 23px;
    background: #C4C4C4;
    border: 4px solid rgb(233, 225, 226);
}

.product input[type="checkbox"]:checked {
    appearance: none;
    width: 23px;
    height: 23px;
    background: rgb(255, 105, 180); /* Цвет фона при выборе */
}
.product label {
    width: 147px;
    font-family: inherit;
    font-weight: 300;
    font-size: 20px;
    padding-left: 16px;
    color: white; /* Цвет текста для товаров */
}
.long label {
    width: 203px;
}
.product_count {
    background: transparent;
    border: 3px solid white;
    outline: none;
    font-size: 20px;
    font-weight: 300;
    text-align: center;
    width: 130px;
    height: 55px;
    color: white; /* Цвет текста */
    font-family: inherit;
}
input::-webkit-outer-spin-button,
input::-webkit-inner-spin-button {

```

```

    -webkit-appearance: none;
    margin: 0;
}
.result {
    margin-top: 39px;
}
.result_cost {
    text-align: center;
    font-weight: 300;
    font-size: 20px;
    color: white; /* Цвет текста результата */
}
.send {
    margin-top: 26px;
    width: 190px;
    height: 55px;
    background-color: #ff69b4; /* Розовая кнопка */
    font-family: inherit;
    font-weight: 500;
    font-size: 20px;
    border: 3px solid transparent;
    color: white; /* Цвет текста кнопки */
}
.send:active {
    border: 3px solid white;
    background: transparent;
    color: white; /* Цвет текста при нажатии */
}

```

## script2.js

```

let finalCost = 0;
function activateProduct(ev) {
    let productElement =
        ev.target.parentElement.getElementsByClassName("product_count")[0];
    productElement.value = ev.target.checked ? 1 : 0;
    productElement.dataset["inCart"] = ev.target.checked ? "1" : "0";
    calcCost();
}
function updateProductCount(ev) {
    let value = ev.data ?? "";
    if (value == "-") {
        ev.target.value = "";
    } else if (!value.match(/(^$|^0$|^[1-9]\d*$)/)) {
        ev.target.value = "0";
    }
    calcCost();
}
function calcCost() {
    let products = document.querySelectorAll(`.product_count[data-in-cart="1"]`);
    let resultCost = document.getElementById("result_cost");
    let cost = 0;
    for (let product of products) {

```

```

    if (product.value != "") {
        cost += parseInt(product.dataset.cost) * parseInt(product.value);
    }
}
finalCost = cost;
resultCost.innerText = `Итого: ${cost} p.`;
}

function sendInformation() {
    let firstName = document.getElementById("first_name").value;
    let lastName = document.getElementById("last_name").value;
    let products = document.querySelectorAll('.product_count[data-in-cart="1"]');

    if (firstName != "" && lastName != "") {
        let orderDetails = `Заказчик: ${lastName} ${firstName}\n\n`;
        orderDetails += "Детали заказа:\n";
        for (let product of products) {
            if (product.value > 0) {
                let productName = product.getAttribute("data-product-name");
                let quantity = product.value;
                let price = parseInt(product.dataset.cost);
                let totalPrice = price * quantity;
                orderDetails += `${productName} - Количество: ${quantity} - Общая цена: ${totalPrice} p.\n`;
            }
        }
        orderDetails += `\nИтого: ${finalCost} p.`;
        alert(orderDetails);
    }
}

```

### Тестирование (рис 10-11):

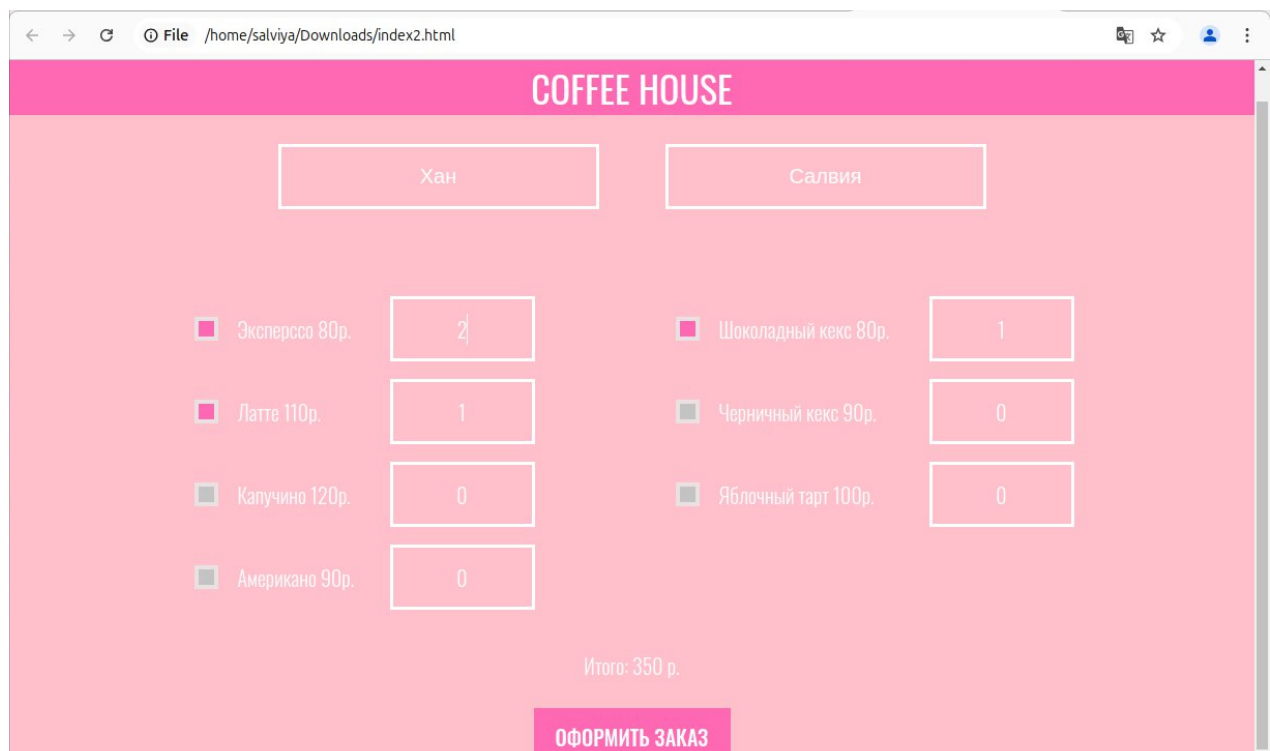


Рисунок 10 – пример окна для выбора

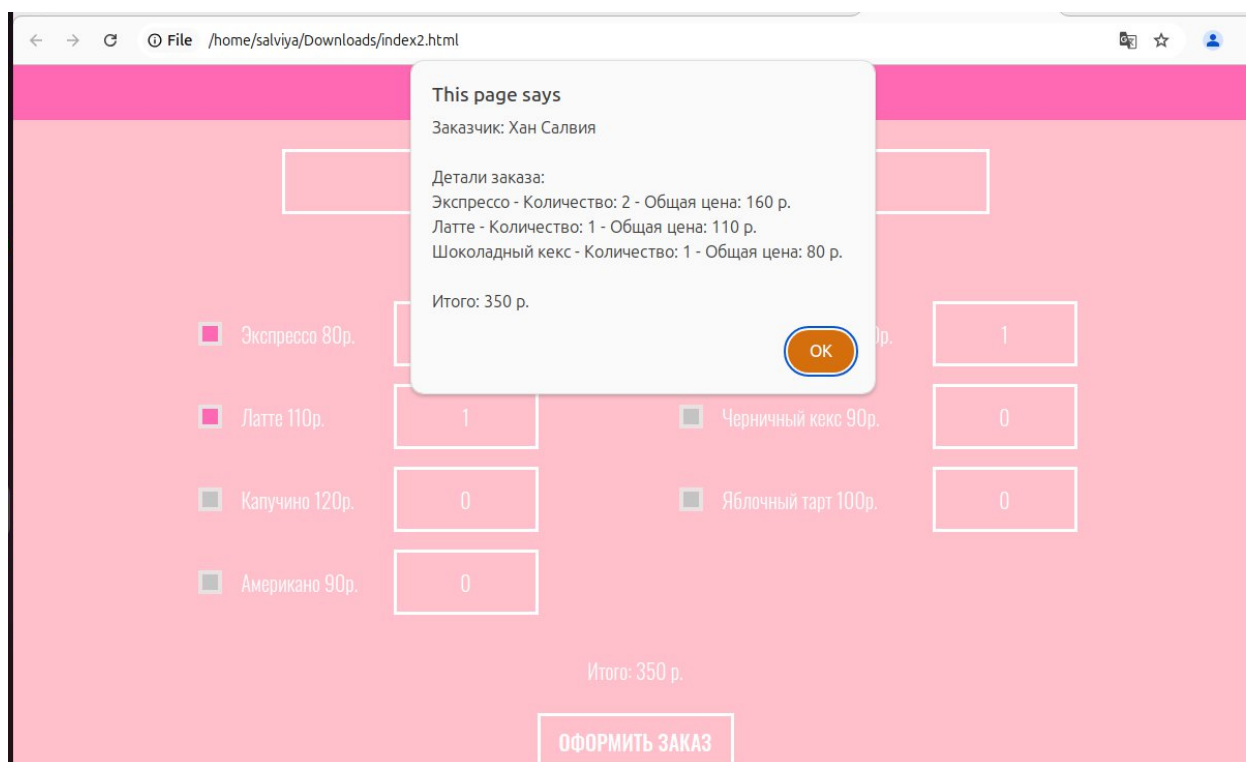


Рисунок 11 – пример оформления заказа

2. Форкнуть репозиторий называется lab-4.

**Заключение:** JS позволяет сделать сайт интерактивным и "живым".

**Список источников:**

1. <https://stepik.org/>