



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 09.04.01 - ИУ6-32Б

О Т Ч Е Т

по лабораторной работе № 1

Название: Основы Javascript

Дисциплина: Языки интернет-программирования

Студент

ИУ6-32Б

(Группа)

(Подпись, дата)

Кондратов С.Ю.

(И.О. Фамилия)

Преподаватель

(Подпись, дата)

Шульман В.Д.

(И.О. Фамилия)

Москва, 2024

Цель работы: научиться работать с Javascript

Задание 1

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».

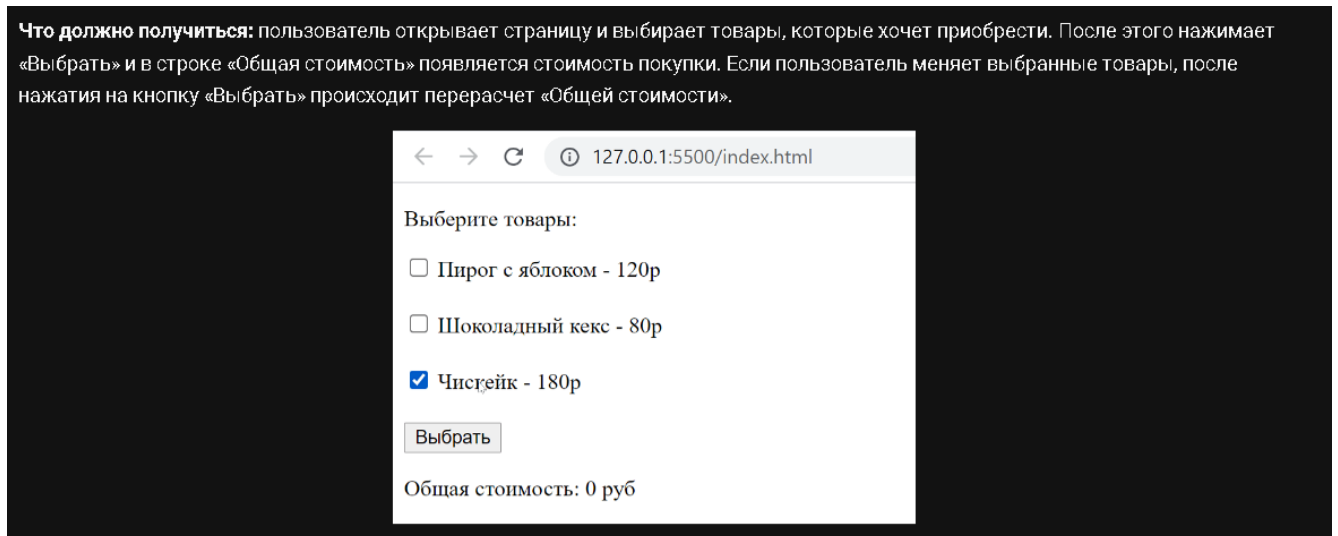


Рисунок 1

На рисунке 2,3,4 представлен мой результат и некоторые части кода.

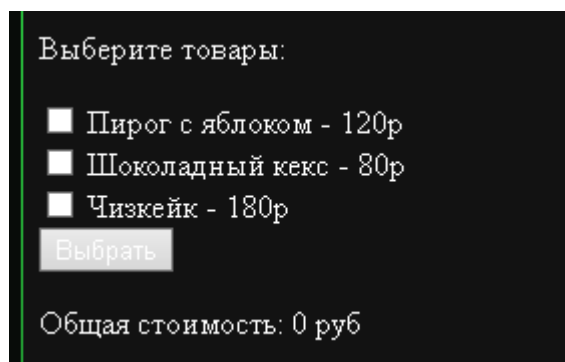


Рисунок 2

```

1  btnElement = document.querySelector(".btn");
2  resultElement = document.querySelector(".result");
3  checkboxButtons = document.querySelectorAll('[name="sweets"]');
4
5  btnElement.addEventListener("click", function(){
6      let Total = 0;
7      for (const checkboxButton of checkboxButtons){
8          if(checkboxButton.checked){
9              Total += parseInt(checkboxButton.value);
10         }
11     }
12     resultElement.textContent = `${Total} руб`;
13 });

```

Рисунок 3

```

<body>
  <p>Выберите товары:</p>
  <div>
    <input type="checkbox" id="sweetsChoice1" name="sweets" value="120">
    <label for="sweetsChoice1">Пирог с яблоком - 120р</label>
  </div>
  <div>
    <input type="checkbox" id="sweetsChoice2" name="sweets" value="80">
    <label for="sweetsChoice2">Шоколадный кекс - 80р</label>
  </div>
  <div>
    <input type="checkbox" id="sweetsChoice3" name="sweets" value="180">
    <label for="sweetsChoice3">Чизкейк - 180р</label>
  </div>
  <button class="btn">Выбрать</button>
  <p>Общая стоимость: <span class="result">0 руб</span></p>
  <script src="js.js"></script>
</body>

```

Рисунок 4

В задания мы должны были подключить Js файл, через который после получения определенных элементов страницы, провести над ними манипуляции.

Выберите товары:

- ☐ Пирог с яблоком - 120р
- ☒ Шоколадный кекс - 80р
- ☒ Чизкейк - 180р

Выбрать

Общая стоимость: 260 руб

Рисунок 5(пример вывода)

Задание 2

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>`.



← → ↻ 127.0.0.1:5500/index.html

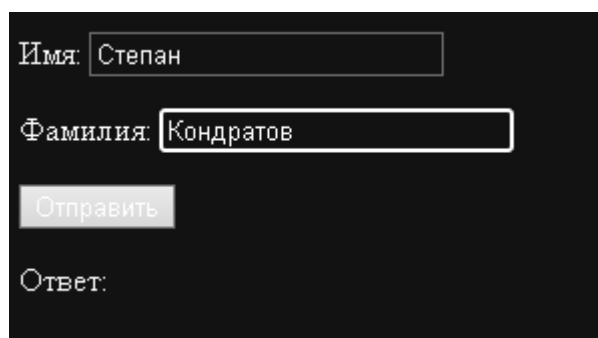
Имя:

Фамилия:

Отправить

Ответ:

Рисунок 6



Имя:

Фамилия:

Отправить

Ответ:

Рисунок 7 (мой результат)

Путем использования JS нам удастся доставать значения input с страницы и выводить этот текст.

Имя:

Фамилия:

Ответ: Здравствуйте, Степан Кондратов!

Рисунок 8 (пример вывода)

На рисунках 9,10 представлен части кода программы

```
1 <!DOCTYPE html>
2 <html lang="en">
3 <head>
4   <meta charset="UTF-8">
5   <meta http-equiv="X-UA-Compatible" content="IE=edge">
6   <meta name="viewport" content="width=device-width, initial-scale=1.0">
7   <title>Test task</title>
8 </head>
9 <body>
10   <p>Имя: <input class="input" type="text" name="myname"></p>
11   <p>Фамилия: <input class="input" type="text" name="mysename" ></p>
12   <p><button class="btn">Отправить</button></p>
13   <p class="result">Ответ:</p>
14   <script src="main.js"></script>
15 </body>
16 </html>
```

Рисунок 9

```
1 let btnElement = document.querySelector(".btn");
2 let myNameElement = document.querySelector('[name="myname"]');
3 let mySenameElement = document.querySelector('[name="mysename"]');
4 let resultElement = document.querySelector(".result");
5
6 btnElement.addEventListener("click", function() {
7   resultElement.textContent = `Ответ: Здравствуйте, ${myNameElement.value} ${mySenameElement
8   myNameElement.value = "";
9   mySenameElement.value = "";
10 });
```

Рисунок 10

Задание 3

COFFEE HOUSE

Иванов

Введите Имя

Мария

Mariia

css

u0912014_db_telegram_bot

u0912014_db_bot

js

☐ Эспresso 80р.

0

☐ Американо 110р.

0

☐ Латте 120р.

0

☐ Капучино 90р.

0

Итого: 0 р.

ОФОРМИТЬ ЗАКАЗ

Рисунок 11

Необходимо выбрать и реализовать 2 из 3 усложнений на выбор:

- 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.

После решения задания полученный код html+css+js необходимо перенести в данный репозиторий

Рисунок 12

COFFE HOUSE

Введите Фамилию

Введите Имя

Напитки

■

Эспрессо 80р.

0

■

Латте 110р.

0

■

Капучино 120р.

0

■

Американо 90р.

0

Десерты

■

Шоколадный кекс 80р.

0

■

Черничный кекс 90р.

0

■

Яблочный тарт 80р.

0

Итого:

ОФОРМИТЬ ЗАКАЗ

Рисунок 13(мой результат)

```

1 // Получение всех необходимых элементов
2 const surnameInput = document.querySelector('input[name="surname"]');
3 const nameInput = document.querySelector('input[name="name"]');
4
5 const checkboxes = document.querySelectorAll('.checkbox');
6 const quantityInputs = document.querySelectorAll('.count-choice');
7
8
9 const resultCount = document.querySelector('.result-count');
10 const orderButton = document.querySelector('.btn');
11
12
13 function updateTotal() {
14     let total = 0;
15
16     checkboxes.forEach((checkbox, index) => {
17         if (checkbox.checked) {
18             total += parseInt(checkbox.value) * parseInt(quantityInputs[index].value);
19         } else {
20             total = total;
21         }
22     });
23
24     resultCount.textContent = total + " руб.";
25 }
26
27
28 checkboxes.forEach((checkbox, index) => {
29     checkbox.addEventListener("change", updateTotal);
30     quantityInputs[index].addEventListener("change", updateTotal);
31 });
32
33 // Функция для обработки изменения значение в input
34 function handleQuantityChange(index) {
35     let quantity = quantityInputs[index].value;
36     console.log(quantity);
37
38     // Проверка на отрицательные числа и наличие ведущих нулей
39     if (!/^[1-9][0-9]*$/i.test(quantity)) {
40         quantityInputs[index].value = 0;
41         checkboxes[index].checked = false; // Сбрасываем чекбокс
42         quantity = 0; // Устанавливаем quantity в 0 для дальнейших расчетов
43     } else {
44         quantity = parseInt(quantity); // Обновляем quantity
45     }
46
47     quantityInputs[index].value = quantity; // Обновляем значение в поле
48     updateTotal(); // Обновляем итоги
49 }
50
51 // Обработка события выбора чекбокса
52 checkboxes.forEach((checkbox, index) => {
53     checkbox.addEventListener('change', () => {
54         if (checkbox.checked) {
55             quantityInputs[index].value = 1; // Устанавливаем количество на 1
56         } else {
57             quantityInputs[index].value = 0; // Обнуляем количество
58         }
59         updateTotal(); // Обновляем итог
60     });
61
62     // Изменение значения quantityInputs также обновляет итог
63     quantityInputs[index].addEventListener('input', () => handleQuantityChange(index));
64 });
65
66
67 orderButton.addEventListener("click", () => {
68     let surname = surnameInput.value.trim();
69     let name = nameInput.value.trim();
70     let price = resultCount.textContent;
71
72     alert(`Заказчик: ${surname} ${name}\nИтого: ${price}`);
73 });

```

Рисунок 14 (код JS)

Для реализации задачи усложнения я воспользовался методом `addEventListener` который позволяет реагировать на изменения `checkbox`-ов и основной кнопки приложения.

URL: `pylabs_git/web-4/projects/store/index.html`

This page says
Заказчик: Кондратов Степан
Итого: 1840 руб.

OK

Кондратов Степан

Напитки

<input checked="" type="checkbox"/> Эспрессо 80р.	1
<input checked="" type="checkbox"/> Латте 110р.	1
<input checked="" type="checkbox"/> Капучино 120р.	10
<input type="checkbox"/> Американо 90р.	0

Десерты

<input type="checkbox"/> Шоколадный кекс 80р.	0
<input checked="" type="checkbox"/> Черничный кекс 90р.	5
<input type="checkbox"/> Яблочный тарт 80р.	0

Итого: 1840 руб.

ОФОРМИТЬ ЗАКАЗ

Рисунок 15 (пример вывода программы)

Заключение

Я научился работать с JavaScript. Как доставать конкретные элементы страницы, добавлять слушатели событий и изменять и считывать значения элементов.

Контрольные вопросы

1. В чём разница между `let` и `var`? -

var объявляет переменную в глобальной области видимости или в области видимости функции, в которой она объявлена. Это означает, что переменная будет доступна во всей функции или даже за пределами функции, если она объявлена вне функции.

let, с другой стороны, объявляет переменную в блоковой области видимости, то есть она доступна только внутри блока, в котором она объявлена. Это означает, что переменная не будет доступна вне блока, даже если блок находится внутри функции.

2. Какие существуют способы подключения js-кода к html-документу?

- В теге `<script>` можно сразу написать целый код на JS либо через параметр `src` передать путь к файлу

3. В чём отличие операторов `===` и `==`?

Оператор `==` выполняет строгий сравнения двух значений. Он сравнивает значения без преобразования типов. Если значения имеют разные типы, то JavaScript пытается преобразовать их в общий тип перед сравнением. Если преобразование невозможно, то значения считаются неравными.

Оператор `===` выполняет строгий сравнения двух значений, при этом также проверяя типы значений. Если значения имеют разные типы, то они считаются неравными. Если значения имеют одинаковый тип, то JavaScript сравнивает их без преобразования типов.

4. Как можно отладить js-код с помощью браузера?

Через инструменты разработчика.

Перейдите на вкладку "Консоль": На этой вкладке можно просматривать ошибки и предупреждения, которые возникают при выполнении JavaScript-кода.

Используйте точки останова: Чтобы остановить выполнение кода в определенной точке, можно использовать точки останова. Для этого нужно нажать на номер строки в коде, где вы хотите остановить выполнение.

Используйте команды отладки: В инструментах разработчика можно использовать команды отладки, такие как `debugger`, `console.log()`, `console.error()` и другие, чтобы отслеживать выполнение кода и выявлять ошибки.

5. Для чего нужна директива `use strict`?

Когда вы включаете строгий режим, JavaScript начинает более строго следить за синтаксисом и семантикой кода. Это означает, что некоторые конструкции, которые в обычном режиме могут быть допущены, в строгом режиме будут вызывать ошибки.

Некоторые примеры того, что меняется при включении строгого режима:

Нельзя использовать не объявленные переменные.

Нельзя удалять переменные и функции с помощью оператора `delete`.

Нельзя использовать ключевые слова как имена переменных.

Нельзя использовать функцию `with`.

Нельзя использовать функцию eval для создания переменных в текущем контексте.