

# 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 № \_3\_

**Название:** <u>Javascript. Добавление динамического поведения HTML.</u> Средства отладки

Дисциплина: Языки Интернет-программирования

 
 Студент
 ИУ6-33Б (Группа)
 Дел. 16.09.2022 (Подпись, дата)
 А.И. Меслауров (И.О. Фамилия)

 Преподаватель
 Э. Р. Маняшев (Подпись, дата)
 (И.О. Фамилия)
 **Цель работы:** Целью работы является 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.

#### Часть 1.

**Пункт 1.** Сформировать страницу с произвольным кодом разметки, но обеспечить уровень вложенности внутри элемента <br/>
<br/>body> не менее 3.

Код страницы (файл index.html):

```
<!DOCTYPE html>
<html lang="en">
  <meta charset="UTF-8">
  <meta name="viewport" content="width=device-width, initial-scale=1.0">
  <style>
     table {
        border-collapse: collapse;
     td {
        padding: 3px;
        border: 1px solid grey;
  </style>
  <script src="script.js"></script>
  <title>Document</title>
</head>
<body>
  <thead>
        caption1
           caption2
        </thead>
     item1
           item2
        item1
           item2
        item1
           item2
```

```
<button id="start">Обход</button>
<div id="result">Результат обхода:</div>
</body>
</html>
```

Программа обхода (файл script.js):

```
window.onload = function() {
   var text = "";
   var c = 0;
   function countElements(node) {
      var elementsNum = 0;
      if (node.nodeType == 1) {
         elementsNum++;
         console.log(node.nodeName+', increment the counter +1');
         for (j = 0; j < c; j++) {
            a = a + '&nbsp'.repeat(4);
         text = text + a + node.nodeName + "<br>";
      } else {
         console.log(node.nodeName+', not an element')
      var children = node.childNodes;
      for (var i=0; i < children.length; i++) {</pre>
         c = c + 1;
         elementsNum += countElements(children[i]);
         c = c - 1;
      return elementsNum;
   function resultInDiv() {
      result = document.getElementById('result');
      var allElementsNum = countElements(document);
      result.innerHTML = "Результат обхода:<br/>- text + 'Количество подсчитанных
элементов: ' + allElementsNum + '.';
   var btn = document.getElementById('start');
   btn.addEventListener('mousedown', resultInDiv);
```

Пример обхода дерева элементов страницы (см. рис. 1):

#### Elements Console Network » Sources caption1 caption2 Default levels ▼ No Issues item1 item2 script.js:8 item1 item2 #document, not an element html, not an element script.js:8 item1 item2 HTML, increment the counter +1 script.js:6 Обход HEAD, increment the counter +1script.js:6 Количество подсчитанных элементов: 25. #text, not an element script.js:8 META, increment the counter +1script.js:6 #text, not an element script.js:8 META, increment the counter +1 script.js:6 #text, not an element script.js:8 STYLE, increment the counter +1 script.js:6 #text, not an element script.js:8 #text, not an element script.js:8 SCRIPT, increment the counter +1script.js:6 #text, not an element script.js:8 TITLE, increment the counter +1script.js:6 #text, not an element script.js:8 #text, not an element script.js:8 #text, not an element script.js:8 BODY, increment the counter +1 script.js:6 #text, not an element script.js:8 TABLE, increment the counter +1 script.js:6 #text, not an element script.js:8 THEAD, increment the counter +1 script.js:6 #text, not an element script.js:8

Рис. 1 – пример обхода дерева элементов страницы.

TR, increment the counter +1

TH, increment the counter +1

TH, increment the counter +1

Highlights from the Chrome 105 update

#text, not an element

#text, not an element

#text, not an element

Console What's New X

script.js:6

script.js:6

script.js:8

script.js:8

script.js:6

script.js:8

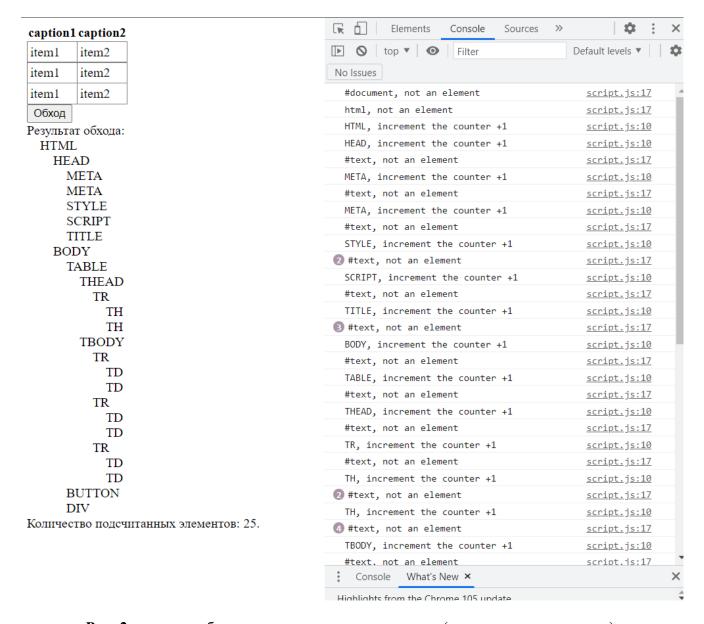


Рис. 2 – пример обхода дерева элементов страницы ( с выводом на страницу).

**Пункт 2.** Добавить внутри элемента <br/> <br/>body> секцию <div>, предназначенную для вывода результата обхода дерева элементов страницы.

Добавлено в коде выше.

**Пункт 3.** Выбрать способ активации рекурсивной программы обхода дерева элементов, реализовать и подключить эту программу.

Выполнено в коде выше.

**Пункт 4.** При проходе по узлам разметки обеспечить отладочный вывод в консоль. Привести в отчете содержимое консоли.

#### Содержимое консоли:

```
script.js:8 #document, not an element
script.js:8 html, not an element
```

```
script.js:6 HTML, increment the counter +1
script.js:6 HEAD, increment the counter +1
script.js:8 #text, not an element
script.js:6 META, increment the counter +1
script.js:8 #text, not an element
script.js:6 META, increment the counter +1
script.js:8 #text, not an element
script.js:6 STYLE, increment the counter +1
script.js:8 #text, not an element
script.js:8 #text, not an element
script.js:6 SCRIPT, increment the counter +1
script.js:8 #text, not an element
script.js:6 TITLE, increment the counter +1
script.js:8 #text, not an element
script.js:8 #text, not an element
script.js:8 #text, not an element
script.js:6 BODY, increment the counter +1
script.js:8 #text, not an element
script.js:6 TABLE, increment the counter +1
script.js:8 #text, not an element
script.js:6 THEAD, increment the counter +1
script.js:8 #text, not an element
script.js:6 TR, increment the counter +1
script.js:6 TH, increment the counter +1
script.js:8 #text, not an element
script.js:8 #text, not an element
script.js:6 TH, increment the counter +1
script.js:8 #text, not an element
script.js:8 #text, not an element
script.js:8 #text, not an element
script.js:6 TBODY, increment the counter +1
script.js:8 #text, not an element
script.js:6 TR, increment the counter +1
script.js:8 #text, not an element
script.js:6 TD, increment the counter +1
script.js:8 #text, not an element
script.js:8 #text, not an element
script.js:6 TD, increment the counter +1
script.js:8 #text, not an element
script.js:8 #text, not an element
script.js:8 #text, not an element
script.js:6 TR, increment the counter +1
script.js:8 #text, not an element
script.js:6 TD, increment the counter +1
script.js:8 #text, not an element
script.js:8 #text, not an element
script.js:6 TD, increment the counter +1
script.js:8 #text, not an element
script.js:8 #text, not an element
script.js:8 #text, not an element
script.js:6 TR, increment the counter +1
script.js:8 #text, not an element
script.js:6 TD, increment the counter +1
script.js:8 #text, not an element
script.js:8 #text, not an element
```

```
script.js:6 TD, increment the counter +1
script.js:8 #text, not an element
script.js:6 BUTTON, increment the counter +1
script.js:8 #text, not an element
script.js:8 #text, not an element
script.js:6 DIV, increment the counter +1
script.js:8 #text, not an element
script.js:8 #text, not an element
script.js:8 #text, not an element
```

Пункт 5. Реализовать вывод на странице.

Реализовано в коде выше.