



**МИНОБРНАУКИ РОССИИ**  
**федеральное государственное бюджетное образовательное учреждение**  
**высшего образования**  
**«Московский государственный технологический университет**  
**«СТАНКИН»**  
**(ФГБОУ ВО «МГТУ «СТАНКИН»)**

---

Институт автоматизации и робототехники  
Кафедра компьютерных систем управления

Образовательная программа 15.03.04  
«Автоматизация технологических процессов и производств»

Дисциплина «Основы системного программного  
обеспечения»

**Отчет**  
**по лабораторной работе №3**  
**«Введение в HTML, CSS и JavaScript»**

Выполнил:

студент

гр.АДБ-21-0

\_\_\_\_\_

(дата)

\_\_\_\_\_

(подпись)

Маслова Е.П.

Принял:

к.т.н., доцент

\_\_\_\_\_

(дата)

\_\_\_\_\_

(подпись)

Ковалев И.А.

**Москва 2024**

**Цель работы:** Научиться создавать простые страницы сайтов используя Язык HTML JavaScript и CSS. Разобраться как происходит процесс разработки простых диалогов.

## Теоретические сведения

### Язык HTML

Язык гипертекстовой разметки (HyperText Markup Language — HTML, рис.1), основной строительный блок веб-страниц, используется для создания и визуального представления веб-страниц. Он определяет содержание страницы, но не её функциональность.



Рис. 1. – HyperText Markup Language.

HTML добавляет разметку в обычный текст. Гипертекст содержит ссылки, которыми веб-страницы связываются друг с другом, делая Всемирную паутину тем, чем она является сегодня. HTML поддерживает как изображения, так и другой медиаконтент. С помощью HTML каждый может создать статический, а также динамический сайт. HTML является языком, описывающим структуру и семантику содержимого веб-документа. Контент веб-страницы размечен с помощью тегов, представляющих HTML-элементы. Примерами таких элементов являются `<img>`, `<title>`, `<p>`, `<div>`, `<picture>` и так далее. Эти элементы формируют строительные блоки для любого веб-сайта.

Большая часть элементов HTML представляет собой элементы, сформированные тегами открытия и тегами закрытия ("`<p>text here</p>`") этих элементов, как и в других языках форматирования. Не каждый элемент должен сопровождаться тегом закрытия, например – есть теги переноса строки `<br>` и тому подобные.

### Объектная Модель Документа (DOM)

Объектная Модель Документа (DOM, рис.2) – это программный интерфейс (API) для HTML и XML документов. DOM предоставляет структурированное представление документа и определяет то, как эта структура может быть доступна из программ, которые могут изменять содержимое, стиль и структуру документа. Представление DOM состоит из структурированной группы узлов и объектов, которые имеют свойства и методы. По

существо DOM соединяет веб-страницу с языками описания сценариев либо языками программирования.

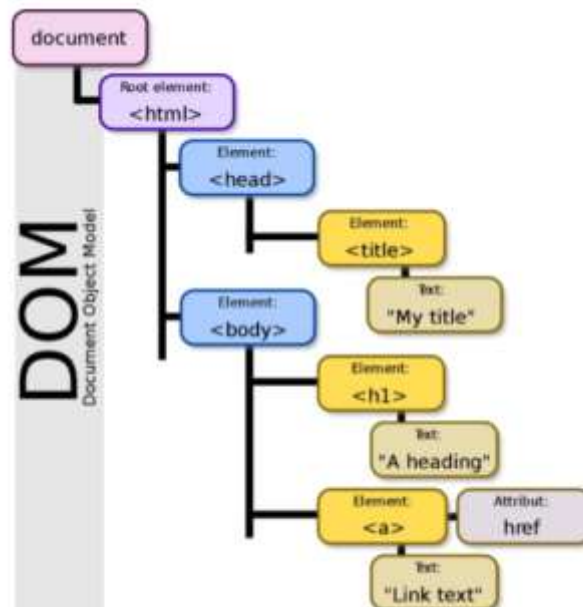


Рис. 2. Модель DOM.

## JavaScript

JavaScript изначально создавался для того, чтобы сделать web-странички «живыми». Программы на этом языке называются скриптами. В браузере они подключаются напрямую к HTML и, как только загружается страничка – тут же выполняются.

Программы на JavaScript – обычный текст. Они не требуют какой-то специальной подготовки. В этом плане JavaScript сильно отличается от другого языка, который называется Java.

Для выполнения программ, не важно на каком языке, существуют два способа:

«компиляция» и «интерпретация»:

- Компиляция – это когда исходный код программы, при помощи специального инструмента, другой программы, которая называется «компилятор», преобразуется в другой язык, как правило – в машинный код. Этот машинный код затем распространяется и запускается. При этом исходный код программы остаётся у разработчика.
- Интерпретация – это когда исходный код программы получает другой инструмент, который называют «интерпретатор», и выполняет его «как есть». При этом

распространяется именно сам исходный код (скрипт). Этот подход применяется в браузерах для JavaScript.

Современные интерпретаторы перед выполнением преобразуют JavaScript в машинный код или близко к нему, оптимизируют, а уже затем выполняют. И даже во время выполнения стараются оптимизировать. Поэтому JavaScript работает очень быстро. Во все основные браузеры встроен интерпретатор JavaScript, именно поэтому они могут выполнять скрипты на странице. Но, разумеется, JavaScript можно использовать не только в браузере. Это полноценный язык, программы на котором можно запускать и на сервере, и даже в стиральной машинке, если в ней установлен соответствующий интерпретатор. Современный JavaScript – это «безопасный» язык программирования общего назначения. Он не предоставляет низкоуровневых средств работы с памятью, процессором, так как изначально был ориентирован на браузеры, в которых это не требуется. Что же касается остальных возможностей – они зависят от окружения, в котором запущен JavaScript. В браузере JavaScript умеет делать всё, что относится к манипуляции со страницей, взаимодействию с посетителем и, в какой-то мере, с сервером:

- Создавать новые HTML-теги, удалять существующие, менять стили элементов, прятать, показывать элементы и т.п.
- Реагировать на действия посетителя, обрабатывать клики мыши, перемещения курсора, нажатия на клавиатуру и т.п.
- Посылать запросы на сервер и загружать данные без перезагрузки страницы (эта технология называется "AJAX").
- Получать и устанавливать cookie, запрашивать данные, выводить сообщения и др.

JavaScript – быстрый и мощный язык, но браузер накладывает на его исполнение некоторые ограничения. Это сделано для безопасности пользователей, чтобы злоумышленник не мог с помощью JavaScript получить личные данные или как-то навредить компьютеру пользователя. Этих ограничений нет там, где JavaScript используется вне браузера, например на сервере. Кроме того, современные браузеры предоставляют свои механизмы по установке плагинов и расширений, которые обладают расширенными возможностями, но требуют специальных действий по установке от пользователя.

Большинство возможностей JavaScript в браузере ограничено текущим окном и страницей. Список ограничений JavaScript представлен ниже:

- JavaScript не может читать/записывать произвольные файлы на жесткий диск, копировать их или вызывать программы. Он не имеет прямого доступа к операционной системе (рис.3).
- Современные браузеры могут работать с файлами, но эта возможность ограничена специально выделенной директорией – «песочницей». Возможности по доступу к устройствам также прорабатываются в современных стандартах и частично доступны в некоторых браузерах.
- JavaScript, работающий в одной вкладке, не может общаться с другими вкладками и окнами, за исключением случая, когда он сам открыл это окно или несколько вкладок из одного источника (одинаковый домен, порт, протокол).
- Есть способы это обойти, и они раскрыты в учебнике, но они требуют специального кода на оба документа, которые находятся в разных вкладках или окнах. Без него, из соображений безопасности, залезть из одной вкладки в другую при помощи JavaScript нельзя.
- Из JavaScript можно легко посылать запросы на сервер, с которого пришла страница. Запрос на другой домен тоже возможен, но менее удобен, т. к. и здесь есть ограничения безопасности.

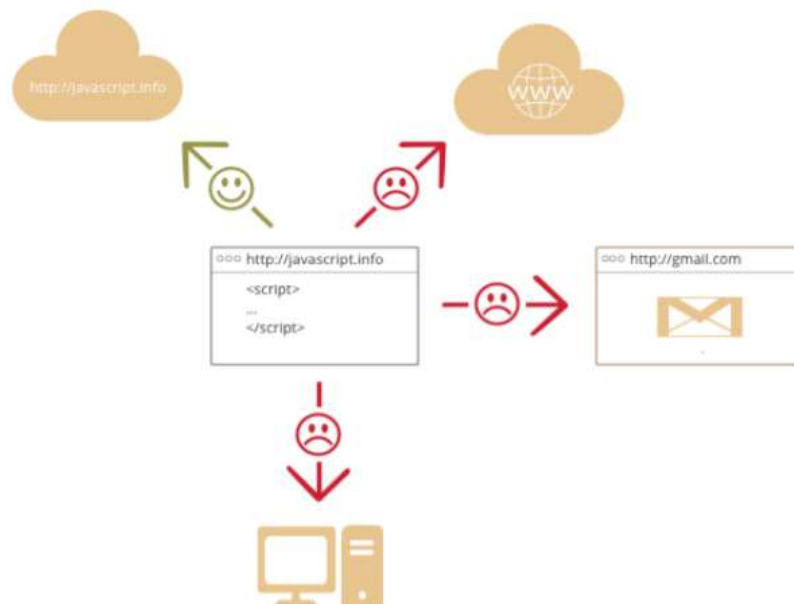


Рис. 3. Ограничения JavaScript.

## CSS

CSS (Каскадные Таблицы Стилей, рис.4) позволяют создавать стилизованные и отформатированные веб-страницы, состоящие из слоев, блоков и так далее. Рассмотрим, что такое CSS и как браузеры превращают HTML в (DOM), как CSS применяется к различным частям DOM, некоторые базовые примеры синтаксиса и подключение CSS к нашей веб-странице.



Рис. 4. – CSS.

Документ обычно является текстовым файлом, который структурирован при помощи языка разметки. HTML это самый распространенный язык разметки, но вы можете столкнуться и с другими (SVG или XML).

Представление документа пользователю значит его конвертацию в понятную для пользователя форму. Браузеры, такие как Firefox, Chrome или Internet Explorer, предназначены для представления документов визуально, например, на экране компьютера, проектор или принтер.

Рассмотрим влияние CSS на HTML. Браузер применяет CSS правила к документу чтобы описать, как он будет отображаться. CSS-правила формируются из:

- Набора свойств, которые имеют значения, устанавливающие, как будет отображаться содержимое (HTML). Например, можно сделать, чтобы ширина элемента равнялась 50% ширины родительского элемента и его фон был красным.
- Селектор, который выбирает (англ. selects) элемент/элементы, к нужно применить измененные значения. Например, можно применить это CSS-правило ко всем параграфам в HTML-документе.

Набор правил CSS, содержащихся в таблице стилей (stylesheet), определяет, как должна выглядеть веб-страница.

## Практическая часть

### Задание №1

Реализуйте одну из следующих функций:

Функция - Нахождение максимума двух чисел.  $\max(a,b)$

```
<!DOCTYPE html>
<html>
<head>
  <title><i>Какое из чисел максимально?</i></title>
</head>
<body>
  <h5>Найдем максимум двух чисел</h5>
  <input type="number" id="n1">
  <b>
    <label1 for = "n1">-первое число</label1>
  </b>
  <br>
  <input type="number" id="n2">
  <b>
    <label for="n2">-второе число</label>
  </b>
  <br>
  <br>
  <h1>
    <button onclick="findMax()">Вычислить</button>
    <br>
    <p id="otvet"></p>
  </h1>

  <script>
    function findMax(){
      var n1 = document.getElementById('n1').value;
      var n2 = document.getElementById('n2').value;

      var maxN = Math.max(n1, n2);

      document.getElementById('otvet').textContent = "Максимальное число: " + maxN;
    }
  </script>
</body>
<body>
  <h2><button onclick="window.location.href = 'index.html';">Назад</button></h2>
</body>
</html>
```

*Какое из чисел максимально?*

Найдем максимум двух чисел

<input type="text" value="4"/>	-первое число
<input type="text" value="97"/>	-второе число

Вычислить

**Максимальное число: 97**

Назад

## Задание №2

Объявите две переменные: admin и name.

Запишите в name строку свое имя.

Скопируйте значение из name в admin.

Выведите admin.

```
<!DOCTYPE html>
<html>
<head>
  <h2>Взаимодействие с пользователем</h2>
</head>
<body>
  <script>
    if (confirm('Доброго времени суток, у этой страницы есть к вам вопрос, вы готовы ответить?')) {
      var name = prompt('Как вас зовут?', 'Введите сюда ваше имя');
      alert("Ваше имя - " + name);
    }
    alert("Спасибо, что уделите время");

    var admin, name;

    //Записываем в name строку с именем
    name = "Елизавета";

    //Копируем значение из name в admin
    admin = name;

    //Выводим admin
    alert("Значение admin: " + admin);
    document.getElementById('res').textContent = "admin: " + admin;

  </script>
</body>
</html>
<h2><button onclick="window.location.href = 'index.html';">Назад</button></h2>
```



**Подтвердите действие**  
Доброго времени суток, у этой страницы есть к вам вопрос, вы готовы ответить?

**Подтвердите действие**  
Как вас зовут?

**Подтвердите действие**  
Ваше имя - Елизавета

**Подтвердите действие**  
Спасибо, что уделите время

**Подтвердите действие**  
Значение admin: Елизавета

### Задание №3

Напишите скрипт, который будет спрашивать ваш возраст и после ввода возраста уточнять его.

```

<!DOCTYPE html>
<html>
<head>
  <h1>Проверка возраста</h1>
</head>
<body>
  <script>
    var year;
    while (true){
      year = prompt('Сколько вам лет?', '');

      if (year != null)
      {
        if (confirm('Вам точно ' + year + ' лет?'))
        {
          break;
        }
      }
      else
      {
        alert('Пожалуйста, введите ваш возраст.');
      }
    }
    alert("Спасибо, что уделите время");
  </script>
</body>
<body>
  <h2><button onclick="window.location.href = 'index.html';">Назад</button></h2>
</body>
</html>

```

Подтвердите действие

Сколько вам лет?

Подтвердите действие

Вам точно 20 лет?

Подтвердите действие

Спасибо, что уделите время

#### Задание №4

Загрузите код страницы из репозитория на GitHub, расположенного по следующему адресу – <https://github.com/Vchekryzhov/labs> .

Выполните любые 5 пунктов:

1. Добавить к анкете ещё два пункта: один - с чекбоксом, второй - с текстом.
2. Сделать проверку данных, при вводе, если пользователь вводит неверные данные, то повторять ввод до тех пор, пока не введет правильно. Имя – только текст, возраст – неотрицательные целые числа, пол – М или Ж.
3. Скрывать элемент «меня возьмут» пока пользователь не заполнит анкету.
4. Запретить изменение всех `checkbox` после того, как пользователь нажал кнопку «меня возьмут».
5. Скрыть элемент «заполнить анкету» после заполнения анкеты.
6. Изменить функцию `check()` так, что бы она показывала положительный результат, если у пользователя есть навыки: знание математики и знание программирования.
7. Изменить функцию `check()` так, что бы она показывала положительный результат, если у пользователя есть высшее образование и ещё два любых навыка.
8. Реализовать возможность редактирования имени, пола и возраста.

```

<!DOCTYPE html>
<html>
<head>
<!--Import Google Icon Font-->
<link href="https://fonts.googleapis.com/icon?family=Material+Icons" rel="stylesheet">
<!--Import materialise.css-->
<link type="text/css" rel="stylesheet" href="css/materialise.min.css" media="screen,projection"/>
<link type="text/css" rel="stylesheet" href="css/main.css" media="screen,projection"/>
<meta charset="utf-8">
<!--Let browser know website is optimized for mobile-->
<meta name="viewport" content="width=device-width, initial-scale=1.0"/>
</head>

<body>
<!--Import jQuery before materialise.js-->
<script type="text/javascript" src="https://code.jquery.com/jquery-3.2.1.min.js"></script>
<script type="text/javascript" src="js/materialise.min.js"></script>

<div class="row">
<div class="col s2">
<div class="row">
<form class="col s12">
<div class="row">
<div class="input-field col s12">
<input id="name" type="text" class="validate">
<label for="name">Имя</label>
</div>
</div>
<div class="row">
<div class="input-field col s12">
<input id="age" type="text" class="validate">
<label for="age">Возраст</label>
</div>
</div>
<div class="row">
<div class="input-field col s12">
<input id="sex" type="text" class="validate">
<label for="sex">Пол</label>
</div>
</div>
<div class="row">
<div class="input-field col s12">
<input id="county" type="text" class="validate">
<label for="county">Гражданство</label>
</div>
</div>
<div>
<p><input type="checkbox" id="education" onclick="myFunction()"> <label for="education">Высшее образование</label> </p>
<p><input type="checkbox" id="mathematic" > <label for="mathematic">Знание математики</label> </p>
<p><input type="checkbox" id="asu" onclick="myFunction()"> <label for="asu">Знание АСУ ТП</label> </p>
<p><input type="checkbox" id="cutting" > <label for="cutting">Знание теории резания</label> </p>
<p><input type="checkbox" id="programming" > <label for="programming">Знание программирования</label> </p>
<p><input type="checkbox" id="gcode" > <label for="gcode">Написание ВП на G-код</label> </p>
<p id="startTest" class="my-button"> <b> Заполнить анкету </b> </p>
<div id="div1">(если хотите узнать, подходите ли вы нам)</div>
</div>
<script type="text/javascript" src="js/main.js"></script>
</div>
</div>

<div class="col s7">
<img id="background" style="background-image: url('https://pichold.ru/wp-content/uploads/2022/11/34.gif'); background-size: cover; height: 100vh;">
</div>
</div>
</body>
<style>
p {
font-size: 16px;
margin-bottom: 10px;
}
button {
padding: 10px 20px;
background-color: #FF00FF;
color: #fff;
border: none;
border-radius: 4px;
cursor: pointer;
}

button:hover {
background-color: #7fff00;
}
</style>
</body>
<h2><button onclick="window.location.href = '../index.html';">Назад</button></h2>
</body>
</html>

```

```

var trigger=0, name, age, sex, county, education, mathematic, asu, cutting, programming, grade; // объявляю переменные которые понадобятся в ходе работы
document.getElementById("startTest").addEventListener("click",test) // добавляю событие клика в элемент с id startTest
function test() { // функция которая выполняет необходимые действия
    name = prompt("Введите имя");
    age = prompt("Введите возраст");
    sex = prompt("Введите пол");
    county = prompt("Введите район");
    education = confirm("Вы еще хотите образование?");
    mathematic = confirm("Вы хотите математику?");
    asu = confirm("Вы хотите АСУ ТП?");
    cutting = confirm("Вы хотите курсы резки?");
    programming = confirm("Вы хотите программирование?");
    grade = confirm("Вы еще хотите повышение зарплаты на 0.5%?");
    writeToPage();
}
function writeToPage() { // функция которая добавляет данные из наших переменных в систему на страницу
    /*
    document.getElementById("name") - возвращает элемент с id name
    document.getElementById("name").innerHTML - вернет этот элемент
    У ЭЛЕМЕНТОВ DIV ТИПА, которые мы хотим вставить checked значение true или false
    document.getElementById("education").checked = education;
    */
    $('#name').val(name)
    $('#age').val(age)
    $('#sex').val(sex)
    $('#county').val(county)
    $('#county').focus();
    $('#name').focus();
    $('#age').focus();
    $('#name').focus();
    document.getElementById("education").checked = education;
    document.getElementById("mathematic").checked = mathematic;
    document.getElementById("asu").checked = asu;
    document.getElementById("cutting").checked = cutting;
    document.getElementById("programming").checked = programming;
    document.getElementById("grade").checked = grade;
    trigger=1;
    myFunction();
}
/* style.display = "none";
function myFunction() { // эта функция автоматически проверит документ на все значения: "хотите ли вы АСУ ТП и резку
    //обработка
    var a = document.getElementById("div");
    if (trigger==1) {
        if (document.getElementById("education").checked ==true && document.getElementById("asu").checked ==true && document.getElementById("grade").checked ==true) {
            var text = 'p class="shake">ОН ХОД ДОПОЛНИТЕЛЬНО';
            a.innerHTML = text;
            a.style.display = "block";
        }
        else {
            var text = 'p class="shake">ОН ХОД НЕ СТЫКИВАЕТСЯ';
            a.innerHTML = text;
        }
    }
}
}

```

Имя

Возраст

Пол

Район

Образование

☐ Высшее образование

☐ Высшее математика

☐ Высшее АСУ ТП

☐ Высшее курсы резки

☐ Высшее программирование

☐ Повышение 0.5% от ЗП

Дополнительно

(если хотите узнать, подведите ли вы итог)

Назад



Имя

Возраст

Пол

Гражданство

☐ Высшее образование

☐ Знание английского

☐ Знание ASP .NET

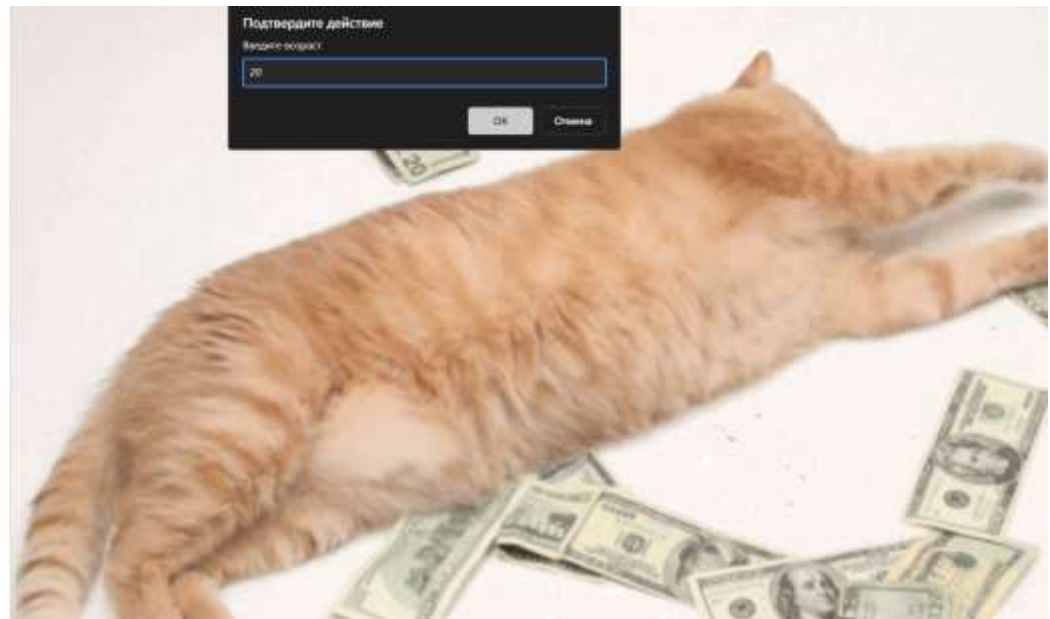
☐ Знание теории решения

☐ Знание программирования

☐ Неполное УТИ не в к-д

**Заполнить анкету**  
(если хотите узнать, подходит ли вам)

**Назад**



Имя

Елизавета

Возраст

20

Пол

Ж

Гражданство

RF

☒ Высшее образование

☒ Знание английского

☐ Знание ASP .NET

☐ Знание теории решения

☒ Знание программирования

☐ Неполное УТИ не в к-д

**Заполнить анкету**

**ВЫ НАМ НЕ ПОДОШЛИ**

**Назад**



Имя

Елизавета

Возраст

20

Пол

Ж

Гражданство

RF

☒ Высшее образование

☒ Знание английского

☒ Знание ASP .NET

☒ Знание теории решения

☒ Знание программирования

☒ Неполное УТИ не в к-д

**Заполнить анкету**

**ВЫ НАМ ПОДОШЛИ**

**Назад**



### Задание на повышенную оценку

3) Сделать радугу из 7ми ячеек таблицы. По нажатию на кнопку радуга циклически «опускается» вниз, т.е. цвета радуги бегают по кругу снизу-вверх таблицы. Так же сделать кнопку, чтобы радуга бегала автоматически с актом 0,3 секунды. HTML+JS.

```
doplnd.html x
1  <!DOCTYPE html>
2  <html>
3  <head>
4    <title>Радужная дискотека</title>
5    <style>
6      table {
7        border-collapse: collapse;
8        width: 100%;
9      }
10
11     td {
12       width: 14.28%;
13       height: 50px;
14     }
15   </style>
16 </head>
17 <body>
18   <table id="rainbowTable">
19     <tr>
20       <td></td>
21       <td></td>
22       <td></td>
23       <td></td>
24       <td></td>
25       <td></td>
26       <td></td>
27     </tr>
28   </table>
29
30   <button id="manualButton">Добавить новый цвет радуги</button>
31   <button id="autoButton">Запустить/Остановить радугу</button>
32
33   <script src="script.js"></script>
34 </body>
35 <body>
36   <h2><button onclick="window.location.href = 'index.html';">Назад</button></h2>
37 </body>
38 </html>
```



```

1  const rainbowColors = ['#FF0000', '#FF7F00', '#FFFF00', '#00FF00', '#0000FF', '#4B0082', '#8B00FF'];
2  const table = document.getElementById('rainbowTable');
3  const manualButton = document.getElementById('manualButton');
4  const autoButton = document.getElementById('autoButton');
5
6  let intervalId;
7
8  function updateRainbow() {
9      const cells = table.getElementsByTagName('td');
10
11      // Move the rainbow down the table
12      for (let i = cells.length - 1; i > 0; i--) {
13          cells[i].style.backgroundColor = cells[i - 1].style.backgroundColor;
14      }
15      cells[0].style.backgroundColor = rainbowColors.shift();
16      rainbowColors.push(cells[0].style.backgroundColor);
17  }
18
19  manualButton.addEventListener('click', updateRainbow);
20
21  autoButton.addEventListener('click', () => {
22      if (!intervalId) {
23          intervalId = setInterval(updateRainbow, 300);
24      } else {
25          clearInterval(intervalId);
26          intervalId = null;
27      }
28  });

```



Добавить новый цвет радуги

Запустить/Остановить радугу

Назад



Добавить новый цвет радуги

Запустить/Остановить радугу

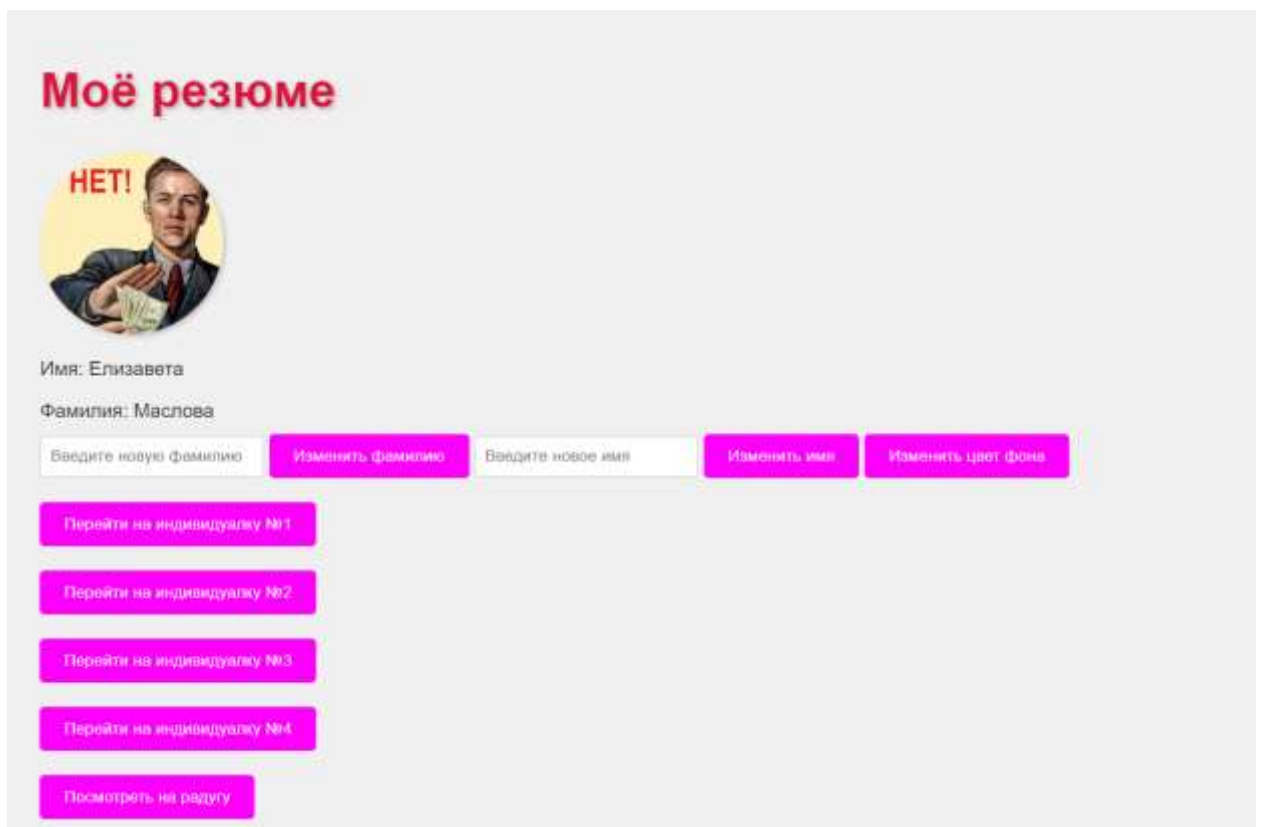
Назад

Так как мы делаем задание повышенной сложности, то объединим все задания в одно на одной странице (можно сделать пару страниц).



```
index.html x
1 <!DOCTYPE html>
2 <html lang="en">
3 <head>
4 <meta charset="UTF-8">
5 <meta name="viewport" content="width=device-width, initial-scale=1.0">
6 <title>Моё песенко</title>
7 <style>
8     body {
9         font-family: Arial, sans-serif;
10        background-color: #f0f0f0;
11        color: #333;
12        padding: 20px;
13    }
14
15    h1 {
16        font-size: 40px;
17        color: #DC143C;
18        text-shadow: 2px 2px 4px rgba(0, 0, 0, 0.2);
19    }
20
21    p {
22        font-size: 16px;
23        margin-bottom: 10px;
24    }
25
26    input[type="text"] {
27        padding: 8px;
28        border: 1px solid #ccc;
29        border-radius: 4px;
30    }
31
32    button {
33        padding: 10px 20px;
34        background-color: #FF00FF;
35        color: #fff;
36        border: none;
37        border-radius: 4px;
38        cursor: pointer;
39    }
40
41    button:hover {
42        background-color: #7fff00;
43    }
44 </style>
45 </head>
```

```
46 <body>
47 <h1>Моё песенко</h1>
48 <img alt="https://avatars.dzeninfra.ru/get-ava_dno/1590239/pub_h6f4a6a28a00b8f176437_b63fa3066a4b76387a5a06/size_1200" alt="ava fove" style="width: 150px; height: 150px;"/>
49 <p><span id="name">Иванов Иван</span></p>
50 <p><span id="surname">Иванов</span></p>
51 <input type="text" id="newSurname" placeholder="Введите новое имя" />
52 <button onclick="changeSurname()">Изменить</button>
53 <input type="text" id="newName" placeholder="Введите новое имя" />
54 <button onclick="changeName()">Изменить</button>
55 <button onclick="changeBackgroundColor()">Изменить цвет фона</button>
56
57 <script>
58 function changeSurname() {
59     var newSurname = document.getElementById("newSurname").value;
60     document.getElementById("name").innerText = newSurname;
61 }
62
63 function changeName() {
64     var newSurname = document.getElementById("newName").value;
65     document.getElementById("surname").innerText = newSurname;
66 }
67
68 function changeBackgroundColor() {
69     var randomColor = '#' + Math.floor(Math.random()*16777215).toString(16);
70     document.body.style.backgroundColor = randomColor;
71 }
72 </script>
73 </body>
74 <div>
75 <button onclick="window.location.href = '181.1.html'">Перейти на индивидуальку #1</button></div>
76 <button onclick="window.location.href = '181.2.html'">Перейти на индивидуальку #2</button></div>
77 <button onclick="window.location.href = '181.3.html'">Перейти на индивидуальку #3</button></div>
78 <button onclick="window.location.href = '181.4.html'">Перейти на индивидуальку #4</button></div>
79 <button onclick="window.location.href = '181.5.html'">Перейти на индивидуальку #5</button></div>
80 </div>
81 </html>
```



**Вывод:** В ходе лабораторной работы мы научились создавать простые страницы сайтов используя Язык HTML JavaScript и CSS.

### **Список использованной литературы:**

1. «Большая книга CSS3», Дэвид Макфарланд. Изд.: Питер, 2016.
2. «JavaScript», Дэвид Макфарланд. Изд.: Символ-Плюс, 2013.
3. «HTML, XHTML and CSS», Andy Harris. Изд.: Машиностроение, 2012.
4. <https://ilizmas.github.io>