

Министерство науки и высшего образования Российской Федерации
Федеральное государственное бюджетное образовательное учреждение

высшего образования

ТОМСКИЙ ГОСУДАРСТВЕННЫЙ УНИВЕРСИТЕТ СИСТЕМ

УПРАВЛЕНИЯ И РАДИОЭЛЕКТРОНИКИ (ТУСУР)

Кафедра КСУП

Отчет по лабораторной работе

по дисциплине «Веб-технологии»

**Тема: «HTML-элементы для создания интерактивных
приложений. Знакомство с JavaScript. Внешние
скрипты.»**

Студент гр. 571- 2

Вьюгин Кирилл Вадимович

4 декабря 2022 г.

Томск 2022

Оглавление

Оглавление

Введение	3
Основная часть.....	5
1 Реализация скриптов «Часики» и «Таймер».	5
2 Реализация скрипта для перехода по страницам.	7
3 Задания по вариантам.....	8
3.1 Задание 1.....	8
3.2 Задание 2.....	10
3.3 Задание 3.....	13
3.4 Задание 4.....	14
Ответы на контрольные вопросы.....	16
Заключение.....	18

Введение

Цель работы:

- знакомство с HTML-элементами для создания форм и меню;
- знакомство с DHTML;
- изучение языка JavaScript;
- изучение его возможностей создания динамических веб-страниц

Задачи лабораторной работы:

1. Для хранения внешних скриптов создайте специальную папку;
2. На каждой страничке реализуйте часики (реализуйте разные варианты отображения часиков);
3. Найдите скрипт, реализующий таймер. Внедрите его на одну из ваших страничек, сопроводив надписью, например, «На моей страничке Вы уже находитесь ... секунд»;
4. Реализуйте вертикальное и горизонтальное меню на одной странице. Если в предыдущих лабораторных работах уже было реализовано меню, то реализуйте обработку нажатия пункта меню;
5. Выполните задания, представленные ниже;
6. Оформите отчет, в котором перечислите, объясните назначение и опишите созданные скрипты;
7. Сгенерировать последовательность с помощью датчика случайных чисел и обработать согласно варианту задания; (5 вариант)
8. Выполнить задание на работу с датой и временем согласно варианту задания; (5 вариант)
9. Напишите функции-обработчики для событий согласно варианту задания; (5 вариант)
10. Напишите скрипт согласно варианту задания. (5 вариант)

Теоретическая часть

Эффектными страницы получаются, если использовать при их создании следующие возможности: управление элементами HTML-документов или даже самим браузером, генерацию новых документов, организацию диалогового взаимодействия с пользователем, выполнение каких-то расчетов и обработку данных. Для реализации данных возможностей используется DHTML – динамический HTML, в котором предусмотрена интеграция со специальными языками программирования. Программы, написанные на этих языках, называются сценариями (scripts).

Из языков, разработанных для программирования сценариев, можно привести JavaScript, VBScript, JScript как наиболее простые. В настоящих лабораторных работах будут рассмотрены приемы программирования только на языке JavaScript.

Язык программирования сценариев JavaScript является значительно упрощенной версией популярного языка Java. В отличие от Java, JavaScript является интерпретируемым (как и другие языки программирования сценариев). Это значит, что при загрузке HTML-документа браузер находит в нем все программы JavaScript, проверяет их на наличие ошибок, после чего выполняет команду за командой.

Таким образом, целью лабораторной работы является знакомство с элементами для создания форм и меню, знакомство с DHTML, изучение языка JavaScript, изучение его возможностей для создания динамических вебстраниц.

Основная часть

1 Реализация скриптов «Часики» и «Таймер».

```
var seconds = 0;
var minutes = 0;
var hours = 0;

function tick(){
    seconds++;
    if (seconds >= 60) {
        seconds = 0;
        minutes++;
        if (minutes >= 60) {
            minutes = 0;
            hours++;
        }
    }
}

function add() {
    tick();
    document.getElementById("timer").textContent = "Вы находитесь на этой страничке " +
        (hours > 9 ? hours : "0" + hours) + ":" + (minutes > 9 ? minutes : "0" + minutes) + ":" + (seconds > 9 ? seconds : "0" + seconds);
    timer();
}

function timer() {
    setTimeout(add, 1000);
}
```

Листинг 1.1 – реализация скрипта для таймера.

```
function time ()
{
    var time=new Date();
    var clock =
        (time.getHours() > 9 ? time.getHours() : "0"+time.getHours())
        +":"+time.getMinutes() > 9 ? time.getMinutes() :
"0"+time.getMinutes()
        +":"+time.getSeconds() > 9 ? time.getSeconds() :
"0"+time.getSeconds();
    var date = time.getDate() + "." + time.getMonth() + "." +
time.getFullYear();
    document.getElementById("clock").textContent = "Текущее время: " + clock;
    document.getElementById("date").textContent = "Дата: " + date;
    setTimeout('time()',1000);
}
```

Листинг 1.2 – реализация скрипта для часиков.

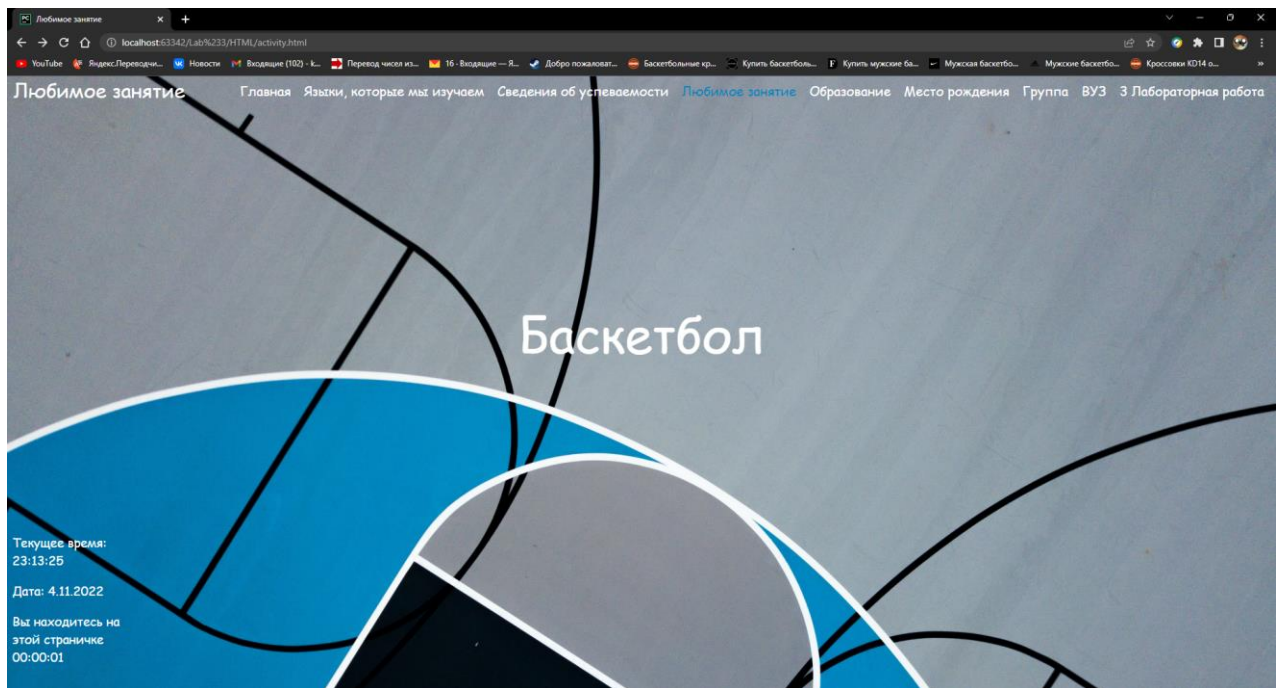


Рисунок 1.1 – работа часов и таймера.

2 Реализация скрипта для перехода по страницам.

```
function transition(href)
{
    window.location.href = href;
}
```

Листинг 2.1 – реализация скрипта для перехода по страницам.

```
<li onclick="transition('hometown.html')" class="hometownActivity">Место
рождения</li>
<li onclick="transition('education.html')" class="hometownLiA">Образование</li>
<li onclick="transition('activity.html')" class="hometownLiA">Любимое
занятие</li>
<li onclick="transition('informationAboutAcademicPerformance.html')"
class="hometownLiA">Сведения об успеваемости</li>
<li onclick="transition('programmingLanguage.html')" class="hometownLiA">Языки,
которые мы изучаем</li>
<li onclick="transition('firstlab.html')" class="hometownLiA">Главная</li>
```

Листинг 2.2 – применение скрипта для перехода по страницам.

3 Задания по вариантам.

3.1 Задание 1.

Задание: Известно сопротивление каждого из элементов электрической цепи. Все элементы соединены параллельно. Определить общее сопротивление цепи.

```
function task1()
{
    var count = document.form1.countElement.value;

    var text = document.form1.arrayInput.value;

    var array = text.split(", ");
    document.getElementById("answer").textContent = "";

    for (var i = 0; i < array.length; i++)
    {
        if (isNaN(parseInt(array[i])))
        {
            document.getElementById("answer").textContent =
                "Ошибка. Введите корректные данные";
            return;
        }

        array[i] = parseInt(array[i]);
    }

    var Result = 0;

    for(var i = 0; i < array.length; i++)
    {
        Result += 1/array[i];
    }

    Result = 1/Result;

    document.getElementById("answer").textContent =
        "Общее сопротивление цепи: " + Result;
}
```

Листинг 3.1.1 – реализация скрипта.

```
<div class="col-1-1">
<h2>Задание 1</h2>
    <form name="form1">
        <h2>Введите количество элементов электрической цепи: </h2>
        <input type="number" name="countElement">
        <h2>Введите сопротивление элементов электрической цепи: <br> В виде: 1,
2, 3, 4</h2>
        <input type="text" name="arrayInput">
        <input type="button" onclick="task1()" value="Проверить">
        <h2 id="answer"></h2>
    </form>
</div>
```

Листинг 3.1.2 – применение скрипта.

Задание 1

Введите количество элементов электрической цепи:

Введите сопротивление элементов электрической цепи:

В виде: 1, 2, 3, 4

Проверить

Рисунок 3.1.3 – результат работы.

3.2 Задание 2.

Задание: Реализуйте скрипт, позволяющий выводить расписание занятий на заданный день в зависимости от недели (четная, нечетная) и дня недели. Можно реализовать вывод параметров занятия (название и аудитория) в зависимости текущего времени.

```
function EvenOrNotEvenPrint()
{
    var year = new Date().getFullYear();
    var month = new Date().getMonth();
    var today = new Date(year, month, 0).getTime();
    var now = new Date().getTime();
    var week = Math.ceil((now - today) / (1000 * 60 * 60 * 24 * 7));
    if (week % 2)
    {
        document.write("Неделя нечетная");
    }
    else
    {
        document.write("Неделя четная");
    }
}

function EvenOrNotEven()
{
    var flag;
    var year = new Date().getFullYear();
    var month = new Date().getMonth();
    var today = new Date(year, month, 0).getTime();
    var now = new Date().getTime();
    var week = Math.ceil((now - today) / (1000 * 60 * 60 * 24 * 7));
    if (week % 2)
    {
        flag = false;
    }
    else
    {
        flag = true;
    }

    return flag;
}

function TimetableEven(Index)
{
    even = ["ТСиСА 13:15-14:50", "Веб-технологии 16:45-18:20", "Веб-технологии 18:30-20:05", "Веб-технологии 20:15-21:50"],
    ["Структуры данных 13:15-14:50", "ЭдпФКиС 15:00-16:35", "Структуры данных 16:45-18:20", "Структуры данных 18:30-20:05"],
    ["ТВиМС 13:15-14:50", "ООП 16:45-18:20"],
    ["ТВиМС 13:15-14:50", "ППНИРР 15:00-16:35", "ОПД 20:15-21:50"],
    ["Иностранный язык 13:15-14:50", "ЭдпФКиС 15:00-16:35"],
    []

    var array = 0;
    array = even[Index];
    array.forEach((item, i, array) => { document.write(array[i] + "<br>"); })
}
```

```

function TimetableNotEven(Index)
{
    notEven = [
        ["ТСиСА 08:50-10:25", "ТСиСА 10:40-12:15"],
        ["Education design 13:15-14:50", "ЭДпФКиС 15:00-16:35"],
        [],
        ["ТВиМС 15:00-16:35", "ОПД 20:15-21:50"],
        ["Иностранный язык 13:15-14:50", "ЭДпФКиС 15:00-16:35", "ООП 16:45-18:20", "ООП 18:30-20:05"],
        []
    ]

    var array = 0;
    array = notEven[Index];
    array.forEach((item, i, array) => { document.write(array[i] + "<br>"); })
}

function Timetable(day)
{
    if (EvenOrNotEven())
    {
        TimetableEven(day);
    }
    else
    {
        TimetableNotEven(day);
    }
}

```

Листинг 3.2.1 – реализация скрипта.

```

<div class="col-1-2">
<div class="containerIn2">
    <h2>
        <script>
            EvenOrNotEvenPrint()
        </script>
    </h2>
</div>
    <div class="containerIn2">
        <h2>
            Понедельник
        </h2>
        <h3>
            <script>
                Timetable(0);
            </script>
        </h3>
    </div>
    <div class="containerIn2">
        <h2>
            Вторник
        </h2>
        <h3>
            <script>
                Timetable(1);
            </script>
        </h3>
    </div>
    <div class="containerIn2">
        <h2>
            Среда
        </h2>
        <h3>

```

```

    <script>
        Timetable(2);
    </script>
</h3>
</div>
<div class="containerIn2">
<h2>
    Четверг
</h2>
<h3>
    <script>
        Timetable(3);
    </script>
</h3>
</div>
<div class="containerIn2">
<h2>
    Пятница
</h2>
<h3>
    <script>
        Timetable(4);
    </script>
</h3>
</div>
<div class="containerIn2">
<h2>
    Суббота
</h2>
<h3>
    <script>
        Timetable(5);
    </script>
</h3>
</div>
<div class="containerIn2">
<h2>
    Воскресенье
</h2>
<h3>
    <script>
        Timetable(6);
    </script>
</h3>
</div>
</div>

```

Листинг 3.2.2 – применение скрипта.

Неделя нечетная						
Понедельник	Вторник	Среда	Четверг	Пятница	Суббота	Воскресенье
ТСИСА 08:50-10:25 ТСИСА 10:40-12:15	Education design 13:15-14:50 ЭдпфКиС 15:00-16:35		ТВиМС 15:00-16:35 ОПД 20:15-21:50	Иностранный язык 13:15-14:50 ЭдпфКиС 15:00-16:35 ООП 16:45-18:20 ООП 18:30-20:05		

Рисунок 3.2.3 – результат работы.

3.3 Задание 3.

Задание: Написать функцию, которая по нажатию на кнопку генерировала бы случайную цифру от 0 до 5 в сообщении: "Ваша случайная цифра: n"

```
function task3() {  
  
    var Result = Math.floor(Math.random() * 5);  
  
    alert('Ваша случайная цифра:' + ' ' + Result);  
  
}
```

Листинг 3.3.1 – реализация скрипта.

```
<div class="col-1-3">  
<h2>Задание 3</h2>  
<h2>Нажмите на кнопку для генерации случайного числа от 0 до 5: </h2>  
<button onclick="task3()">  
    <h2>  
        Сгенерировать  
    </h2>  
</button>  
</div>
```

Листинг 3.3.2 – применение скрипта.

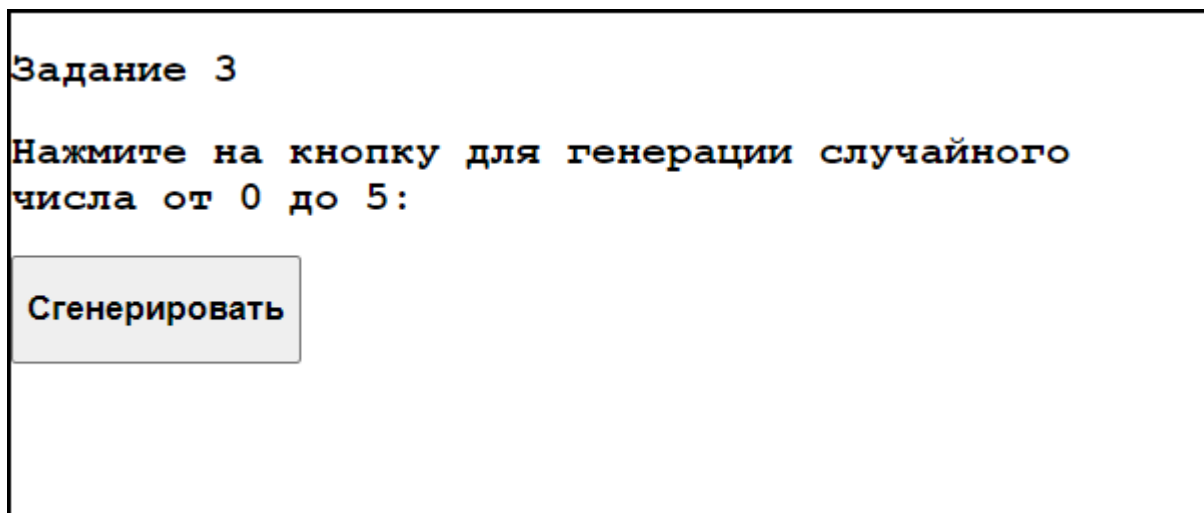


Рисунок 3.3.3 – результат работы.

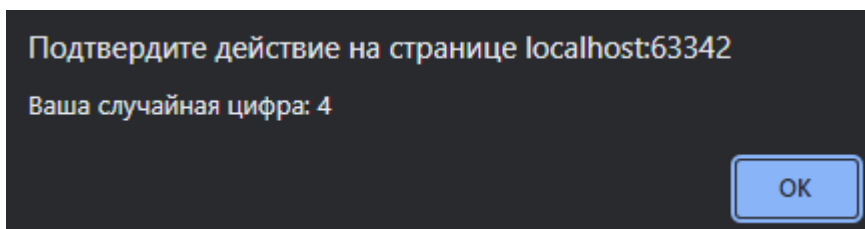


Рисунок 3.3.4 – результат работы.

3.4 Задание 4.

Задание: Создайте функцию, которая вызовет два запроса (prompt). Первый попросит пользователя ввести свое имя, второй — отчество. Затем та же функция должна вызвать окно предупреждения (alert) с текстом: Привет, имя отчество, добро пожаловать на адрес страницы, мою замечательную страницу!

```
function task4()
{
    var name = prompt('Как вас зовут?', "Имя");
    var patronymic = prompt('Ваше отчество?', "Отчество");
    var page_address = document.location.href;

    alert('Привет, ' + name + ' ' + patronymic + ', добро пожаловать на ' +
page_address + ', мою замечательную страницу!');
}
```

Листинг 3.4.1 – реализация скрипта.

```
<div class="col-1-4">
<h2>Задание 4</h2>
<button onclick="task4()">
    <h1>
        НАЖМИ!
    </h1>
</button>
</div>
```

Листинг 3.4.2 – применение скрипта.



Рисунок 3.4.3 – результат работы.

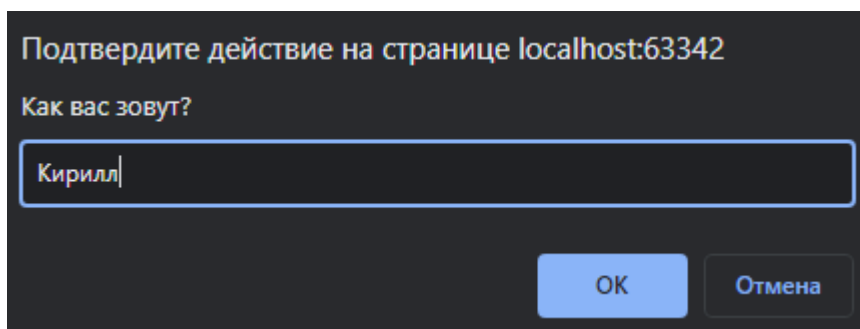


Рисунок 3.4.4 – результат работы.

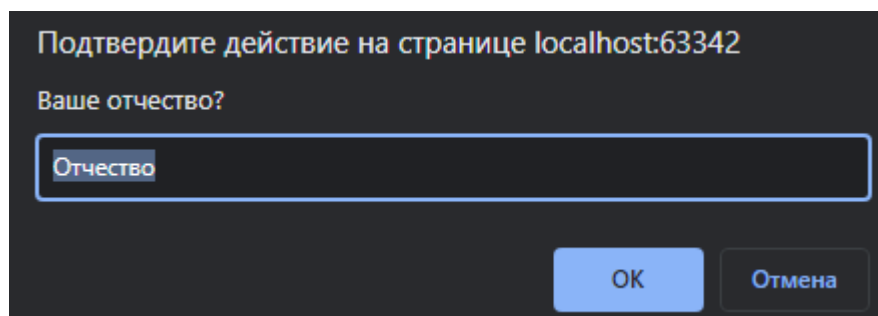


Рисунок 3.4.5 – результат работы.

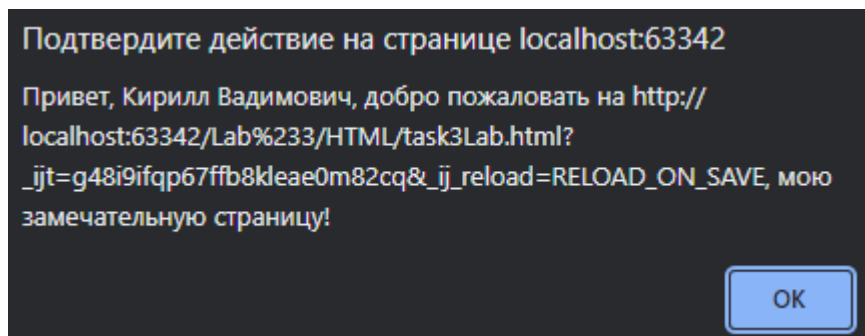


Рисунок 3.4.6 – результат работы.

Ответы на контрольные вопросы.

1. Какой тег используется для организации списка?
<select>
2. Какой тег используется для подписи элементов формы?
<label>
3. Какой тег используется для группировки элементов формы?
<optgroup>
4. Какой объект используется для работы с датой?
Date
5. Как создать объект Date?
var time = new Date();
6. Какие методы используются для получения текущей даты?
getDate, getHours, getMinutes, getSeconds, getDay, getMonth, getYear, getFullYear.
7. Какие методы используются для установки текущей даты?
setDate, setHours, setMinutes, setSeconds, setYear, setMonth, setTime
8. Что такое DHTML?
Динамический HTML, в котором предусмотрена интеграция со специальными языками программирования.
9. Для чего используется язык программирования JavaScript?
 - 1) Создание динамических страниц, т.е. страниц, содержимое которых может меняться после загрузки.
 - 2) Проверка правильности заполнения пользовательских форм.
 - 3) Решение «локальных» задач с помощью сценариев.
10. Опишите правила формирования имени переменной JavaScript?
Имя переменной должно начинаться с буквы латинского алфавита или с символа подчеркивания. Само имя может включать буквы латинского алфавита, цифры и знак подчеркивания.
11. Какой тег используется для размещения сценария в документе?
<script>
12. Опишите средства, используемые для отладки сценария
Весьма удобным средством отладки является Консоль, входящая в состав браузеров.
13. Опишите варианты исполнения сценария
При загрузке документа, при возникновении события, а также при вызове его интерпретатором.
14. Какой тег используется для организации формы?
<form>
15. Перечислите стандартные элементы управления
Стандартными являются все элементы управления, которые можно поместить на HTML-форму (однострочное текстовое поле, поле для

ввода пароля, флажки, переключатели, кнопки как пользовательские, так и выполняющие стандартные действия, поля имен файлов и т.д.)

16.Как организовать вертикальное и горизонтальное меню?

Как из вариантов, при помощи тега <table>

17.Как создать внутренний и внешний скрипт?

Внешний – создать в директории с проектом файл с расширением .js, после чего подключить его к странице внутри тега <head> при помощи тега <script> с аргументом src, который в свою очередь принимает путь до скрипта.

Внутренний – описать скрипт внутри тега <script>

Заключение

В ходе выполнения работы мною были изучены основы языка программирования JavaScript и возможности работы со страницей.