МИНИСТЕРСТВО ОБРАЗОВАНИЯ РЕСПУБЛИКИ БЕЛАРУСЬ

# УЧРЕЖДЕНИЙ ОБРАЗОВАНИЯ

«БРЕСТСКИЙ ГОСУДАРСТВЕННЫЙ ТЕХНИЧЕСКИЙ УНИВЕРСИТЕТ»

КАФЕДРА ЭВМиС

Лабораторная работа №3

“DOM, работа с событиями”

Выполнили:

студенты 3 курса

группы МС-2

Супрун А.И

Литвинович В.С

Проверил:

Луцюк П.Н

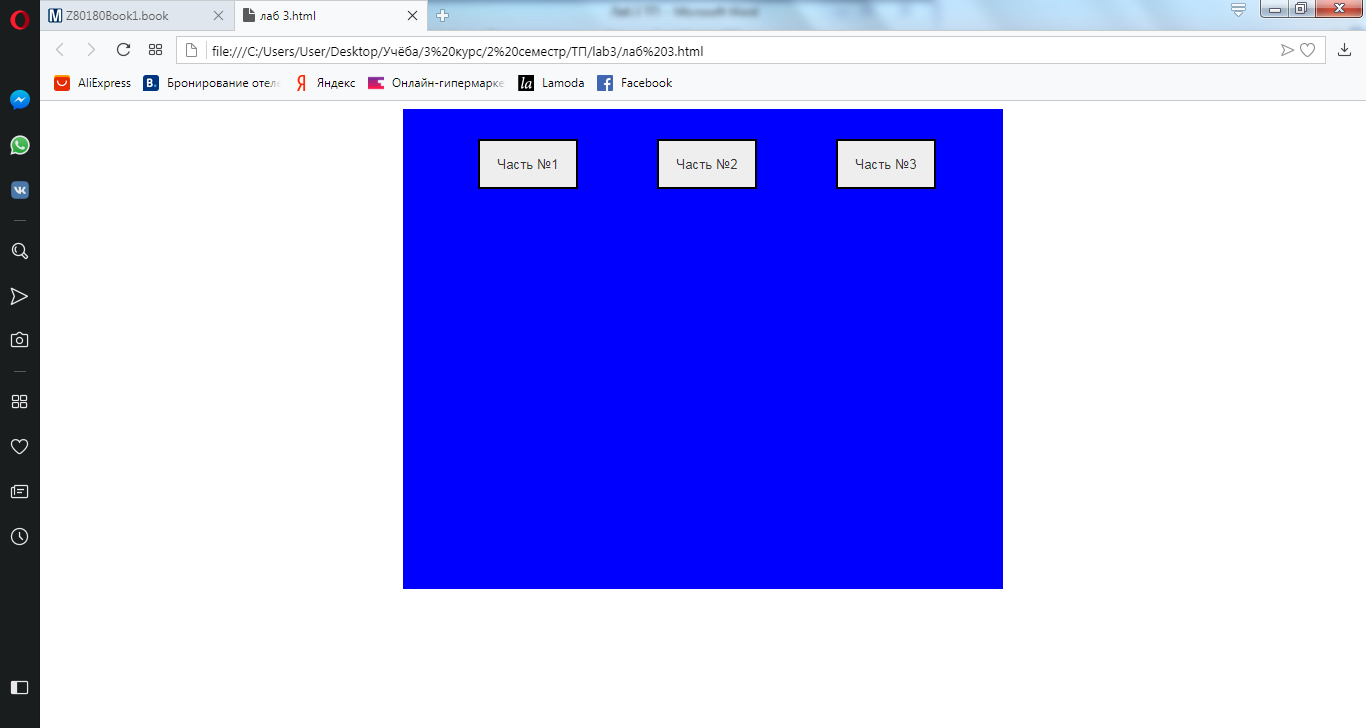
Брест, 2019

**Цель:** Изучить и на практике закрепить понятия: объектная модель документа и события.

**Ход работы:**

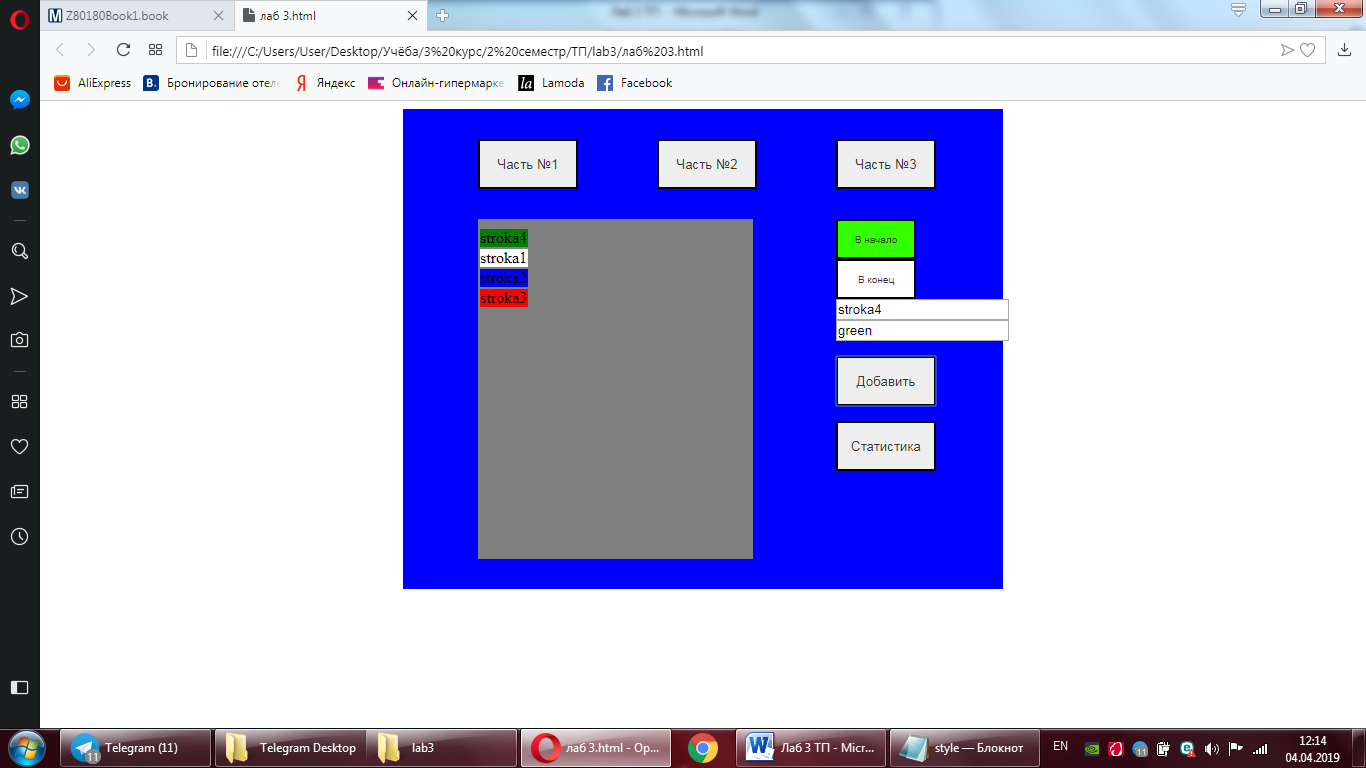
**Задание 1-4 (Часть 1) результат:**

В начале создаем общее начальное окно, которое включает в себя три кнопки, каждая из которых вызывает определенную часть задания.

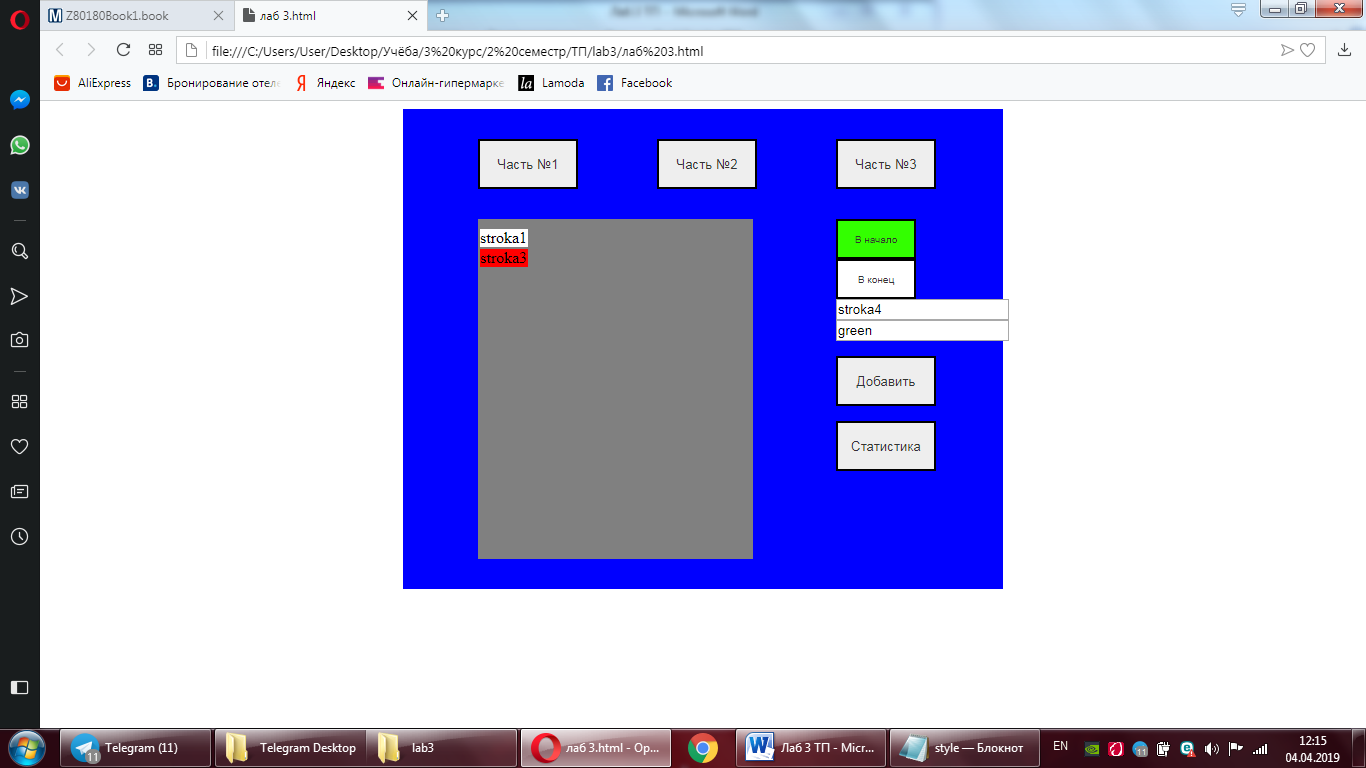


Как видно,после нажатия на кнопку «Часть №1» начинают отрисовываться необходимые элементы меню, необходимые для выполнения задания.

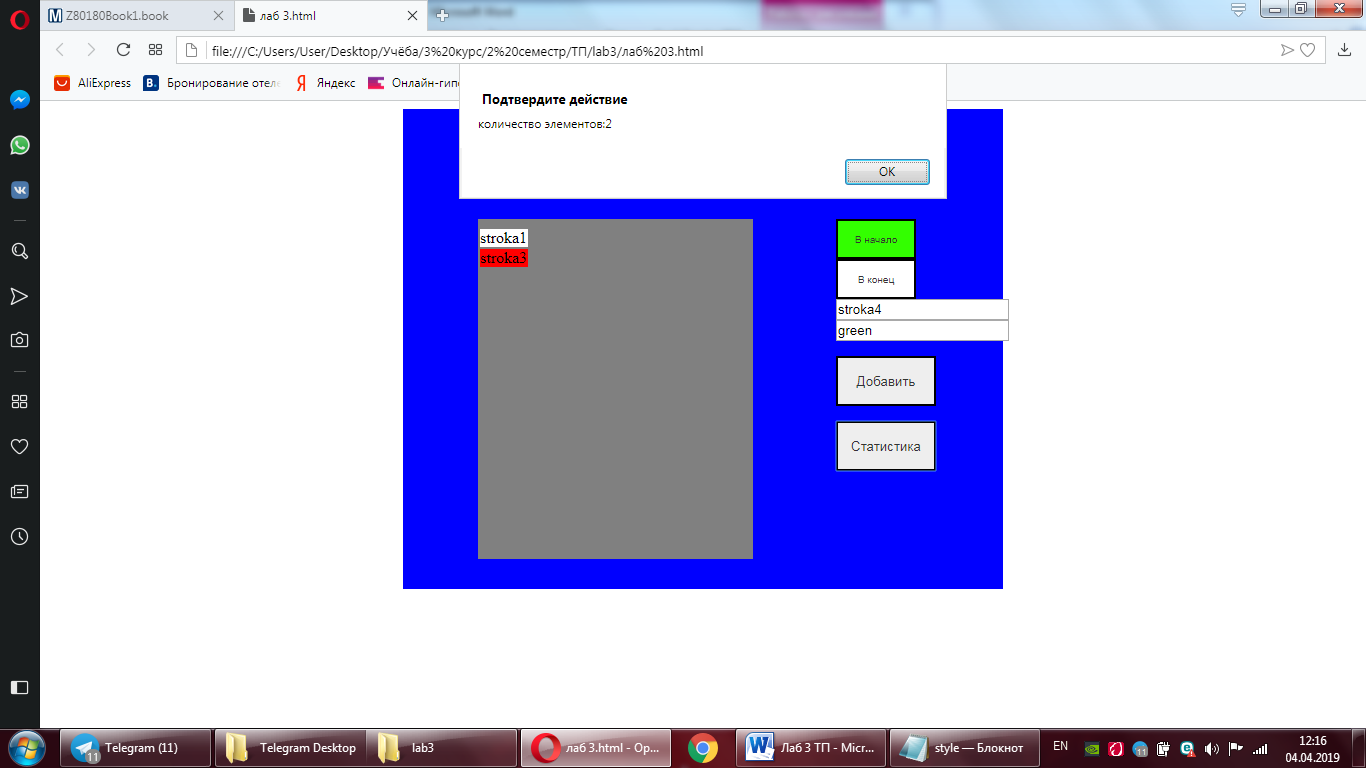
Теперь попробуем добавить три разные строки с разным цветом фона. Важно отметить что длина строки ограничена на 30 символов. Попробуем ввести 4 строку, но с режимом добавления строки в начало:



Теперь удалим «Строку2» и «Строку4» нажатием на них:

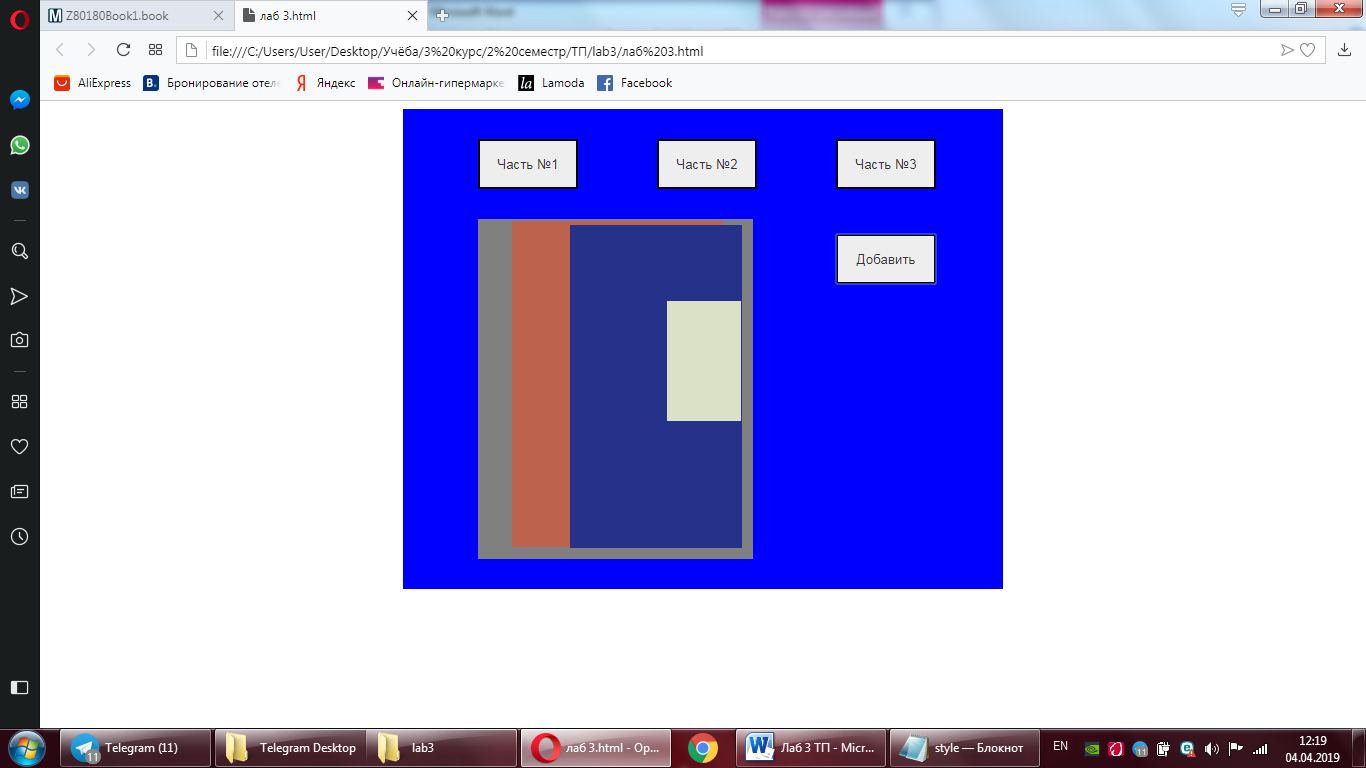


Теперь после нажатия на кнопку «Статистика» должно вызваться окно «alert» и вывести количество строк, находящихся в сером блоке:



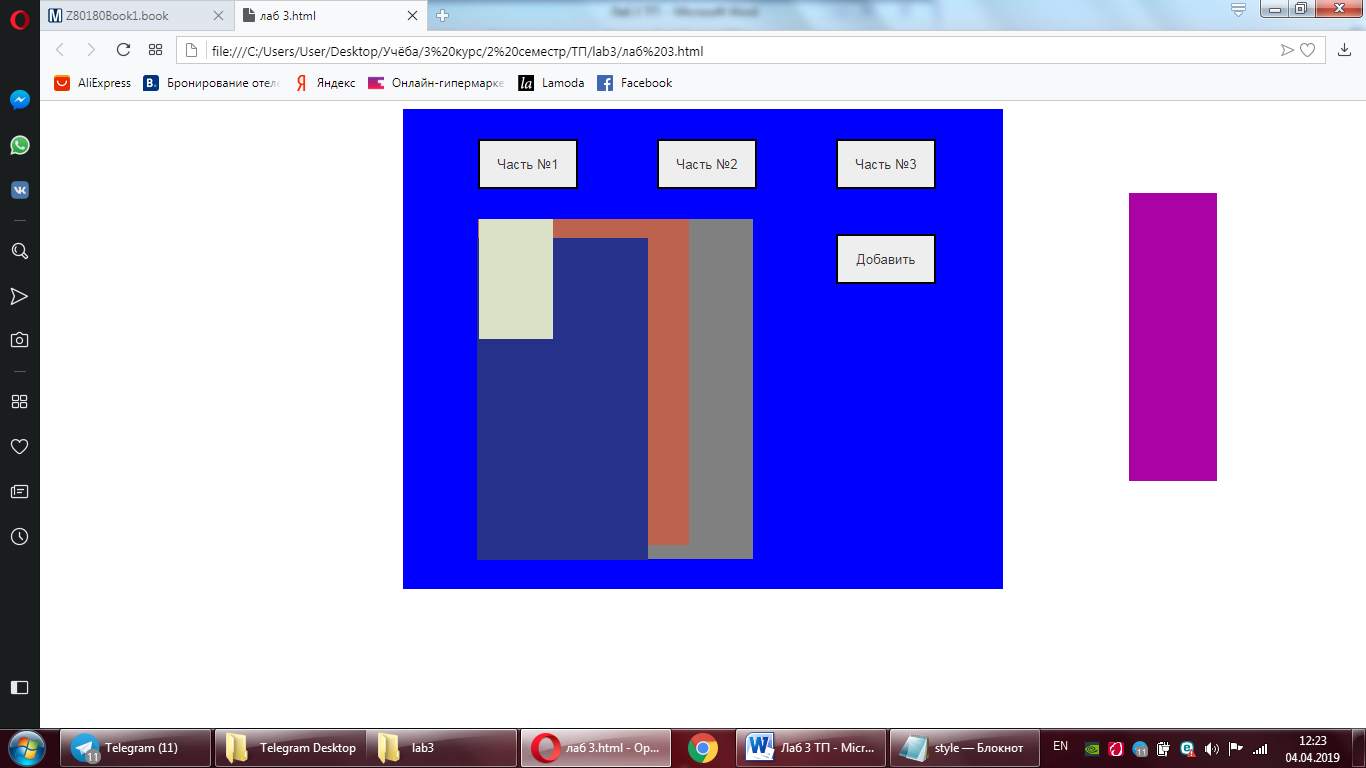
**Задание 5-9 (Часть 2) результат:**

Нажимаем на кнопку «Часть №2» и переходим ко второй части задания. Как видим ненужные элементы с первого задания убираются . Попробуем добавить 3прямоугольника с помощью кнопки добавить



Как видно все добавилось корректно в область и все прямоугольники имеют случайный размер и случайный цвет.

Теперь попробуем с помощью стрелочек на клавиатуре переместить эти треугольники в разные углы:

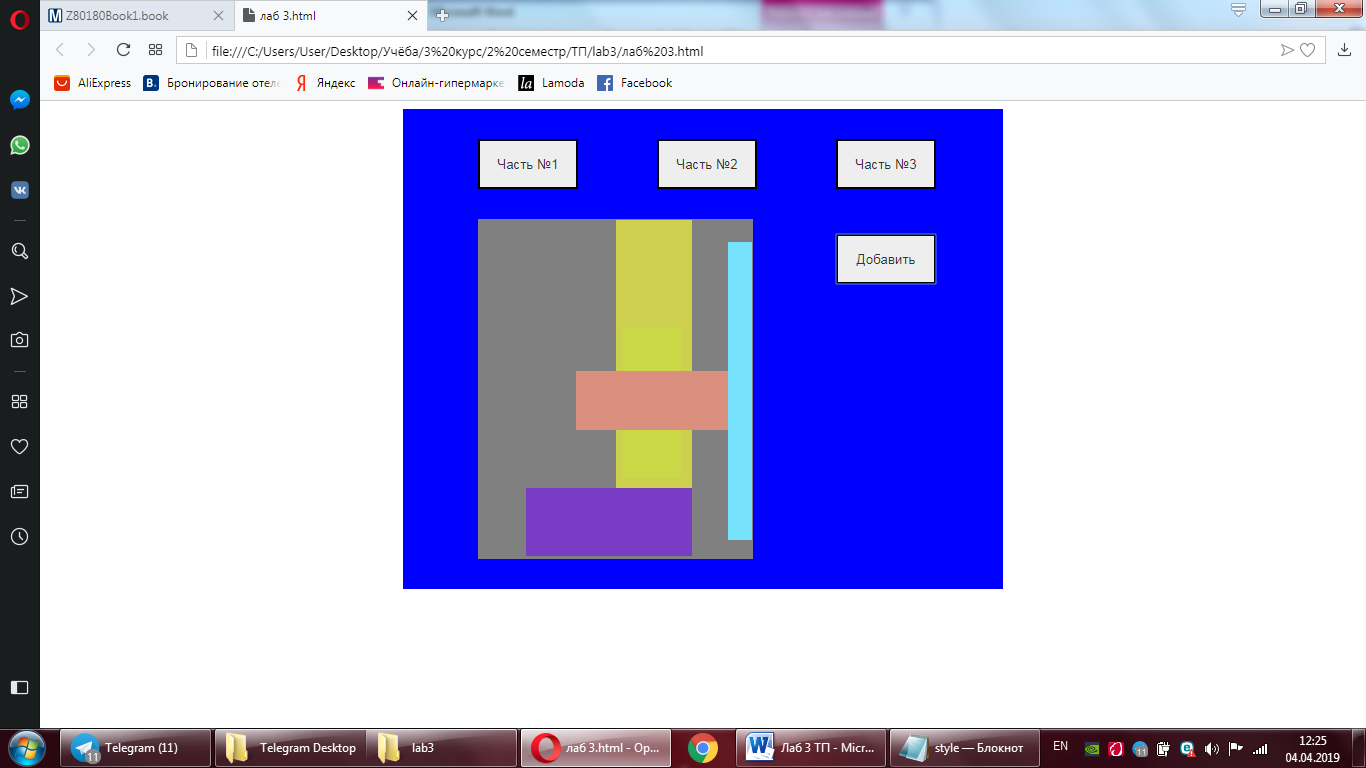


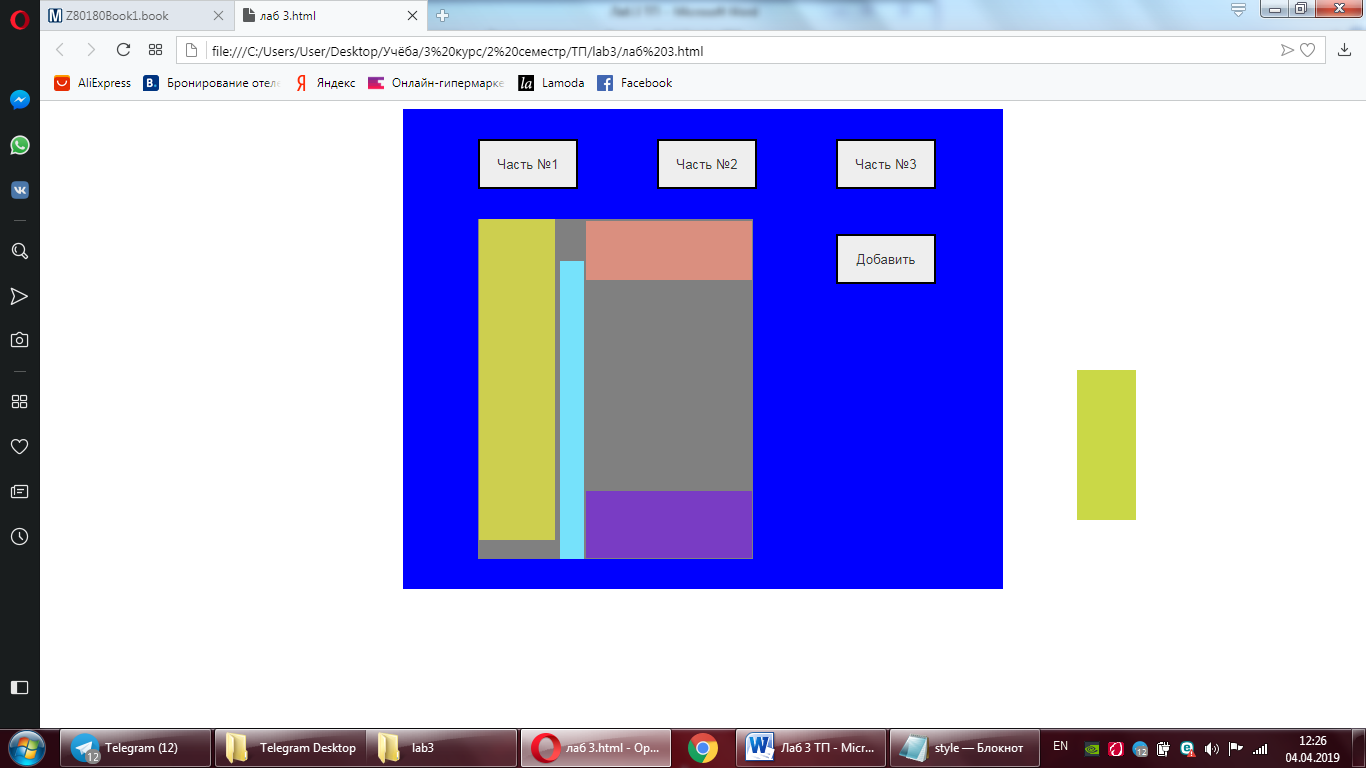
Как видно треугольники не вылезают за границы и перемещаются свободно, так же тот треугольник, который на данный момент перемещается, выделяется синей рамкой.

**Задание 9-11 (Часть 3) результат:**

Задания из части 3 похожи на задания из части 2, за исключением того, что перемещение треугольников нужно реализовать с помощью мышки (Drag and Drop):

Нажмем на кнопку «Часть №3», затем с помощью кнопки добавить добавим 4 прямоугольника и попробуем их переместить с помощью мышки.





Как видно перемещение мышкой работает.

**Исходный код HTML:**

<!DOCTYPE html>

<html>

<head>

<title>

</title>

<link rel="stylesheet" href="style.css">

</head>

<body>

<div id = "menu">

<button class="menubuttons">Часть №1</button>

<button class="menubuttons">Часть №2</button>

<button class="menubuttons">Часть №3</button>

</div>

<script type="text/javascript" src="script.js"></script>

</body>

</html>

**Исходный код СSS:**

#menu

{

height: 480px;

width: 600px;

background: blue;

margin-left: auto;

margin-right: auto;

}

.menubuttons

{

text-decoration: none;

background-color: #EEEEEE;

color: #333333;

width: 100px;

height: 50px;

border-style: solid solid solid solid;

border-color: black;

margin-top: 30px;

margin-left: 75px;

padding: 12px 12px 12px 12px;

}

.part1buttons

{

text-decoration: none;

background-color: #EEEEEE;

color: #333333;

width: 100px;

height: 50px;

border-style: solid solid solid solid;

border-color: black;

margin-top: 15px;

padding: 12px 12px 12px 12px;

}

.part1BeginEnd

{

font-size: 10px;

background-color: #EEEEEE;

color: #333333;

width: 80px;

height: 40px;

border-style: solid solid solid solid;

border-color: black;

padding: 12px 12px 12px 12px;

}

table{

position: relative;

top: -350px;

}

**Исходный код Java Script:**

function addtext (value,table,color)

{

if(i<=15)

{

var elem = document.createElement('tr');

elem.innerText = value;

if(napravlenie == 0 && i!=0)

{

tr = document.querySelector("tr");

table.insertBefore(elem,tr);

elem.style.backgroundColor = color;

}

else

{

elem.style.backgroundColor = color;

table.appendChild(elem);

}

i++;

elem.addEventListener('click',function(){table.removeChild(elem);i--;},false);

}

};

function moveBlock(event)

{

console.log(event);

if(event.keyCode == 26)

{

figure.style.left ="110px";

}

}

function getRandomColor() {

var letters = '0123456789ABCDEF';

var color = '#';

for (var i = 0; i < 6; i++) {

color += letters[Math.floor(Math.random() \* 16)];

}

return color;

}

function getRandomParam()

{

var param = [];

param[0] = Math.random()\*(340-1)+1; //height

param[1] = Math.random()\*(275-1)+1; // width

param[2] = Math.random()\*((275-param[1])-0)+0; //left

param[3] = Math.random()\*((340-param[0])-0)+0; //top

return param;

}

function addblock()

{

var param = getRandomParam();

var figure = document.createElement('div');

figure.style.backgroundColor = getRandomColor();

figure.style.height = param[0] +'px';

figure.style.width = param[1] +'px';

figure.style.left = param[2] +'px';

figure.style.top = param[3] +'px';

figure.style.position ='absolute';

figure.className ="figure";

box2.style.position ='absolute';

figure.setAttribute('tabIndex','0');

box2.appendChild(figure);

figure.onkeydown = function(event){

if(event.key =='ArrowRight' && event.srcElement.offsetLeft < (275 - param[1]))

{

var right = event.srcElement.offsetLeft + 1;

figure.style.left = right + 'px';

}

if(event.key == 'ArrowDown' && event.srcElement.offsetTop < (340 - param[0]))

{

var down = event.srcElement.offsetTop + 1;

figure.style.top = down + 'px';

}

if(event.key == 'ArrowUp' && event.srcElement.offsetTop > 0)

{

var up = event.srcElement.offsetTop - 1;

figure.style.top = up + 'px';

}

if(event.key == 'ArrowLeft' && event.srcElement.offsetLeft > 0)

{

var left = event.srcElement.offsetLeft - 1;

figure.style.left = left + 'px';

}

};

figure.onmousedown = function(event)

{

var shiftX = event.pageX - event.srcElement.offsetLeft;

var shiftY = event.pageY - event.srcElement.offsetTop;

function moveAt(event) {

figure.style.left = event.pageX - shiftX + 'px';

figure.style.top = event.pageY - shiftY + 'px';

};

document.onmousemove = function(event) {

moveAt(event);

}

figure.onmouseup = function() {

document.onmousemove = null;

figure.onmouseup = null;

}

};

}

function generalPart ()

{

if(start == 1)

{

textbox.setAttribute('maxlength', '35');

colorbox.setAttribute('maxlength', '7');

addButton.innerText = "Добавить";

statButton.innerText ="Статистика";

podpis1.innerText ="В начало";

podpis2.innerText ="В конец";

addButton.className ="part1buttons";

statButton.className ="part1buttons";

podpis1.className ="part1BeginEnd";

podpis2.className ="part1BeginEnd";

textbox.value='Текст';

textbox.at

colorbox.value ='Цвет';

box2.style.cssText =" height: 340px;\

width: 275px;\

background: gray;\

margin-left: 75px;\

margin-top: 30px;"

box[0].appendChild(box2);

box2.appendChild(box3);

box2.appendChild(table);

box3.style.cssText =" position:relative;\

left:358px;\

height: 358px; \

width:190px ";

box3.appendChild(podpis1);

box3.appendChild(podpis2);

box3.appendChild(textbox);

box3.appendChild(colorbox);

box3.appendChild(addButton);

box3.appendChild(statButton);

start = 0;

event();

}

if(part == 1)

{

var figures = document.getElementsByClassName('figure');

if(figures.length != 0)

{

for (var s = 0;s < figures.length; s++)

{

figures[s].style.display ='none';

}

}

textbox.style.display = 'block';

colorbox.style.display = 'block';

table.style.display = 'block';

podpis1.style.display = 'block';

podpis2.style.display = 'block';

statButton.style.display = 'block';

}

if(part == 2)

{

var figures = document.getElementsByClassName('figure');

if(figures.length != 0)

{

for (var s = 0;s < figures.length; s++)

{

figures[s].style.display ='block';

}

}

table.style.display = 'none';

podpis1.style.display = 'none';

podpis2.style.display = 'none';

statButton.style.display = 'none';

textbox.style.display = 'none';

colorbox.style.display = 'none';

}

if(part == 3)

{

table.style.display = 'none';

podpis1.style.display = 'none';

podpis2.style.display = 'none';

statButton.style.display = 'none';

textbox.style.display = 'none';

colorbox.style.display = 'none';

}

};

function add()

{

if(part == 1)

{

addtext(textbox.value,table,colorbox.value);

}

if(part == 2 || part == 3)

{

addblock();

}

}

function event()

{

var buttons = document.getElementsByTagName('button');

buttons[5].addEventListener('click',add,false);

buttons[3].addEventListener('click',function(){podpis1.style.background = '#33FF00'; napravlenie = 0; podpis2.style.backgroundColor = 'white';},false);

buttons[4].addEventListener('click',function(){podpis2.style.backgroundColor = '#33FF00'; podpis1.style.backgroundColor = 'white'; napravlenie = 1;},false);

buttons[6].addEventListener('click',function(){alert("количество элементов:"+i)},false);

}

var statButton = document.createElement('button')

,podpis1 = document.createElement('button')

,textbox = document.createElement('input')

,colorbox = document.createElement('input')

,podpis2 = document.createElement('button')

,table = document.createElement('table')

,box2 = document.createElement('div')

,addButton = document.createElement('button')

,box = document.getElementsByTagName('div')

,box3 = document.createElement('div');

var part = 0,start = 1, i = 0,napravlenie = 1;

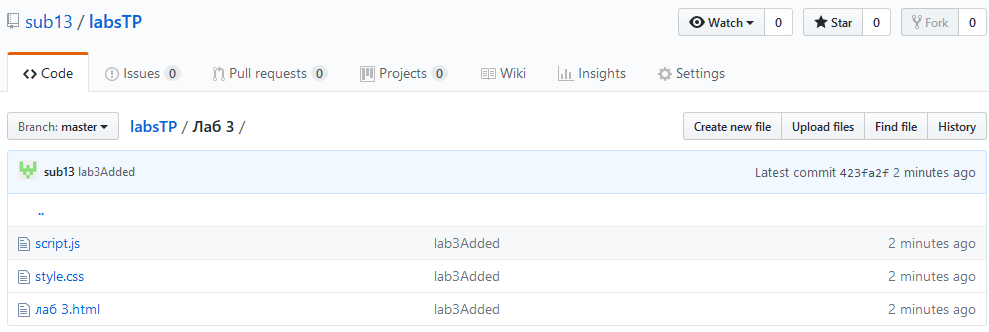
var buttons = document.getElementsByTagName('button');

buttons[0].addEventListener('click',function(){part = 1;generalPart();},false);

buttons[1].addEventListener('click',function(){part = 2;generalPart();},false);

buttons[2].addEventListener('click',function(){part = 3;generalPart();},false);

**Результат загрузки на gitgub.com**

****

**Вывод:**

Изучили основные особенности, основные принципы работы документной модели объектов, изучили события. На практике закрепили изученное. Загрузили разработанный проект в распределённую систему контроля версий «GIT».