МИНИСТЕРСТВО НАУКИ И ВЫСШЕГО ОБРАЗОВАНИЯ

РОССИЙСКОЙ ФЕДЕРАЦИИ

Федеральное государственное бюджетное

образовательное учреждение высшего образования

«Тверской государственный университет»

Факультет прикладной математики и кибернетики

Направление 01.03.02 – Прикладная математика и информатика

Профиль «Математическое моделирование»

Отчет по итогам производственной практики

(практика по получению профессиональных умений и опыта профессиональной деятельности)

2019-2020 уч. год, 6 семестр

**Автор**: студент 3 курса 34 гр.

Померанцев Олег Игоревич

**Руководитель практики:**

д.ф.-м.н., профессор

Зингерман К.М.

\_\_\_\_\_\_\_\_\_\_\_\_\_

*(подпись)*

**Оценка**: \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

Тверь – 2020

**Оглавление**

Введение ................................................................................................................ 3

Цели работы .......................................................................................................... 4

График практики .................................................................................................. 4

Описание задания ................................................................................................. 5

Описание выполнения ......................................................................................... 6

Приложения...........................................................................................................12

Вывод .................................................................................................................... 25

Список литературы .............................................................................................. 25

**Введение**

В рамках практики была поставлена задача обучиться методам front-разработки. Front-разработка предполагает создание клиентской части программного продукта. То есть непосредственно той части, с которой работает пользователь.

Front-разработка является важной частью создания программного продукта. Недостаточно создать продукт, способный корректно выполнять поставленную задачу. Необходимо так же дать пользователю доступ ко всему функционалу этого продукта, с учетом того, что пользователь, скорее всего, не является программистом.

Важным навыком для такой разработки является умение работать с языком верстки HTML. Этот язык позволяет создавать веб-страницы с необходимым дизайном. Он является достаточно удобным и гибким инструментов в создании необходимого графического интерфейса пользователя.

Для придания веб-странице уникального вида, который заинтересует пользователя и облегчит ему работу с программным продуктом, используется язык описания внешнего вида документа CSS. Он позволяет задать HTML странице необходимый дизайн, при этом не изменяя основной файл.

Однако, по-настоящему живое взаимодействие пользователя и программного продукта может достигаться только с помощью языка JavaScript. С его использованием на страницу можно вносить интерактивные элементы, значительно упрощающее взаимодействие пользователя с программным продуктом.

Для front-разработчика необходимо уметь пользоваться всеми тремя языками – HTML, CSS и JavaScript. Во время практики было произведено обучение всем тремя языкам, а так же решен ряд задач на проверку качества полученных навыков.

**Цели работы**

На время производственной практики были поставлены следующие задачи:

* Получить общее представление о языках, необходимых для front-разработки: HTML, CSS и JavaScript;
* Изучить и освоить методы разработки с помощью данных языков;
* Решить поставленные задачи для демонстрации полученных навыков и закрепления пройденного материала;
* Ознакомится с процессом front-разработки на личном опыте.

**График практики**

01.06.2020 – ознакомление с программой практики, вводная лекция.

02.06.2020 – краткая информация про HTML, CSS, JavaScript

03.06.2020 – подробнее про современный front-end

04.06.2020–09.06.2020 – самостоятельное выполнение практического задания, индивидуальные консультации в удобное время.

10.06.2020 – встреча с руководителем практики и демонстрация завершённого задания.

11.06.2020 – Подведение итогов работы.

**Описание задания**

С целью поэтапного освоения материала, начиная с основных понятий и переходя к более продвинутым, задание разделено на три части.

Первая часть рассчитана на освоение базовых принципов работы с CSS, HTML и JavaScript. Задание включает в себя 6 задач на создание веб-страниц средствами HTML и CSS, а так же 3 задачи на создание веб-страниц с интерактивным функционалом JavaScript.

Вторая часть состоит из одного задания. В нем необходимо создать макет веб-страницы для программы дистанционного банковского обслуживания. Для выполнения этого задания требуется применить все навыки HTML, JavaScript и CSS, полученные в ходе выполнения первой части.

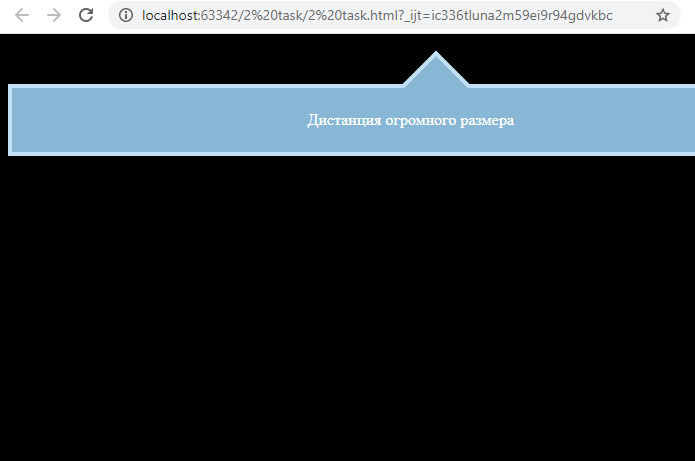
Третья часть предполагает перенос задания из части два на фреймворк

React.

**Описание выполнения**

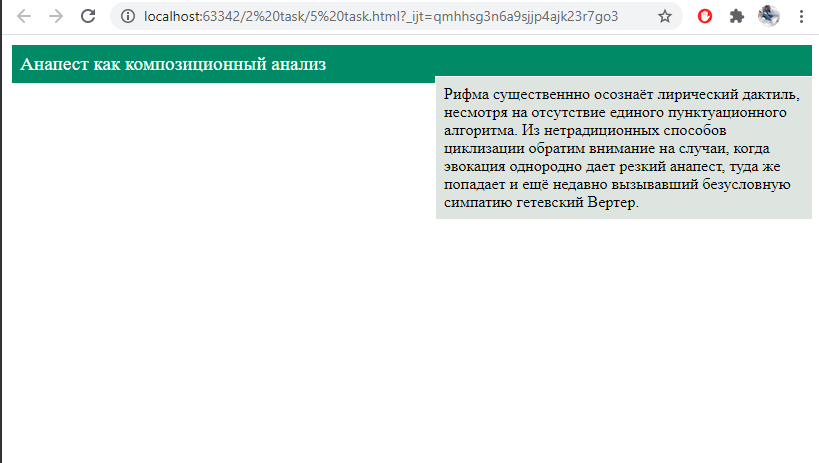
Выполнение заданий практики производилось в порядке, составленном руководителем практики. В первую очередь, были решены задачи на освоение языков HTML и JavaScript.

Первое задание: сверстать страницу. Ширина блока резиновая и меняется в зависимости от размеров окна браузера. Указатель всегда располагается посередине блока, размеры указателя фиксированы.



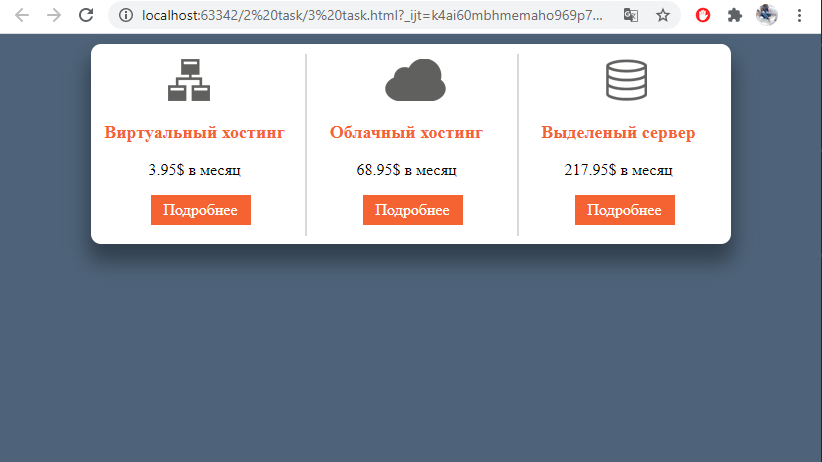
Программная реализация в приложении.

Второе задание: сверстать страницу на HTML5.



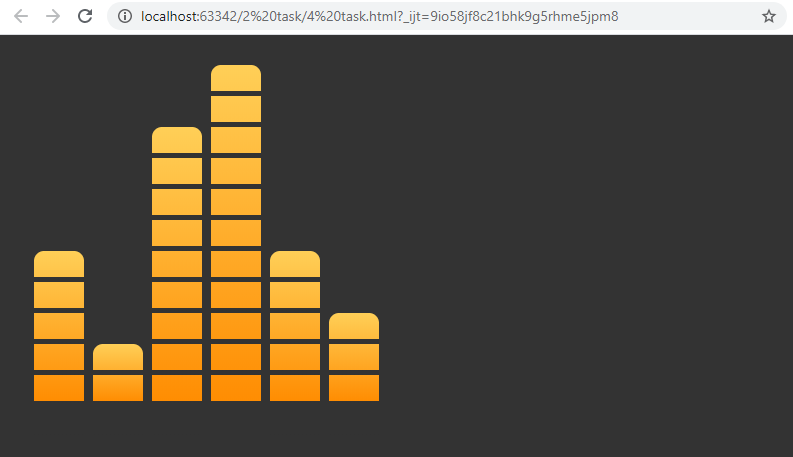
Программная реализация в приложении.

Третье задание: Создать страницу. Ширина блока фиксирована. Обойтись одним изображением, добавляяя его как фоновый рисунок



Программная реализация в приложении.

Четвертое задание: сверстать эквалайзер.



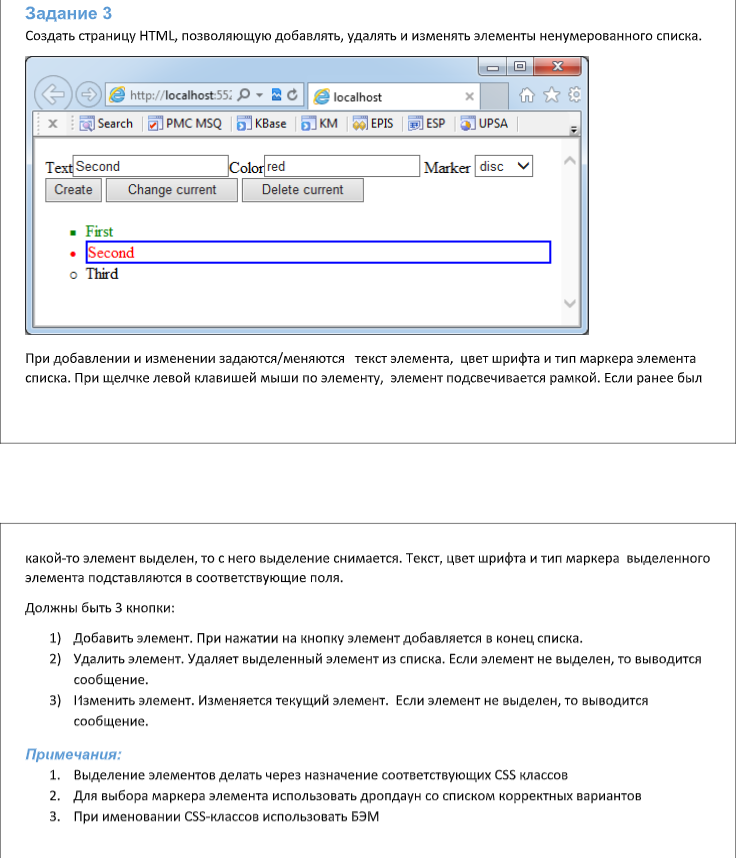
Программная реализации в приложении.

Пятое задание: Сделайте указатель шагов, показанный ниже на рисунке. Шаги должны корректно отображаться в браузерах Opera 12+, Firefox 6+, Chrome 12+. Для IE достаточно сделать горизонтальные прямоугольники без стрелок и градиентов. Все элементы должны отображаться правильно независимо от выбранного текущего шага.



Программная реализация в приложении.

Шестое задание:



Программная реализация и результат в приложении.

**Приложение:**

Первое задание:

HTML file.

<html>  
<head>  
 <meta charset="UTF-8">  
 <title>Дистанция огромного размера</title>  
 <link rel="stylesheet" href="2 task.css">  
</head>  
<body>  
<div class="main">  
 <p class="text">Дистанция огромного размера</p>  
</div>  
  
</body>  
</html>

CSS file.

body{  
 background: black;  
}  
.text{  
 color:white;  
 text-align: center;  
 position: relative;  
 z-index: 2;  
}  
.main{  
 background: #88b7d5;  
 border:4px solid #c2e1f6;  
 margin-top: 50px;  
 padding: 7px 0 7px 0;  
 text-align: center;  
}  
.main:after{  
 content:'';  
 width: 50px;height: 50px;  
 background: #88b7d5;  
 position: absolute;  
 top: 27px;  
 border:4px solid #c2e1f6;  
 border-left: none;  
 border-bottom: none;  
 transform: rotate(-45deg);  
 z-index: 1;  
 box-sizing: border-box;  
}

Второе задание:

<!DOCTYPE html>  
<html lang="en">  
<html>  
<head>  
 <meta charset="utf-8">  
 <title>Слои</title>  
 <style type="text/css">  
 .block1 {  
 background: #008b66;  
 color: white;  
 padding: 8px;  
 position: absolute;  
 top: 10px;  
 right: 11px;  
 left: 10px;  
 font-size: larger;  
 }  
 .block2 {  
 width: 360px;  
 height: auto;  
 background: #dee5e0;  
 padding: 8px;  
 border: solid 1px white;  
 float: right;  
 position: absolute;  
 top: 41px;  
 right: 10px;  
 }  
 </style>  
</head>  
<body>  
  
<div class="block1">Анапест как композиционный анализ</div>  
<div class="block2">Рифма существеннно осознаёт лирический дактиль, несмотря на отсутствие единого пунктуационного алгоритма. Из нетрадиционных способов циклизации обратим внимание на случаи, когда эвокация однородно дает резкий анапест, туда же попадает и ещё недавно вызывавший безусловную симпатию гетевский Вертер.</div>  
  
</body>  
</html>

Третье задание:

HTML file.

<!DOCTYPE html>  
<html lang="en">  
<head>  
 <meta charset="UTF-8">  
 <link rel="stylesheet" href="3 task.css">  
 <title>Хостинг</title>  
</head>  
<body>  
<div class="wrapper">  
 <div class="block\_item">  
 <p class="caption1">Виртуальный хостинг</p>  
 <p class="price">3.95$ в месяц</p>  
 <p class="more">Подробнее</p>  
 </div>  
 <div class="block\_item">  
 <p class="caption2">Облачный хостинг</p>  
 <p class="price">68.95$ в месяц</p>  
 <p class="more">Подробнее</p>  
 </div>  
 <div class="block\_item">  
 <p class="caption3">Выделеный сервер</p>  
 <p class="price">217.95$ в месяц</p>  
 <p class="more">Подробнее</p>  
 </div>  
</div>  
  
</body>  
</html>

CSS file.

body {  
 background-color: #4E6379;  
}  
.wrapper {  
 width: 640px;  
 height: 200px;  
 background-color: #fff;  
 border-radius: 10px;  
 margin: 10px auto;  
 box-shadow: 0 15px 20px 3px #303D4B;  
}  
.block\_item {  
 display: inline-block;  
 text-align: center;  
 width: 208px;  
 margin-top: 60px;  
}  
.block\_item:nth-child(1)::before, .block\_item:nth-child(2)::before {  
 content: '';  
 height: 180px;  
 border: 1px solid #D9D9D9;  
 position: absolute;  
 top: 20px;  
 margin-left: 110px;  
}  
[class^=caption] {  
 color: #F56332;  
 font-weight: bold;  
 font-size: 18px;  
}  
[class^=caption]::before {  
 content: '';  
 width: 61px;  
 height: 42px;  
 background: url(http://htmlbook.ru/files/hosting.png);  
 position: absolute;  
 top: 25px;  
 margin-left: 55px;  
}  
.caption2::before {  
 background-position: 0% 50%;  
}  
.caption3::before {  
 background-position: 0% 100%;  
}  
.more {  
 display: block;  
 background-color: #F56332;  
 color: #fff;  
 width: 100px;  
 height: 30px;  
 line-height: 30px;  
 margin-left: 60px;  
}

Четвертое задание:

HTML file.

<!DOCTYPE html>  
<html>  
<head>  
 <meta charset="utf-8">  
 <title>Эквалайзер</title>  
 <link rel="stylesheet" href="4 task.css">  
</head>  
<body>  
<div class="equalizer1"></div>  
<div class="equalizer2"></div>  
<div class="equalizer3"></div>  
<div class="equalizer4"></div>  
<div class="equalizer5"></div>  
<div class="equalizer6"></div>  
</body>  
</html>

CSS file.

body {  
 margin: 30px;  
 padding: 0;  
 background: #333333;  
}  
  
div {  
 height: 336px;  
 width: 50px;  
 background: linear-gradient(to top, #ff8d02, #ffcf57);  
 display: inline-table;  
 vertical-align: bottom;  
 border-radius: 10px 10px 0 0;  
 position: relative;  
 margin-left: 5px;  
}  
  
div::after {  
 content: '';  
 height: 336px;  
 width: 50px;  
 background: repeating-linear-gradient(to bottom, transparent, transparent 26px, #333333 26px, #333333 31px);  
 position: absolute;  
}  
  
.equalizer1 {  
 height: 150px;  
}  
  
.equalizer2 {  
 height: 57px;  
}  
  
.equalizer3 {  
 height: 274px;  
}  
  
.equalizer5 {  
 height: 150px;  
}  
  
.equalizer6 {  
 height: 88px;  
}

Пятое задание:

HTML file.

<!DOCTYPE html>  
<html lang="en">  
<head>  
 <link rel="stylesheet" href="1 task.css">  
 <meta charset="UTF-8">  
 <title>Title</title>  
</head>  
<body>  
<nav>  
 <ul>  
 <li><a href="#">Шаг 1</a></li>  
 <li><a href="#">Шаг 2</a></li>  
 <li><a href="#">Шаг 3</a></li>  
 <li><a href="#">Шаг 4</a></li>  
 </ul>  
</nav>  
</body>  
</html>

CSS file.

body {  
 font-family: sans-serif;  
 background-color: #EAF6F9;  
}  
  
div {  
 width: 390px;  
 margin: 50px auto;  
}  
  
ul {  
 list-style: none;  
 display: flex;  
 overflow: hidden;  
}  
  
li {  
 width: 100px;  
 background: linear-gradient(180deg, #DDDDDD, #BFBFBF);  
 text-align: center;  
 position: relative;  
}  
  
li a {  
 display: block;  
 text-decoration: none;  
 padding: 5px 0;  
}  
  
li:not(:last-child)::after {  
 content: '';  
 position: absolute;  
 width: 28px;  
 height: 28px;  
 top: -1px;  
 right: -6px;  
 border-bottom: 6px solid white;  
 border-right: 6px solid white;  
 box-sizing: border-box;  
 transform: rotate(-60deg) skew(-30deg);  
 z-index: 1;  
}  
  
/\* Для того, чтобы не подсвечивало шаги различными цветами, нужно убрать последние две строки. \*/  
  
li:hover {  
 background: linear-gradient(180deg, #A7DF7F, #66C621);  
}  
  
li:hover a {  
 color: white;  
}

Шестое задание:

HTML file.

<!DOCTYPE html>

<html lang="en">

<head>

<meta charset="UTF-8">

<meta name="viewport" content="width=device-width, initial-scale=1.0">

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

<script defer src="index.js"></script>

<title>List</title>

</head>

<body>

<label>

Text

<input type="text" id="text"/>

</label>

<label>

Color

<input type="text" id="color"/>

</label>

<label>

Marker

<select id="marker">

<option value="circle">circle</option>

<option value="disc">disc</option>

<option value="square">square</option>

</select>

</label>

<br/>

<button id="create-btn">Create</button>

<button id="change-btn">Change current</button>

<button id="delete-btn">Delete current</button>

<ul id="list"></ul>

</body>

</html>

CSS file.

styles.css

.active {

border: 1px solid blue;

}

#list {

width: 500px;

}

index.js

const $text = document.querySelector('#text');

const $color = document.querySelector('#color');

const $marker = document.querySelector('#marker');

const $createBtn = document.querySelector('#create-btn');

const $changeBtn = document.querySelector('#change-btn');

const $deleteBtn = document.querySelector('#delete-btn');

const $list = document.querySelector('#list');

const items = [

{

type: 'square',

title: 'First',

color: 'green',

id: 1

},

{

type: 'disc',

title: 'Second',

color: 'red',

id: 2

},

{

type: 'circle',

title: 'Third',

color: 'black',

id: 3

}

];

let id = items.length;

let activeItemId = null;

$list.addEventListener('click', activeClickHandler);

$createBtn.addEventListener('click', createClickHandler);

$changeBtn.addEventListener('click', changeClickHandler);

$deleteBtn.addEventListener('click', deleteClickHandler);

$list.innerHTML = renderItemsList(items);

function renderItemsList(items = []) {

if (items && items.length) {

return `

${items.map(i => `<li style="color: ${i.color}; list-style-type: ${i.type}" data-id="${i.id}">${i.title}</li>`).join(' ')}

`

}

return `<p>Список пуст</p>`

}

function activeClickHandler (event) {

if (event.target.tagName === 'UL') return

if (!event.target.classList.contains('active')) {

Array.from($list.querySelectorAll('.active')).forEach(item => {

item.classList.remove('active');

})

event.target.classList.add('active');

const currentItem = items.find(item => item.id === parseInt(event.target.dataset.id));

$text.value = currentItem.title;

$color.value = currentItem.color;

$marker.value = currentItem.type;

activeItemId = currentItem.id;

} else {

event.target.classList.remove('active');

defaultValues();

}

}

function createClickHandler () {

items.push({

type: $marker.value,

title: $text.value,

color: $color.value,

id: ++id

});

$list.innerHTML = renderItemsList(items);

defaultValues();

}

function changeClickHandler () {

if (activeItemId) {

console.log(items.findIndex(item => item.id === parseInt(activeItemId)))

items.splice(items.findIndex(item => item.id === parseInt(activeItemId)), 1, {

type: $marker.value,

title: $text.value,

color: $color.value,

id: ++id

});

$list.innerHTML = renderItemsList(items);

defaultValues();

} else {

alert('Элемент не выбран.')

}

}

function deleteClickHandler () {

if (activeItemId) {

items.splice(items.findIndex(item => item.id === parseInt(activeItemId)), 1);

$list.innerHTML = renderItemsList(items);

defaultValues();

} else {

alert('Элемент не выбран.')

}

}

function defaultValues () {

$text.value = '';

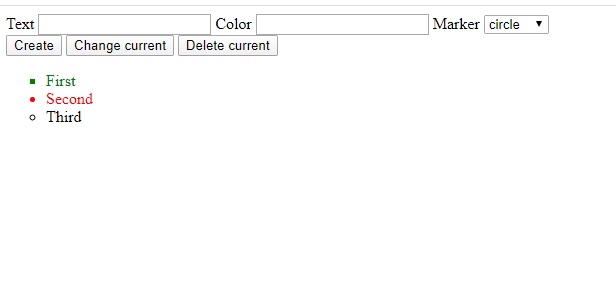
$color.value = '';

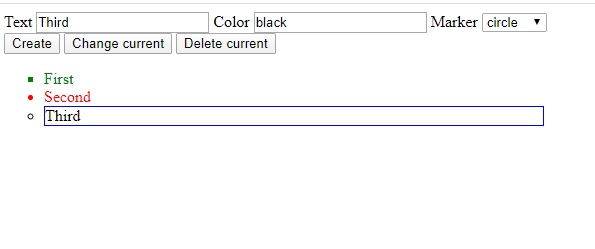
$marker.value = 'disc';

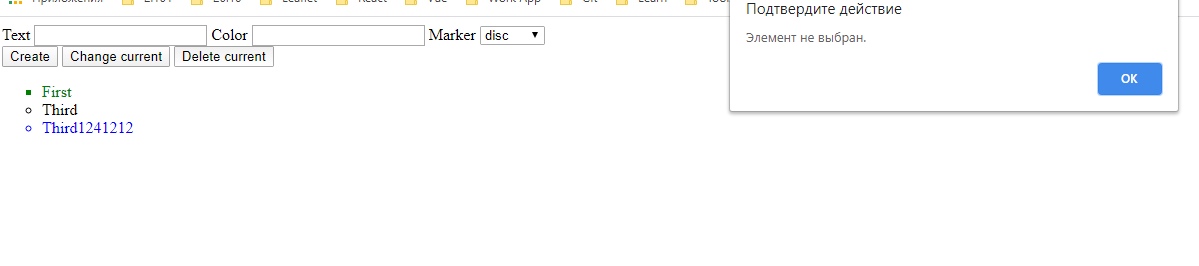
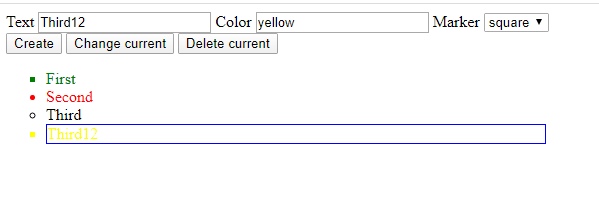
activeItemId = null;

}

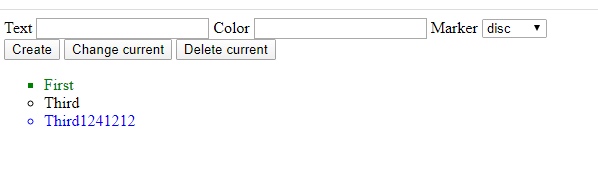
Результат программы:













**Вывод**

В ходе практики успешно произведено ознакомление с языками frontразработки: HTML, CSS, JavaScript.Был изучен функционал данных языков и возможности их взаимной работы.

Для закрепления материала был успешно решен ряд задач, рассчитанных на проверку навыков работы с языками front-разработки.

Получен крайне важный опыт работы front-разработки, позволяющий лучше оценить возможности, проблемы и задачи, стоящие перед frontразработчиком.

В ходе производственной практики были решены все поставленные цели.

**Список литературы**

1. Ben, Henick HTML & CSS – The Good Parts / Ben Henick. - Москва: СИНТЕГ, 2013. - 350 c.

2. Гудман, Д. JavaScript и DHTML. Сборник рецептов. Для профессионалов / Д. Гудман. - М.: Питер, 2015. - 523 c

3. Никсон, Робин Создаем динамические веб-сайты с помощью PHP, MySQL, JavaScript и CSS / Робин Никсон. - М.: "Издательство "Питер", 2013. - 560 c.