



МИНОБРНАУКИ РОССИИ

Федеральное государственное бюджетное образовательное учреждение высшего
образования
«МИРЭА – Российский технологический университет»

РТУ МИРЭА

Институт информационных технологий (ИТ)

Кафедра инструментального и прикладного программного обеспечения (ИиППО)

ОТЧЕТ ПО ПРАКТИЧЕСКИМ РАБОТАМ

по дисциплине

«Разработка клиентских частей интернет-ресурсов»

Выполнил студент группы ИКБО-20-19

Московка А.А.

Принял
Ассистент

Меркулов Е.С.

Практические работы выполнены

« » 2020 г.

(подпись студента)

«Зачтено»

« » 2020 г.

(подпись руководителя)

Москва 2020

Оглавление

Практическая работа №8 «JavaScript-анимация»	3
Задания на тему «JavaScript-анимации»	3
Задание 2: «Функция «draw»	4
Задание 3: «Анимированный прыгающий мяч»	7
Вывод:	9

Практическая работа №8 «JavaScript-анимация»

Задания на тему «JavaScript-анимации»

Выбрать изображение любого вида транспорта и реализовать анимацию (при нажатии на данное изображение) через последовательность кадров, каждый из которых немного меняет HTML/CSS-свойства. Например, изменение `style.left` от 0px до 100px – двигает элемент. И если это будет выполняться с помощью `setInterval`, изменяя на 2px с небольшими интервалами времени, например, 50 раз в секунду, тогда изменения будут выглядеть плавными. Принцип такой же, как в кино: 24 кадров в секунду достаточно, чтобы создать эффект плавности.

Для реализации необходимо:

1. Запомнить время начала анимации;
2. Добавить функцию для расчета времени, которое прошло с начала анимации;
3. Закончить анимацию через 2 секунды;
4. В то время как `timePassed` идёт от 0 до 2000 `left` изменяет значение от 0px до 400px.

Листинг 1 – HTML-файл с встроенным JS кодом (Рисунок 1.1):

```
1 <!DOCTYPE html>
2 <html lang="en">
3 <head>
4   <meta charset="UTF-8">
5   <title>Практическая работа №8 - задание 1</title>
6   <style>
7     #kchaw {
8       position: relative;
9       cursor: pointer;
10    }
11  </style>
12 </head>
13 <body>
14   
16   <script>
17     kchaw.onclick = function() {
18       let start = Date.now();
19       let timer = setInterval(function() {
20         let timePassed = Date.now() - start;
21         kchaw.style.left = timePassed / 6 + 'px';
22         if (timePassed > 2000) clearInterval(timer);
23       }, 10);
24     }
25   </script>
26
27 </body>
28 </html>
```

Рисунок 1.1 – Код программы с встроенным JS-кодом

Результат выполнения кода на странице (Рисунок 1.2):



Рисунок 1.2 – Результат работы кода на странице до активации анимации



Рисунок 1.3 – Результат работы кода на странице после активации анимации

Задание 2: «Функция «draw»

Создать анимацию набирающегося текста в элементе формы `textarea` при нажатии на кнопку. Добавить в `<head>` следующую функцию для анимации (дописав функции для `textArea` и расчёта времени `bounce`):

```
function animate(options) {  
  
    var start = performance.now();
```

```
requestAnimationFrame(function animate(time) {  
    // timeFraction от 0 до 1  
    var timeFraction = (time - start) / options.duration;  
    if (timeFraction > 1) timeFraction = 1;  
  
    // текущее состояние анимации  
    var progress = options.timing(timeFraction)  
  
    options.draw(progress);  
  
    if (timeFraction < 1) {  
        requestAnimationFrame(animate);  
    }  
  
});  
}
```

Листинг 2 – HTML-документ с встроенным JS сценарием (Рисунок 2.1):

```

1  <!DOCTYPE html>
2  <html lang="en">
3  <head>
4    <meta charset="UTF-8">
5    <title>Практическая работа №8 - задание 2</title>
6    <script>
7      function animate(options) {
8        var start = performance.now();
9        requestAnimationFrame(function animate(time) {
10          // timeFraction от 0 до 1
11          var timeFraction = (time - start) / options.duration;
12          if (timeFraction > 1) timeFraction = 1;
13          // текущее состояние анимации
14          var progress = options.timing(timeFraction)
15          options.draw(progress);
16          if (timeFraction < 1) {
17            requestAnimationFrame(animate);
18          }
19        });
20      }
21    </script>
22  </head>
23  <body>
24    <textarea id="textExample" rows="9" cols="60">
25      Стонет новый грузовик,
26      Он к работе не привык.
27      Он к работе не приучен,
28      Он заплакан и замучен.
29      Не грусти грузовичок,
30      Ты не первый новичок.
31      По большим дорогам ездить
32      Интересно, если вместе.
33    </textarea>
34    <br>
35    <button onclick="animateText(textExample)">Набрать текст</button>
36    <script>
37      function animateText(textArea) {
38        let text = textArea.value;
39        let to = text.length,
40            from = 0;
41        animate({
42          duration: 5000,
43          timing: bounce,
44          draw: function(progress) {
45            let result = (to - from) * progress + from;
46            textArea.value = text.substr(0, Math.ceil(result))
47          }
48        });
49      }
50      function bounce(timeFraction) {
51        for (let a = 0, b = 1, result; 1; a += b, b /= 2) {
52          if (timeFraction >= (7 - 4 * a) / 11) {
53            return -Math.pow((11 - 6 * a - 11 * timeFraction) / 4, 2) + Math.pow(b, 2)
54          }
55        }
56      }
57    </script>
58  </body>
59  </html>

```

Рисунок 2.1 – Код программы с набором текста с помощью draw

Результат выполнения кода на странице (Рисунок 2.2-2.3):

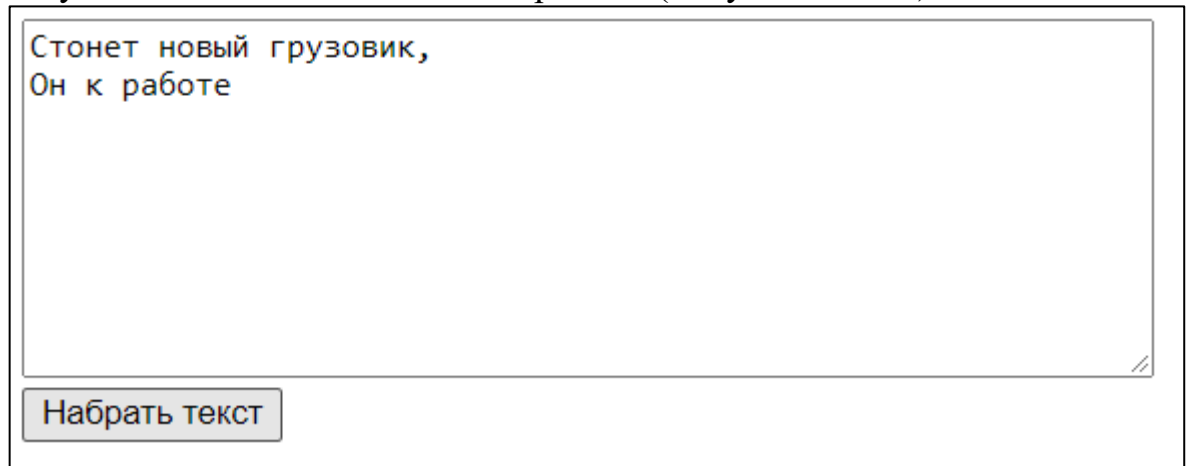


Рисунок 2.2 – Результат выполнения кода во время набора текста

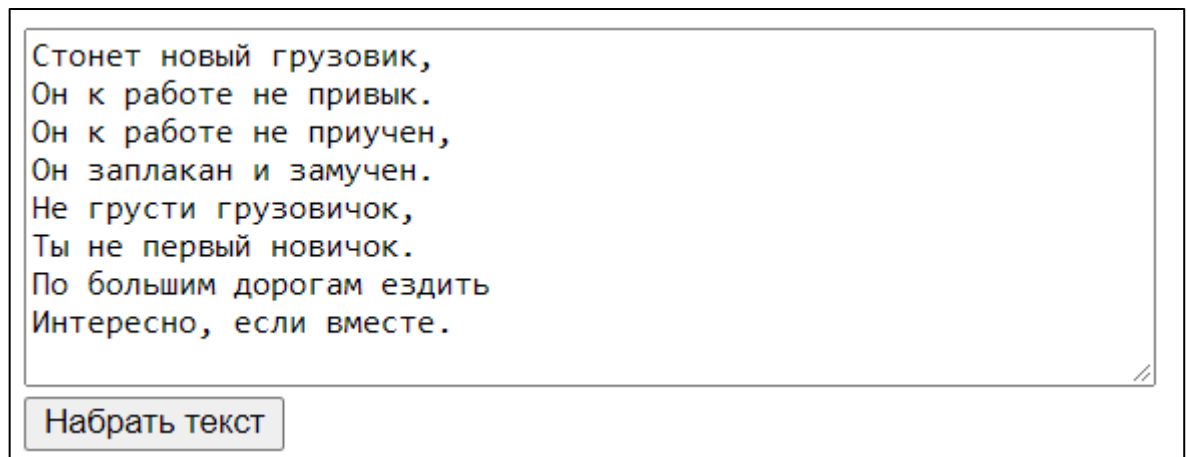


Рисунок 2.3 – Результат выполнения кода после набора текста

Задание 3: «Анимированный прыгающий мяч»

Создать анимацию с прыгающим мячом. При нажатии на мяч он должен падать вниз до созданной горизонтальной линии и несколько раз подскакивать от этой поверхности. Чтобы получить эффект «скачущего» мяча, можно использовать функцию расчёта времени bounce в режиме easeOut.

Листинг 3 – HTML-документ (Рисунок 3.1):

```
1 <!DOCTYPE html>
2 <html lang="en">
3 <head>
4   <meta charset="UTF-8">
5   <title>Практическая работа №8 - задание 3</title>
6   <style>
7     #field {height: 200px;
8     border-bottom: 3px #43c83e groove;
9     position: relative;}
10    #ball {position: absolute;
11    cursor: pointer;}
12  </style>
13  <script>
14    function animate(options) {
15      var start = performance.now();
16      requestAnimationFrame(function animate(time) {
17        // timeFraction on 0 to 1
18        var timeFraction = (time - start) / options.duration;
19        if (timeFraction > 1) timeFraction = 1;
20        // текущее состояние анимации
21        var progress = options.timing(timeFraction)
22        options.draw(progress);
23        if (timeFraction < 1) {
24          requestAnimationFrame(animate);
25        }
26      });
27    }
28  </script>
29 </head>
30 <body>
31   <div id="field">
32     
33   </div>
34   <script>
35     function makeEaseOut(timing) {
36       return function(timeFraction) {
37         return 1 - timing(1 - timeFraction);
38       }
39     }
40     function bounce(timeFraction) {
41       for (let a = 0, b = 1, result; 1; a += b, b /= 2) {
42         if (timeFraction >= (7 - 4 * a) / 11) {
43           return -Math.pow((11 - 6 * a - 11 * timeFraction) / 4, 2) + Math.pow(b, 2)
44         }
45       }
46     }
47     ball.onclick = function() {
48       let to = field.clientHeight - ball.clientHeight;
49       animate({
50         duration: 2000,
51         timing: makeEaseOut(bounce),
52         draw(progress) {
53           ball.style.top = to * progress + 'px'
54         }
55       });
56     };
57   </script>
58 </body>
59 </html>
```

Рисунок 3.1 – Код программы

Результат выполнения кода на странице (Рисунок 3.2-3.3):

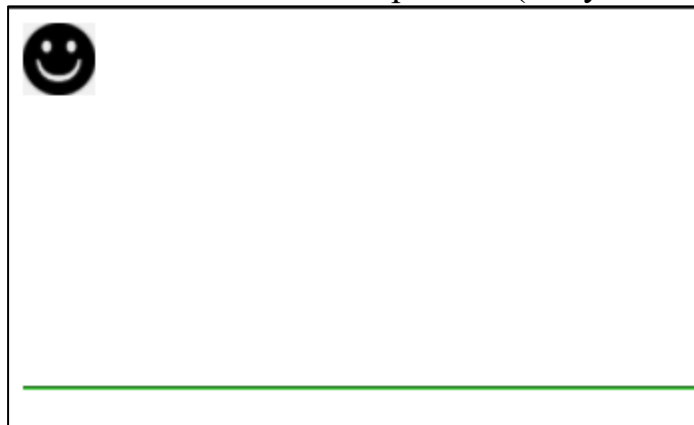


Рисунок 3.2 – Результат выполнения кода на странице до проигрывания анимации



Рисунок 3.3 – Результат выполнения кода на странице после проигрывания анимации

Вывод:

В результате данной практической работы были получены навыки работы с JavaScript, использованием `setInterval`, функции `draw()` и основ анимации.