

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](#_Toc59463169)

[Задания на тему «JavaScript-анимации» 3](#_Toc59463170)

[Задание 2: «Функция «draw» 4](#_Toc59463171)

[Задание 3: «Анимированный прыгающий мяч» 7](#_Toc59463172)

[Вывод: 9](#_Toc59463173)

Практическая работа №8 «JavaScript-анимация»

Задания на тему «JavaScript-анимации»

Выбрать изображение любого вида транспорта и реализоватьанимацию (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):



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):

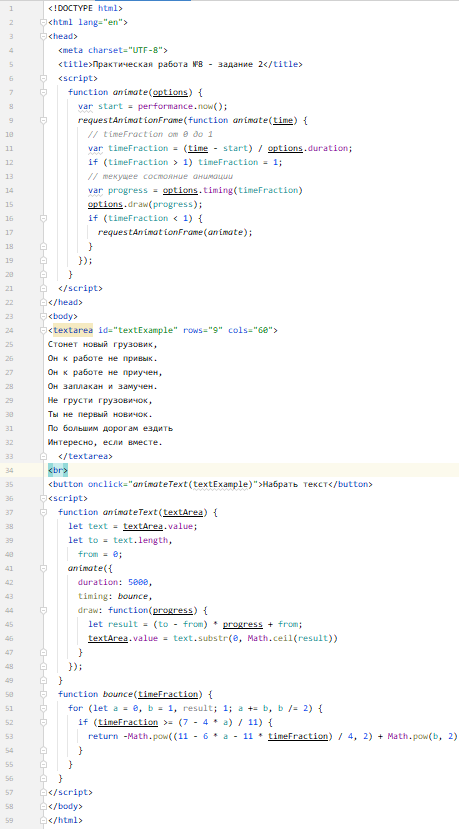


Рисунок 2.1 – Код программы с набором текста с помощью draw

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

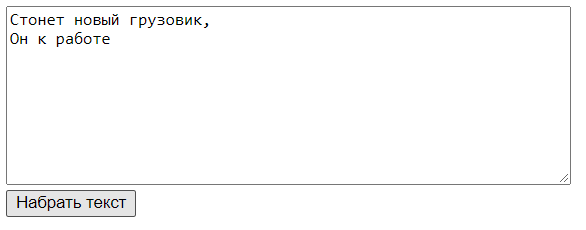


Рисунок 2.2 – Результат выполнения кода во время набора текста

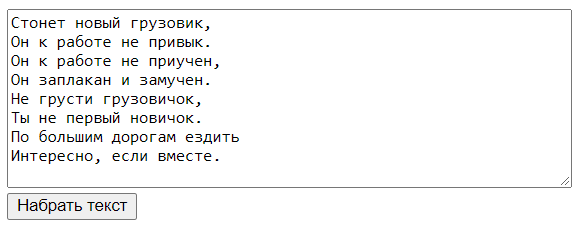


Рисунок 2.3 – Результат выполнения кода после набора текста

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):

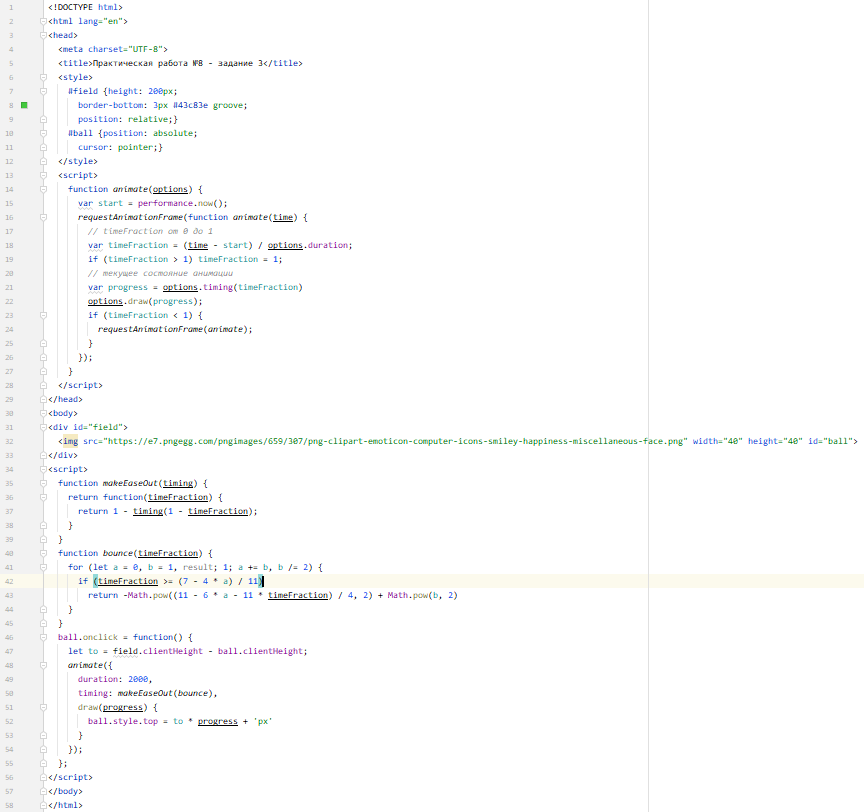


Рисунок 3.1 – Код программы

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.