

Демонстрация ЛР:  
Гапанюк. Ю.Е.

**Отчет по лабораторной работе № 8 по курсу  
РИП**

**"Javascript"**

ИСПОЛНИТЕЛЬ:

студент группы **ИУ5-54**

**Наседкин И.А.**

## Задание и порядок выполнения

Разработать приложение для построения графиков тригонометрических функций на

языке Javascript с HTML интерфейсом.

Пример интерфейса:

Ход работы:

1. Ознакомиться с теоретической частью

2. Создайте новый проект PyCharm

тип проекта: Pure Python

(мы не будем использовать Python в этой работе, просто это позволяет создать абсолютно пустой проект без зависимостей)

3. Добавьте в проект 2 файла:

a. index.html

b. index.js

4. Сверстайте страницу со следующими элементами:

a. два поля ввода для области определения аргумента (<input>)

b. поле для ввода функции (<input>)

c. кнопка “Построить график” (<button>)

d. поле вывода графика (<div>)

5. При помощи css укажите размеры блока графика, отличные от нуля

6. Присвойте каждому полю уникальный class (например, from, to, fun, output и т.д.)

7. Убедитесь, что ваша страница отображается в браузере нормально

8. Подключите jQuery, flot и ваш скрипт в index.html, используя теги <script>

```
<script src= "https://code.jquery.com/jquery-2.2.4.min.js" ></script>
```

```
<script src= "https://cdnjs.cloudflare.com/ajax/libs/flot/0.8.3/jquery.flot.js" ></script>
```

9. Переходим к разработке скрипта

10. Дождитесь загрузки страницы

```
$(function() {
```

```
// ...
```

```
})
```

11. Найдите все элементы управления на вашей странице

```
var $from = $('from');
```

12. Подпишитесь на событие нажатия кнопки

```
$button.click(onClick);
```

13. Отмените действие по-умолчанию (отправку формы)

```
e.preventDefault()
```

14. Получите значения из полей ввода

```
$from.val()
```

15. Не забудьте преобразовать числовые значения из строк в числа

```
parseFloat, parseInt
```

16. Создайте массив пар значений

```
const points = [[x1, y1], ..., [xn, yn]];
```

17. Для того, чтобы получить значение функции, заданной в виде строки, используйте функцию eval()

```
const x = 0.1 ;
```

```
const fun = 'Math.sin(x)' ;
```

```
const y = eval (fun);
```

18. Постройте график по точкам

```
$. plot ( $ output, [ points ], { });
```

19. Проверьте правильность работы приложения, в случае проблем, воспользуйтесь отладчиком Chrome DevTools

20. Проверьте построение графиков функций:

a. Math.sin(x)

b. Math.random()

c. Math.exp(x)

21. Выведите название построенной функции в легенду:

<http://www.flotcharts.org/plot/examples/basic-options/index.html>

22. Дополнительное задание:

сделайте анимацию графика функции как на осциллографе

для этого по таймеру setInterval() / clearInterval() перестраивайте график функции, прибавляя к  $x$  изменяющийся коэффициент  $dx$

## Текст программы

### index.html

```
<!DOCTYPE html>
<html lang="en">
<head>
  <meta charset="UTF-8">
  <script src="https://code.jquery.com/jquery-2.2.4.min.js"></script>
  <script
src="https://cdnjs.cloudflare.com/ajax/libs/flot/0.8.3/jquery.flot.js"></script>
  <script src='index.js'></script>
  <title>LAB_8</title>
  <style type="text/css">
    button {
      background-color: white; width: 170px;
      height: 40px;font-size: 16px;margin-top: 20px;
    }
    input{
      font-size: 14px;
    }
  </style>
</head>
<body>
  <div style="float: left; width: 600px; text-align: center; font-size: 20px;
margin-top: 50px;">
    <label> От</label>
    <input class="from" size = '46'>
    <br>
    <label > До </label>
    <input class="to" size = '46'>
    <br>
    <label style="width: 40px">Функция</label>
    <input list ="functions" class="fun" size = '37'>
    <datalist id="functions">
    <option>Math.sin (x) </option>
```

```

<option>Math.random()</option>
<option>Math.exp(x)</option>
</datalist>
<br>

<button class="plot" >Построить график</button>
</div>
<div class="graph" style="width: 500px;height: 500px; float: left; margin-
right: 180px;margin-top: 50px;">
</div>
</body>
</html>

```

## index.js

```

$(function() {

    var $from=$("#.from");
    var $to=$("#.to");
    var $fun=$("#.fun");
    var $graph=$("#.graph");
    var $plot=$("#.plot");

    $plot.click(function (e) {

        e.preventDefault();
        var x = parseFloat($from.val());
        const to = parseFloat($to.val());
        const fun = $fun.val();

        var mas_points = [x, eval(fun)];

        for ( ; x <= to; x +=0.01 )
            mas_points.push([x, eval(fun)]);

        $.plot($graph, [{label: fun, data: mas_points}], {});
    });
});

```

## Результаты работы программы

