

**Отчет по лабораторной работе №8
по курсу «Разработка Интернет-приложений»**

Вариант № <21>

ИСПОЛНИТЕЛЬ:
студент группы ИУ5-52
Ромичева Е.В.
16.12.2016 г.

Содержание

Задание и порядок выполнения	3
Исходный код	4
Результат.....	5

Задание и порядок выполнения

Разработать приложение для построения графиков тригонометрических функций на языке Javascript с HTML интерфейсом.

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.

Исходный код

Index.html

```
<!DOCTYPE html>
<html lang="en">
<head>
  <link rel="stylesheet" type="text/css" href="style.css">
  <meta charset="UTF-8">
  <title>График</title>
</head>
<body>
  <form>
    <div id="right">
      <p>From: <input class='from'></p>
      <p> To: <input class='to'></p>
      <p> Fun: <input class="fun"></p>
    </div>
    <p><button class='plot' type=button>Plot!</button></p>
  </form>
  <div class="graph"></div>
  <script src="https://code.jquery.com/jquery-3.1.1.min.js"></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>
  <script src='index.js'></script>
</body>
</html>
```

Index.js

```
var graf;
$('.plot').click(function (e) {
  e.preventDefault()
  clearInterval(graf);
  console.log(graf);
  var options = {
    series: {
      color: "#0F0",
      lines: { show: true, lineWidth: 3, fill: false, fillColor: "#00F" },
```

```

    points: { show: false, fill: false }
  }
};
var x = parseFloat($('.from').val());
const x1 = x;
var i = x;
const x2 = parseFloat($('.to').val());
const fun = ($.fun).val();
var points = [x, eval(fun)];
if (i < x2)
{
  graf=setInterval(function () {
    $.plot($('.graph'), [{label: fun, data: points}], options);
    x = x + (x2 - x1) / 100;
    // console.log(points);
    if (points.length > 100) {
      points.splice(1, 1)
    }
    points.push([x, eval(fun)])
  }, 100);
}
else
{
  clearInterval(graf);
}
});

```

Результат

From:

To:

Fun:

