**Московский государственный технический**

**университет им. Н.Э. Баумана.**

Факультет «Информатика и системы управления»

Кафедра ИУ5. Курс «Разработка интернет приложений»

Отчет по лабораторной работе №5

**«JavaScript»**

Выполнил: Проверил:

студент группы ИУ5-52 доцент каф. ИУ5

Ефимов К.А. Гапанюк Ю.Е.

Подпись и дата: Подпись и дата:

Москва, 2017 г.



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:
   1. два поля ввода для области определения аргумента (<input>)
   2. поле для ввода функции (<input>)
   3. кнопка “Построить график” (<button>)
   4. 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.)

1. Убедитесь, что ваша страница отображается в браузере нормально
2. 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>
3. Переходим к разработке скрипта
4. Дождитесь загрузки страницы  $(function() { // ...  })
5. Найдите все элементы управления на вашей странице  var $from = $(‘.from);
6. Подпишитесь на событие нажатия кнопки  $button.click(onClick);
7. Отмените действие по-умолчанию (отправку формы)  e.preventDefault()
8. Получите значения из полей ввода  $from.val()
9. Не забудьте преобразовать числовые значения из строк в числа  parseFloat, parseInt
10. Создайте массив пар значений  const points = [[x1, y1], ..., [xn, yn]];
11. 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 ] , {});

1. Проверьте правильность работы приложения, в случае проблем, воспользуйтесь отладчиком Chrome DevTools
2. Проверьте построение графиков функций: a. Math.sin(x)  b. Math.random()  c. Math.exp(x)
3. Выведите название построенной функции в легенду:  http://www.flotcharts.org/flot/examples/basic-options/index.html
4. Дополнительное задание: сделайте анимацию графика функции как на осциллографе для этого по таймеру setInterval() / clearInterval() перестраивайте график функции, прибавляя к ***x*** изменяющийся коэффициент ***dx***

Теория

Javascript

JavaScript обычно используется как встраиваемый язык для программного доступа к объектам приложений. Наиболее широкое применение находит в браузерах как язык сценариев для придания интерактивности веб-страницам. Основные архитектурные черты: динамическая типизация, слабая типизация, автоматическое управление памятью, прототипное программирование, функции как объекты первого класса.

Javascript обычно выполняется на стороне клиента, в браузере. С помощью JS можно взаимодействовать с DOM (дерево HTML элементов) текущей открытой страницы, управлять навигацией, отправлять данные на сервер (AJAX) и получать обратно.

Однако браузер является лишь окружением для Javascript, предоставляя ему своё API. Javascript также может использоваться и как серверный язык, аналогично, например, Python (Django, Tornado) или C# (ASP .Net). Также возможно собрать standalone десктопное приложение аналогично C++ (Qt) или C# (WPF). Однако это уже скорее экзотика.

Отличное руководство по JS можно найти здесь: https://learn.javascript.ru

**Код**

Index.js

$(function(){

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

var $to = $('.to');

var $fun = $('.fun');

var $button = $('button');

var $plot1 = $('.plot1');

var $plot2 = $('.plot2');

$button.click(function(e){

e.preventDefault();

$from = parseFloat($from.val());

$to = parseFloat($to.val());

$fun = $fun.val()

var $dx = 0.05

var $values = []

for(var x = $from; x <= $to; x += $dx){

$values.push([x, eval($fun)])

}

$.plot($plot1, [{label: $fun, data: $values}],{})

setInterval(function(){

$from += $dx;

$to += $dx;

$values = []

for(var x = $from; x <= $to; x += $dx){

$values.push([x, eval($fun)])

}

$.plot($plot2, [{label: $fun, data: $values}], {})

}, 50);

});

})

Index.htmp

<b>Область определения:</b><br>

<input type="text" class="from" size="10"><input type="text" class="to" size="10">

<br><br>

<b>Функция:</b><br>

<input type="text" class="fun">

<br><br>

<button>Построить график</button>

<br><br>

<b>График:</b>

<div class="plot1" style="width: 400px; height:200px;">

</div>

<div class="plot2" style="width: 400px; height:200px;"></div>

<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 type="text/javascript" src="index.js"></script>

