**Московский государственный технический университет им. Н.Э.Баумана**

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

Отчет по лабораторной работе №8

«Javascript»

|  |  |  |
| --- | --- | --- |
| Выполнил: |  | Преподаватель: |
| Катков Александр, ИУ5-52 |  | Гапанюк Ю.Е. |
| \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ |  | \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ |
|  |  |  |

2017 г.

1. **Задание лабораторной работы.**

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/flot/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.***

1. **Листинг**

**Index.html**

<!DOCTYPE **html**>  
<**html lang="en"**>  
<**head**>  
 <**meta charset="UTF-8"**>  
 <**link rel="stylesheet" type="text/css" href="index.css"**>  
 <**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**>  
 <**title**>Построение графиков</**title**>  
</**head**>  
<**body**>  
 <**h1**>Построение графиков</**h1**>  
 <**div class="form"**>  
 <**label**>  
 From: <**input type="text" class="from" placeholder="From"**>  
 </**label**>  
 <**label**>  
 To: <**input type="text" class="to" placeholder="To"**>  
 </**label**>  
 <**label**>  
 Function: <**input type="text" class="func" placeholder="Function"**>  
 </**label**>  
 <**button type="button" class="btn"**>Построить график</**button**>  
 </**div**>  
 <**div class="output"**>  
 </**div**>  
 <**button type="button" class="stop"**>Стоп</**button**>  
</**body**>  
</**html**>

**Index.js**

$(**function**() {  
 **var** $from = $(**'.from'**);  
 **var** $to = $(**'.to'**);  
 **var** $func = $(**'.func'**);  
 **var** $button = $(**'.btn'**);  
 **var** $output = $(**'.output'**);  
 **var** $stop = $(**'.stop'**);  
  
 $button.click(**function** (e) {  
 e.preventDefault();  
 $stop.show();  
 **var** begin = *parseFloat*($from.val());  
 **var** end = *parseFloat*($to.val());  
 **var** func = $func.val();  
 **var** point = [];  
 **const** dx = 0.2;  
  
 **var** dynamic = *setInterval*(**function** () {  
 **for** (**var** x = begin; x<=end; x+=0.1) {  
 point.push([x,*eval*(func)]);  
 }  
  
 **var** points = [{**data**: point, **label**: func}];  
 $.plot($output, points, {});  
  
 begin += dx;  
 end += dx;  
 point = [];  
 }, 100);  
  
 $stop.click(**function** () {  
 *clearInterval*(dynamic);  
 })  
 });  
});

**Index.css**

**body** {  
 **padding**: 0;  
 **margin-left**: 20**px**;  
}  
  
.**form** {  
 **width**: 240**px**;  
 **height**: 150**px**;  
}  
  
.**form label** {  
 **float**: **right**;  
 **padding-bottom**: 10**px**;  
}  
  
.**form button** {  
 **float**: **right**;  
}  
  
.**output** {  
 **width**: 500**px**;  
 **height**: 400**px**;  
}  
  
.**stop** {  
 **display**: **none**;  
}

1. **Результат:**

