

Утверждаю: _____

Согласовано: _____

"__" ____ 2016 г.

"__" ____ 2016 г.

«Javascript»

Отчет по лабораторной работе №8

(вид документа)

писчая бумага формата А4

(вид носителя)

(количество листов)

Исполнитель:

студент группы РТ5-51

_____ Попков В.Е.

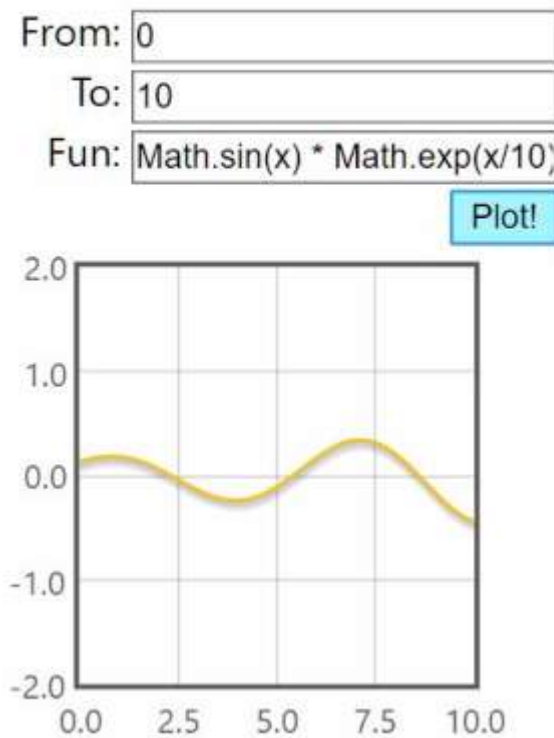
"__" ____ 2016 г.

Москва – 2016

1.Задание и порядок выполнения

Разработать приложение для построения графиков тригонометрических функций на язык 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: а. два поля ввода для области определения аргумента б. поле для ввода функции с. кнопка “Построить график” () d. поле вывода графика
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
9. Переходим к разработке скрипта
10. Дождитесь загрузки страницы $\$(function() \{ // \dots \})$

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: а. `Math.sin(x)` б. `Math.random()` с. `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`

HTML-файл:

```
index.html x  index.js x
1  <!DOCTYPE html>
2  <html lang="en">
3  <head>
4      <meta charset="UTF-8">
5      <title>Title</title>
6      <script src="https://code.jquery.com/jquery-2.2.4.min.js"></script>
7      <script src="https://cdnjs.cloudflare.com/ajax/libs/flot/0.8.3/jquery.flot.js"></script>
8      <script src="/untitled/index.js"></script>
9  </head>
10 <body>
11     <p> From:<input type="text" class="FROM" ></p>
12     <p> To:<input type="text" class="TO"></p>
13     <p> Fun:<input type="text" class="FUN"></p>
14     <p> <input type="button" value="Строить" id="INPUT"></p>
15     <div id="placeholder" style="..."></div>
16
17 </body>
18 </html>
```

JavaScript-файл

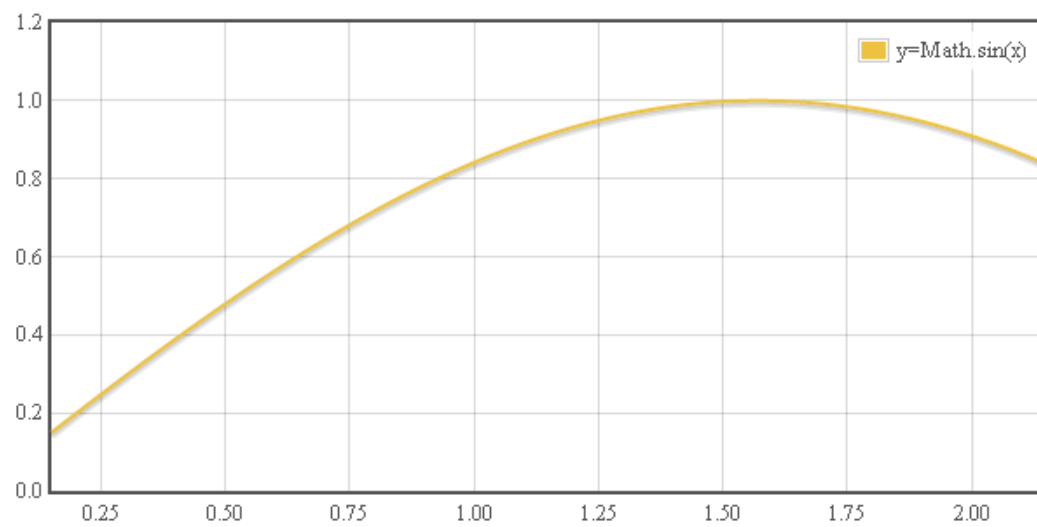
```
index.html x index.js x
1 var fromX, toX, point=[], x, timerId
2 $(function() {
3   $("#INPUT").click(function (e) {
4     clearInterval(timerId)
5     point=[]
6     var from = $('#FROM').val();
7     var to = $('#TO').val();
8     var fun = $('#FUN').val();
9
10    step = 0.01
11    alert(fun);
12    for (x = parseFloat(from); x<parseFloat(to); x+=step)
13    {
14      point.push([x, eval(fun)]);
15    }
16    $.plot($("#placeholder"), [{ label: 'y='+fun, data: point }], {});
17    timerId = setInterval(() => {move(from, fun, step); }, 100);
18  }
19 }
20 );
21 function move(from, fun, step) {
22   point.splice(0, 1);
23   point.push([x, eval(fun)]);
24   $.plot($("#placeholder"), [{ label: 'y='+fun, data: point }], {});
25   x = parseFloat(x) + parseFloat(step);
26 }
27 })
28
```

Работа приложения

From:

To:

Fun:

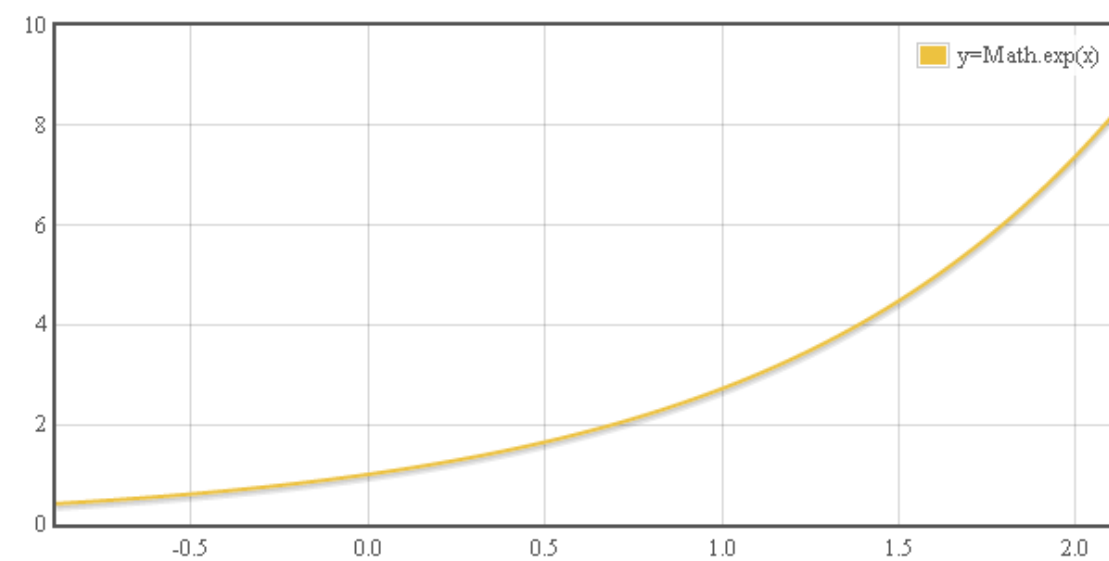


Пример работы до. условия:

From:

To:

Fun:



From: -1

To: 2

Fun: $\text{Math.exp}(x)$

Строить

