Московский государственный технический университет им. Н.Э. Баумана

Факультет «Информатика и системы управления»

Кафедра «Автоматизированные системы обработки информации и управления»



**Отчет**

#### По лабораторной работе №7

#### По курсу «Разработка интернет приложений»

«Работа с DOM с использованием JavaScript»

Вариант 19

**ИСПОЛНИТЕЛЬ:**

Попов Илья

Группа ИУ5-51Б

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

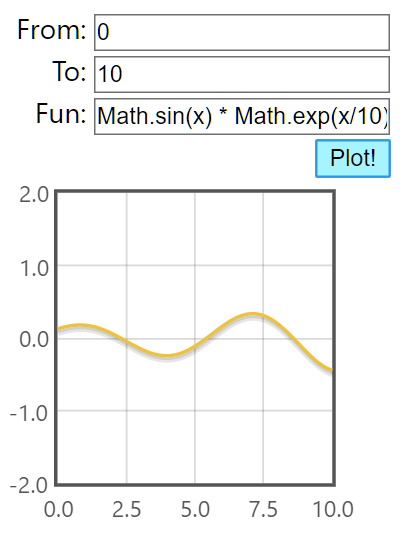
"\_\_"\_\_\_\_\_\_\_\_\_\_\_2020 г.

Москва 2020

Задание

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

Пример интерфейса:

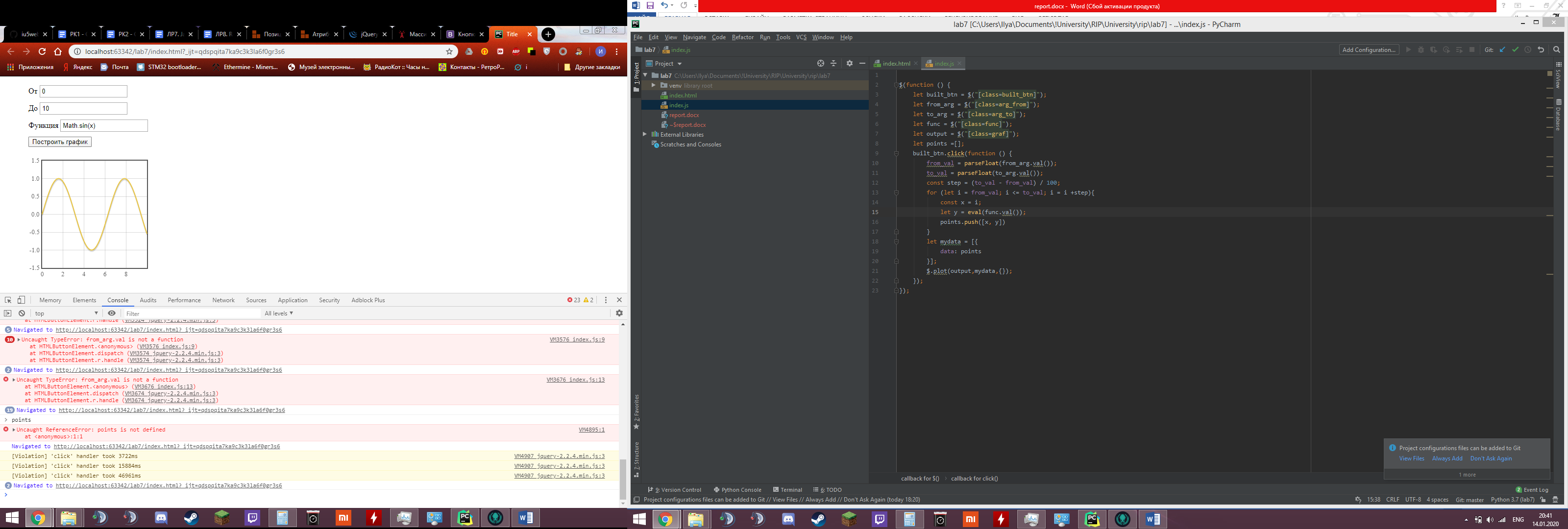


Ход работы:

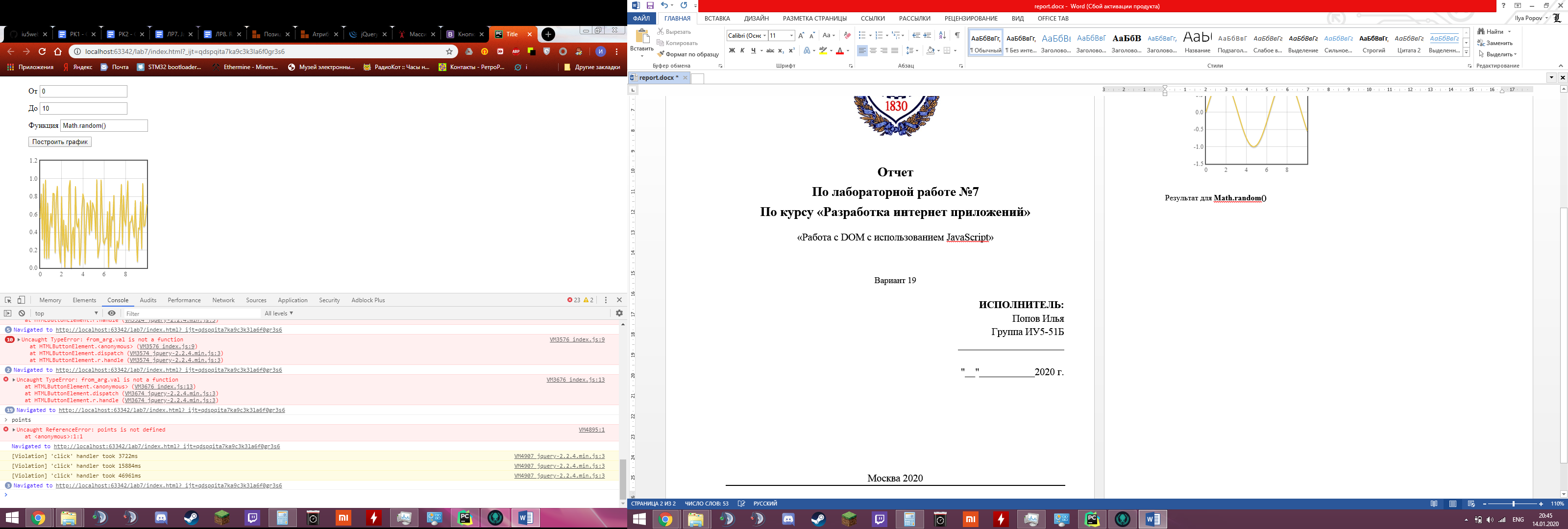
1. Ознакомиться с теоретической частью
2. Создайте новый проект PyCharm  
   тип проекта: Pure Python  
   (мы не будем использовать Python в этой работе, просто это позволяет создать абсолютно пустой проект без зависимостей)
3. Добавьте в проект 2 файла:
4. index.html
5. index.js
6. Сверстайте страницу 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>)
7. При помощи css укажите размеры блока графика, отличные от нуля
8. Присвойте каждому полю уникальный class (например, from, to, fun, output и т.д.)
9. Убедитесь, что ваша страница отображается в браузере нормально
10. 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>
11. Переходим к разработке скрипта
12. Дождитесь загрузки страницы  
    $(function() {  
        // …  
    })
13. Найдите все элементы управления на вашей странице  
    var $from = $(‘.from);
14. Подпишитесь на событие нажатия кнопки  
    $button.click(onClick);
15. Отмените действие по-умолчанию (отправку формы)  
    e.preventDefault()
16. Получите значения из полей ввода  
    $from.val()
17. Не забудьте преобразовать числовые значения из строк в числа  
    parseFloat, parseInt
18. Создайте массив пар значений  
    const points = [[x1, y1], …, [xn, yn]];
19. 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);
20. Постройте график по точкам  
    $.plot($output, [ points ], {});
21. Проверьте правильность работы приложения, в случае проблем, воспользуйтесь отладчиком Chrome DevTools
22. Проверьте построение графиков функций:
    1. Math.sin(x)
    2. Math.random()
    3. Math.exp(x)
23. Выведите название построенной функции в легенду:

Выполнение

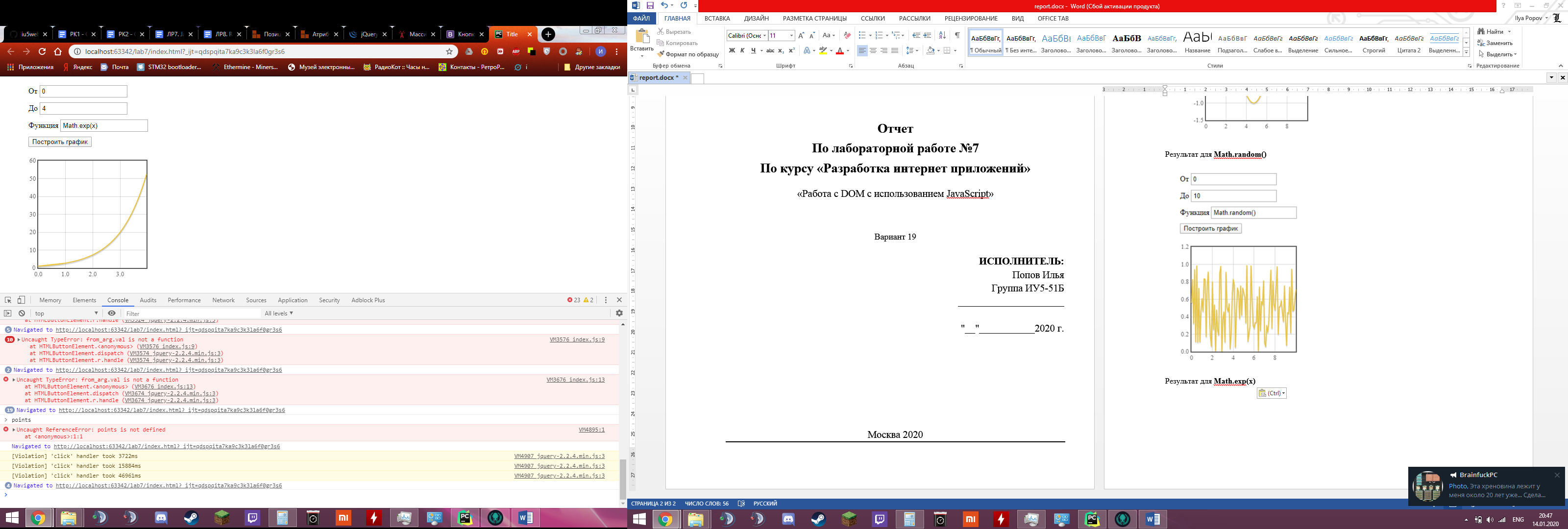
Результат для **Math.sin(x)**



Результат для **Math.random()**



Результат для **Math.exp(x)**



Исходный код

**index.html**

<!DOCTYPE html>  
<html lang="en">  
<head>  
 <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>  
 <meta charset="UTF-8">  
 <title>Title</title>  
 <style>  
 .main{  
 padding-left: 50px;  
 top: 50px;  
 }  
 input{  
 margin-top: 10px;  
 padding: 3px;  
 }  
 .graf{  
 margin-top: 20px;  
 height: 250px;  
 width: 250px;  
 }  
 button{  
 margin-top: 10px;  
 }  
 </style>  
 <script src="index.js"></script>  
</head>  
<body>  
<div class="main">  
 <div class="input\_block">  
 <label for="id\_arg\_from">От</label>  
 <input id="id\_arg\_from" type="number" class="arg\_from"><br>  
 <label for="id\_arg\_to">До</label>  
 <input id="id\_arg\_to" type="number" class="arg\_to"><br>  
 <label for="id\_func">Функция</label>  
 <input id="id\_func" class="func"><br>  
 <button class="built\_btn">Построить график</button>  
 </div>  
 <div class="graf">  
  
 </div>  
</div>  
</body>  
</html>

**index.js**

$(function () {  
 let built\_btn = $("[class=built\_btn]");  
 let from\_arg = $("[class=arg\_from]");  
 let to\_arg = $("[class=arg\_to]");  
 let func = $("[class=func]");  
 let output = $("[class=graf]");  
 let points =[];  
 built\_btn.click(function () {  
 from\_val = parseFloat(from\_arg.val());  
 to\_val = parseFloat(to\_arg.val());  
 func\_val = func.val();  
 const step = (to\_val - from\_val) / 100;  
 for (let i = from\_val; i <= to\_val; i = i +step){  
 const x = i;  
 let y = eval(func\_val);  
 points.push([x, y])  
 }  
 let mydata = [{  
 data: points  
 }];  
 $.plot(output,mydata,{});  
 });  
});