

Министерство науки и высшего образования Российской Федерации Федеральное государственное бюджетное образовательное учреждение высшего образования

«Московский государственный технический университет имени Н.Э. Баумана

(национальный исследовательский университет)» (МГТУ им. Н.Э. Баумана)

ФАКУЛЬТЕТ ИНФОРМАТИКА И СИСТЕМЫ УПРАВЛЕНИЯ

КАФЕДРА КОМПЬЮТЕРНЫЕ СИСТЕМЫ И СЕТИ (ИУ6)

(Группа)

Преподаватель

НАПРАВЛЕНИЕ ПОДГОТОВКИ – 09.03.01 Информатика и вычислительная техника

ОТЧЕТ

по лабораторной работе № 9

Название:	СОЗДАНИЕ ПРОСТЕЙ	и́ших веб-приложени́	ІЙ RUBY ON
RAILS. AJAX			
Дисциплина	Языки интернет-програ	иммирования	
Студент	ИУ6-32Б		Джафаров Э.Э.

(И.О. Фамилия)

Петрова Я.С. (И.О. Фамилия)

(Подпись, дата)

(Подпись, дата)

Цель работы — углубление теоретических сведений о принципах работы асинхронного веб-интерфейса и получение практических навыков создания веб-приложения с использованием средств
Ruby on Rails и технологии AJAX.

Задание:

- 1. При помощи Javascript модифицировать код ЛР 8 таким образом, чтобы для отображения результатов вычисления браузер не выполнял полную перезагрузку страницы.
- 2. Сформировать тесты для проверки работы программы при помощи Katalon Recorder / Selenium Webdriver.

Решение:

1. Контроллер calc_controller.rb

```
class CalcController < ApplicationController
       def input
       render layout: false
       end
       def show
         @right numbers = []
         @a = params[:power].to i
8
         (10**(@a - 1)...(10**@a)).step(1) do |number|
         @right numbers.push(number) if find(number) == number
10
         end
11
12
        rjson
13
       end
14
15
       def rjson
         respond to do |format|
17
           format.html
18
           format.json do
             render json:
             { length: @right numbers.length, value: @right numbers }
20
21
           end
22
         end
       end
23
24
25
       def find(number)
26
         summa = 0
         @a.times do
27
           tri = number % 10
29
           number /= 10
           summa += tri**@a
30
31
         end
32
         summa
33
       end
     end
34
```

2. input.html.erb

3. routes.rb

```
Rails.application.routes.draw do
root 'calc#input'
get 'calc/show'
# Define your application routes per the DSL in https://guides.re

# Defines the root path route ("/")
# root "articles#index"

end

9
```

4. index.html

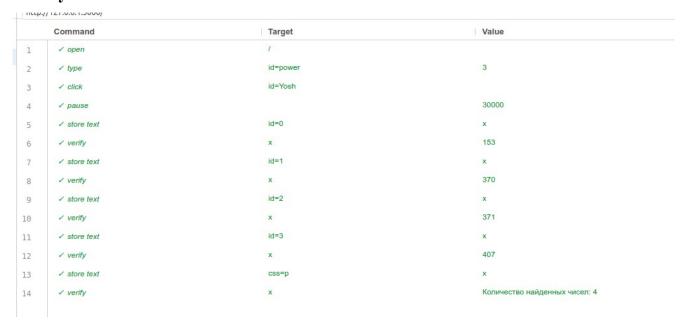
```
<!DOCTYPE html>
     <html>
     <head>
         <meta http-equiv="Content-Type" content="text/html; charset=utf-8" />
         <title>Четыре</title>
         <script>
              input url = location.protocol + "//" + location.host + "/calc/show.json"
              function send_query() {{
10
                  var frm = document.forms.input form;
                  param str = "?" +
11
12
                      "power=" + frm.power.value + "&commit=Yosh"
                  if (param_str == "") return false;
var my_JSON_object = {};
13
14
15
                  var http_request = new XMLHttpRequest();
16
                  http_request.open("GET", input_url + param_str, true);
17
                  http request.onreadystatechange = function () {
                      var done = 4, ok = 200;
18
                      if (http_request.readyState == done &&
19
20
                          http_request.status == ok) {
21
                          my JSON object =
22
                               JSON.parse(http request.responseText);
                           show result(my JSON object);
23
24
25
                  http request.send(null);
26
27
                  return false;
28
```

```
function show_result(data) {
                                  var result = document.getElementById("result");
                                 var table = document.createElement("table");
var tr = document.createElement('tr');
                                  table.setAttribute("border", 1)
result.innerHTML = "<hr/>Result is:</br>";
                                  var th1 = document.createElement('th'); // это ячейка заголовка
var text1 = document.createTextNode("Порядковый номер"); // текст заголовка
35
                                  var th2 = document.createElement('th');
var text2 = document.createTextNode("Число");
                                 Var text2 = document.createrextNode( число );
thl.appendChild(text1); // прикрепляем текст к заголовку
th2.appendChild(text2);
tr.appendChild(th1); // прикрепляем заголовок к ряду
tr.appendChild(th2);
table.appendChild(tr); // прикрепляем ряд к таблице
for (let i = 0; i < data.length; i++) { // это длина моего массива, которое я передевал через контролле
                                         var tr = document.createElement('tr'); // создаем ряд
var tdl = document.createElement('td'); // ячейку
var td2 = document.createElement('td'); // вторую ячейку
                                          var text1 = document.createTextNode(i); // παμωρα τεκττ
var text2 = document.createTextNode(data.value[i]);
                                         Var text2 = document.createrextwode(data.vatde[1]),
tdl.appendChild(text1); // прикрепляем текст к ячейке
td2.appendChild(text2);
td2.setAttribute("id", i)
tdl.setAttribute("align", "center") // устанавливаем атрибут, чтобы текст был в центре
td2.setAttribute("align", "center")
51
52
54
55
                                          tr.appendChild(td1); // прикрепляем ячейку к ряду tr.appendChild(td2);
                                          table.appendChild(tr); // прикрепляем ряд к таблице
                                  result.appendChild(table)
                                  result.innerHTML += "Количество найденных чисел: " + data.length + "<hr/>"
                  </script>
          </head>
```

5. Тест Selenium

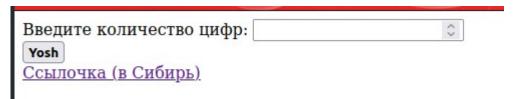
	Command	Target	Value
1	open	I	
2	type	id=power	3
3	click	id=Yosh	
4	pause		30000
5	store text	id=0	x
6	verify	х	153
7	store text	id=1	x
8	verify	х	370
9	store text	id=2	х
10	verify	х	371
11	store text	id=3	x
12	verify	х	407
13	store text	css=p	х
14	verify	х	Количество найденных чисел: 4

6. Результат теста:

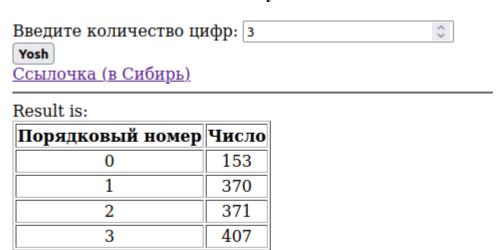


7. Результат работы:

7.1 Страница ввода:



7.2 После нажатия на кнопку:



Количество найденных чисел: 4

8. Проверка Rubocop:

```
k/IPL-Laboratory9$ rubocop -A app/controllers/calc_controller.rb
Inspecting 1 file
.
1 file inspected, no offenses detected
```

Вывод:

Я познакомился с АЈАХ и научился работать с асинхронными запросами