|  |  |
| --- | --- |
| **Gerb-BMSTU_01** | **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)**

НАПРАВЛЕНИЕ ПОДГОТОВКИ **09.03.03 «Прикладная информатика»**

**Отчет**

|  |  |
| --- | --- |
| **по лабораторной работе №** | 9 |

**Название:**

Создание простейших веб-приложений Ruby on Rails. AJAX

**Дисциплина:** Языки Интернет-программирования

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Студент | ИУ6-34Б |  | 20.11.2022 | С. А. Рахманов |
|  | (Группа) |  | (Подпись, дата) | (И.О. Фамилия) |
|  |  |  |  |  |
| Преподаватель |  |  |  | В. М. Малахов |
|  |  |  | (Подпись, дата) | (И.О. Фамилия) |

Москва, 2022

**Цель работы:** углубление теоретических сведений о принци-  
пах работы асинхронного веб-интерфейса и получение практиче-  
ских навыков создания веб-приложения с использованием средств  
Ruby on Rails и технологии AJAX.

**Задание:**

Все консольные приложения Ruby следует реализовывать в виде трех

отдельных файлов:

1. основная программа;

2. программа для взаимодействия с пользователем через консоль;

3. программа для автоматического тестирования на основе MiniTest::Unit

или RSpec. Везде, где это возможно, данные для проверки должны

формироваться автоматически по правилам, указанным в задании.

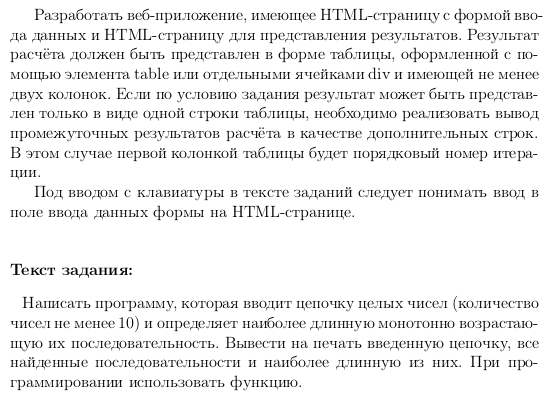
При реализации программ везде, где это возможно, следует избегать

использования циклов for, do, while. Вместо них используйте методы из

примеси Enumerable.

Все тексты программ должны быть проверены на соответствие стилю

программирования Ruby при помощи rubocop.ru или reek.



**Тексты измененных файлов**

*seq\_controller.rb*

def correct(arr)

found = []

tmp\_found = []

arr.take(arr.size - 1).each\_index do |i|

tmp\_found << arr[i]

unless arr[i] < arr [i + 1]

found << tmp\_found

tmp\_found = []

end

end

tmp\_found << arr[-1]

found << tmp\_found

sizes = found.map(&:size)

@max\_found = found[sizes.index(sizes.max)].join(' ')

@found = found.map { |x| x.join(' ') }.join(' | ')

end

# Top-level class

class SeqController < ApplicationController

def input; end

def view

@str = !params[:str] || params[:str] == '' ? 'Error' : params[:str]

@max\_found = @str

@found = @str

arr = @str&.split&.map(&:to\_i)

arr && !arr&.empty? ? correct(arr) : nil

respond\_to do |format|

format.html

format.js

end

end

end

*input.html.erb*

<h1>Seq#input</h1>

<p>Find me in app/views/seq/input.html.erb</p>

<%= form\_with url: '/seq/view.js', method: :get, local: false do |form| %>

<%= form.label("Input string:") %> <br/>

<%= form.text\_field(:str, value: "0 1 5 5 5 10 20 25 -5 1 5 25 125 625")%> <br/><br/>

<%= form.submit("Correct and print")%> <br/>

<% end %>

<div id="result"></div>

*view.js.erb*

document.getElementById('result').innerHTML = `

<h3>Результаты:</h3>

<%== '<p>НЕТ ПОСЛЕДОВАТЕЛЬНОСТЕЙ</p>' if !@found %>

<%== alert if @str&.match(/[\s\d]/).nil? %>

<p id="str">str: <%= @str %></p>

<p id="max found">max found: <%= @max\_found %></p>

<p id="found">found: <%= @found %></p>

`;

*routes.rb*

Rails.application.routes.draw do

get 'seq/input'

get 'seq/view'

root 'seq#input'

# Define your application routes per the DSL in https://guides.rubyonrails.org/routing.html

# Defines the root path route ("/")

# root "articles#index"

end

*seq\_controller\_test.rb*

# frozen\_string\_literal: true

require 'test\_helper'

class SeqControllerTest < ActionDispatch::IntegrationTest

test 'should get input' do

get seq\_input\_url

assert\_response :success

end

test 'should get view' do

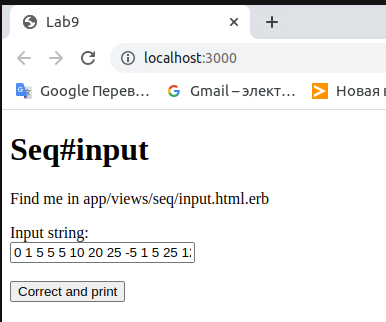
get seq\_view\_url

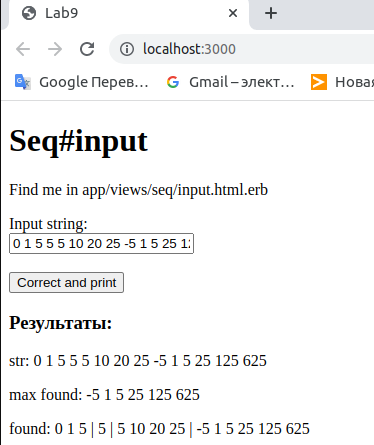
assert\_response :success

end

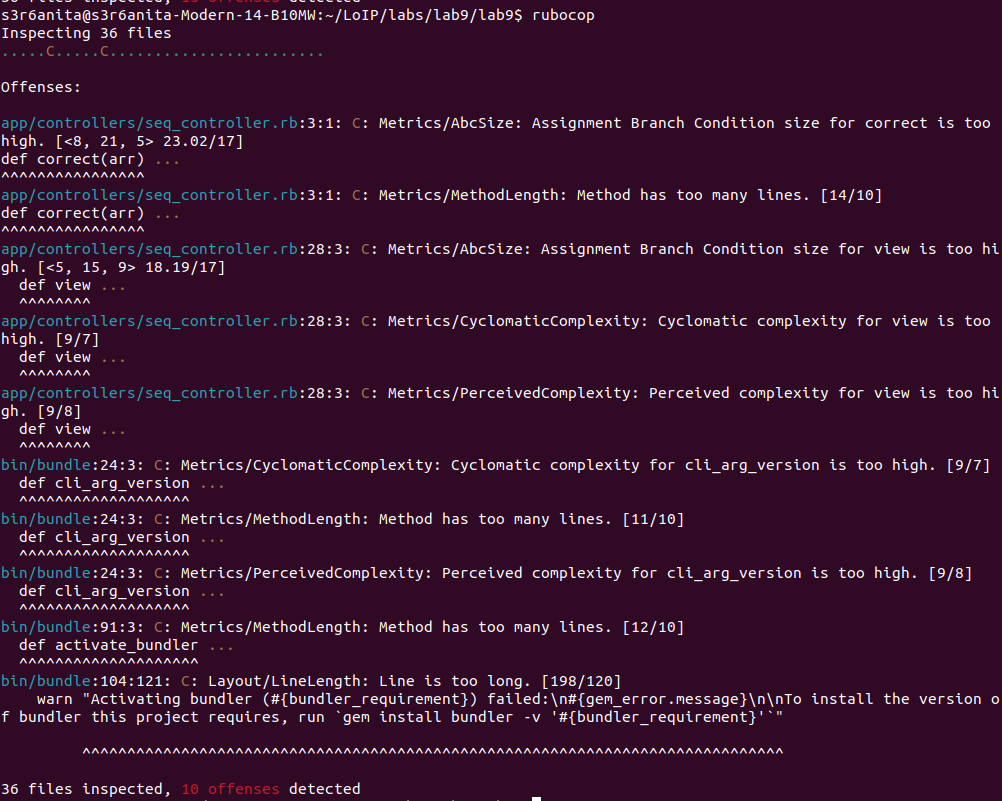
end

**Результаты выполнения:**

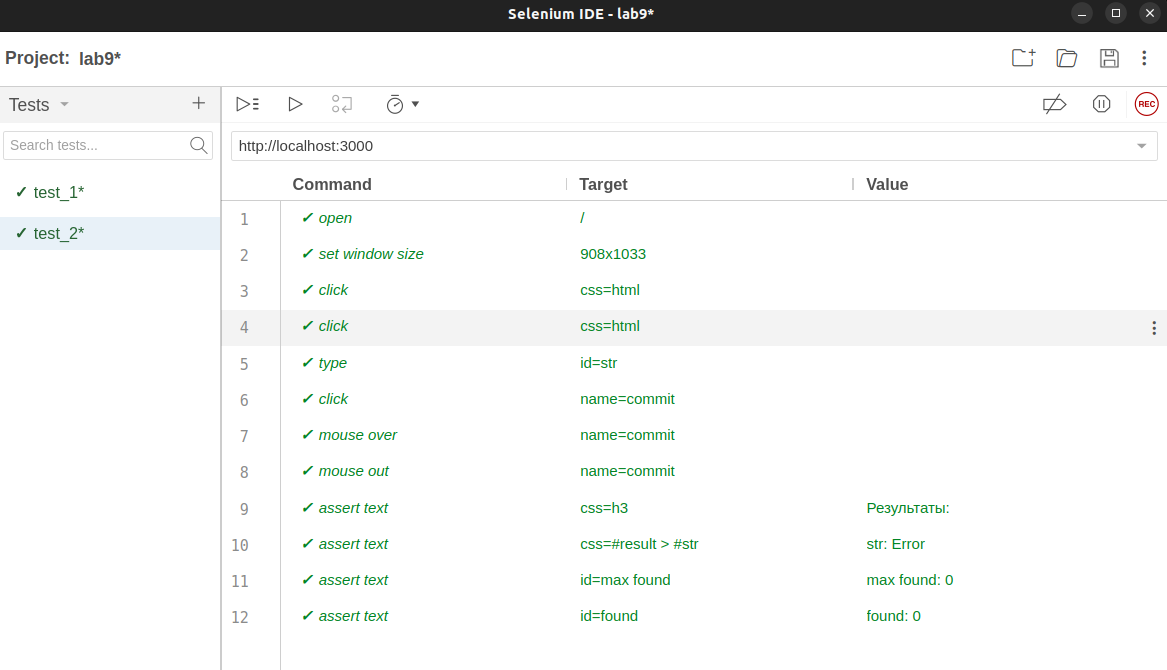


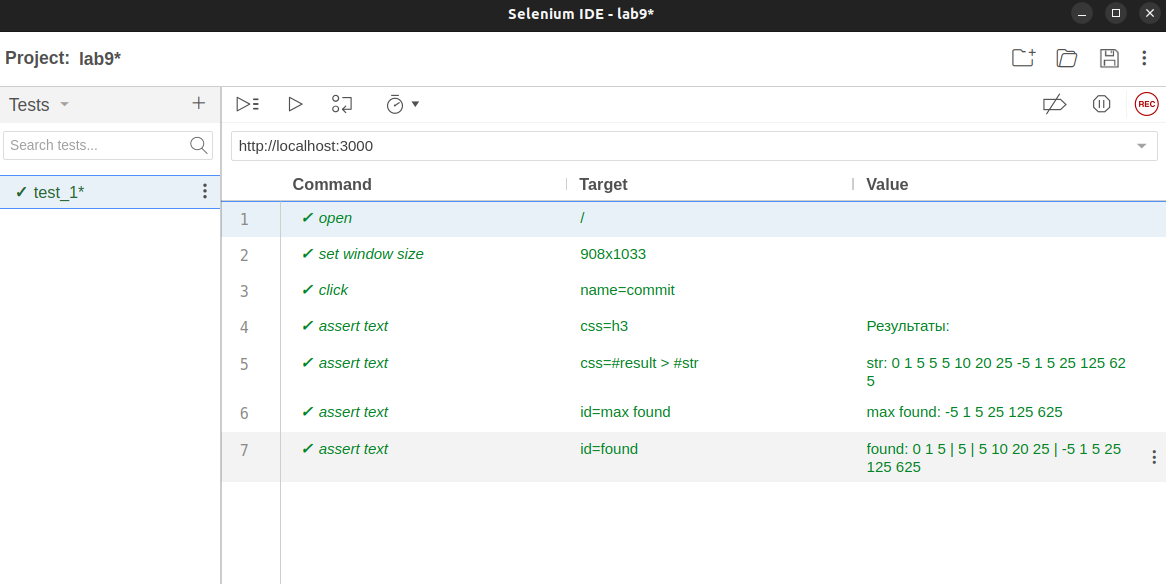
****

**Результаты проверки анализатором rubocop:**

****

**Тестирование с помощью Selenium IDE:**





**Вывод:** в результате выполнения лабораторной работы были получены практические навыки создания асинхронного веб-интерфейса, интеграционного тестирования с помощью среды Sleneium IDE.