

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)

О т ч е т

по лабораторной работе № 8

Название лабораторной работы: Простейшее RoR приложение

Дисциплина: Языки Интернет-программирования

Студент гр. ИУ6-31Б _____
(Подпись, дата)

Корнеев К.А.
(И.О. Фамилия)

Преподаватель _____
(Подпись, дата)

Малахов Д.В.
(И.О. Фамилия)

Москва, 2022

11 Вариант

Цель работы: научиться создавать простейшие Rails приложения, работать с контроллерами и представлениями

Задания:

ЛР 8. Ruby on Rails

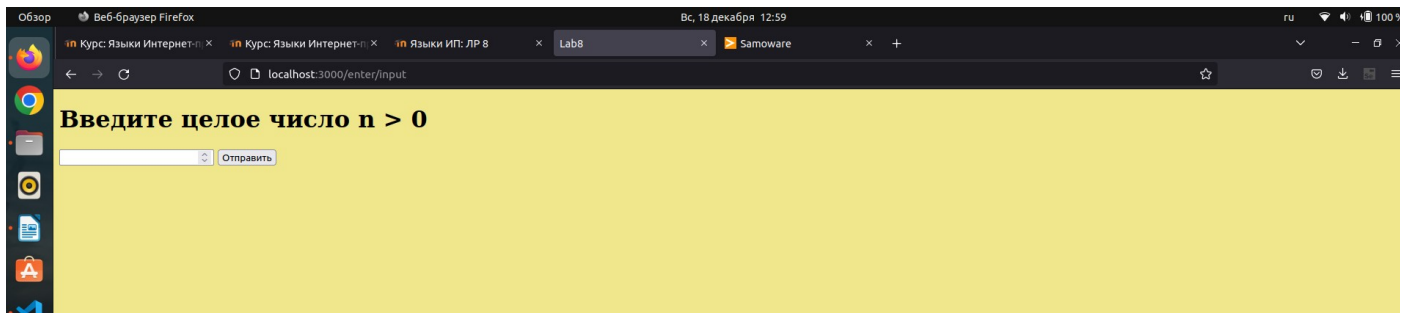
Разработать веб-приложение, имеющее HTML-страницу с формой ввода данных и HTML-страницу для представления результатов. Результат расчёта должен быть представлен в форме таблицы, оформленной с помощью элемента `table` или отдельными ячейками `div` и имеющей не менее двух колонок. Если по условию задания результат может быть представлен только в виде одной строки таблицы, необходимо реализовать вывод промежуточных результатов расчёта в качестве дополнительных строк. В этом случае первой колонкой таблицы будет порядковый номер итерации.

Под вводом с клавиатуры в тексте заданий следует понимать ввод в поле ввода данных формы на HTML-странице.

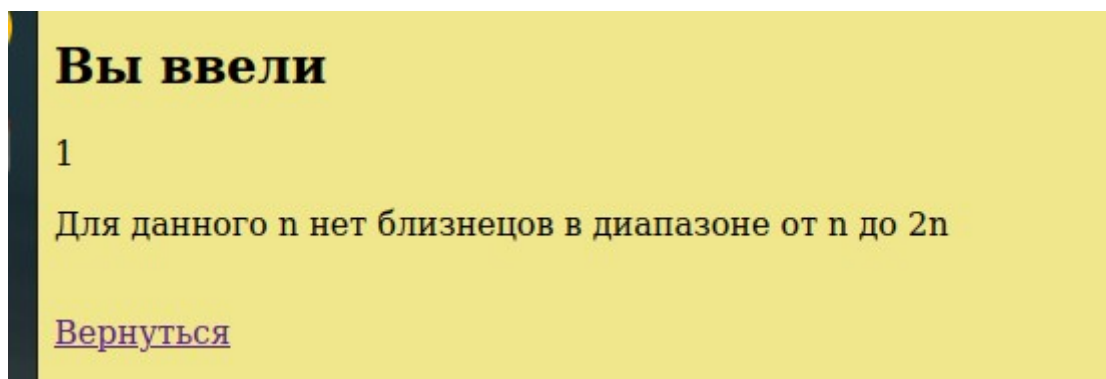
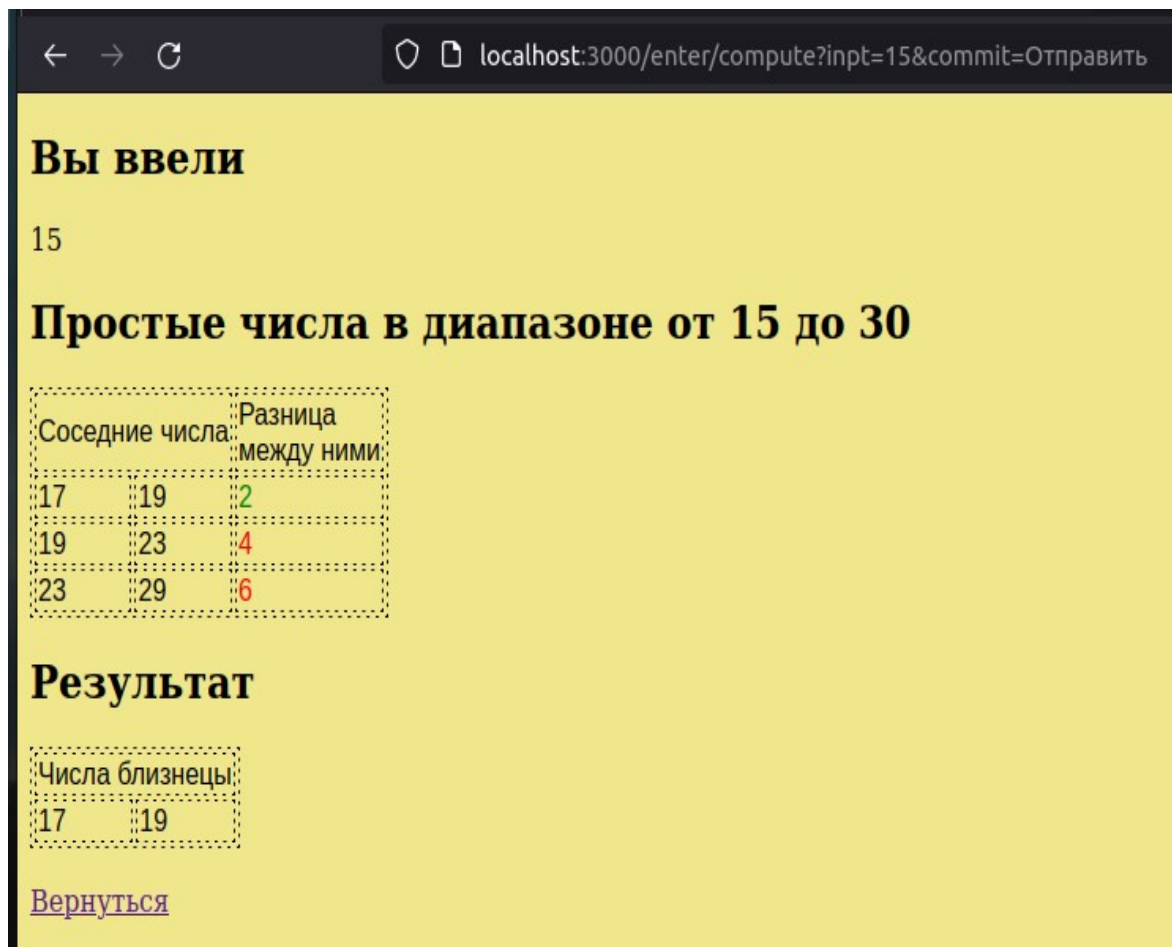
Текст задания:

Дано натуральное число n . Написать программу, определяющую, есть ли среди чисел $n, n + 1, n + 2, \dots, 2n$ близнецы, то есть простые числа, разность между которыми равна 2. Например, 11 и 13, 17 и 19. Вывести на печать все обнаруженные пары, а если таких пар нет, то соответствующее сообщение. Для определения близнецов и их наличия использовать функции.

Начальный экран

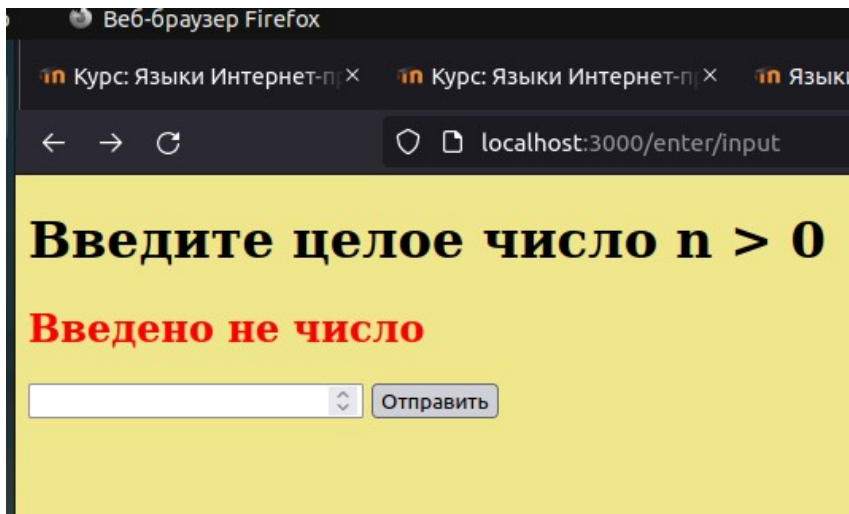


Ввод корректного значения

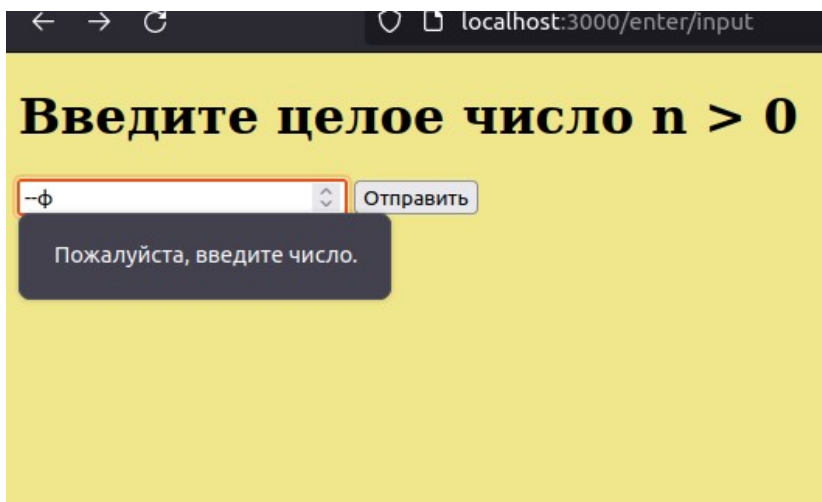


Ввод неправильных значений(в том числе через адресную строку)

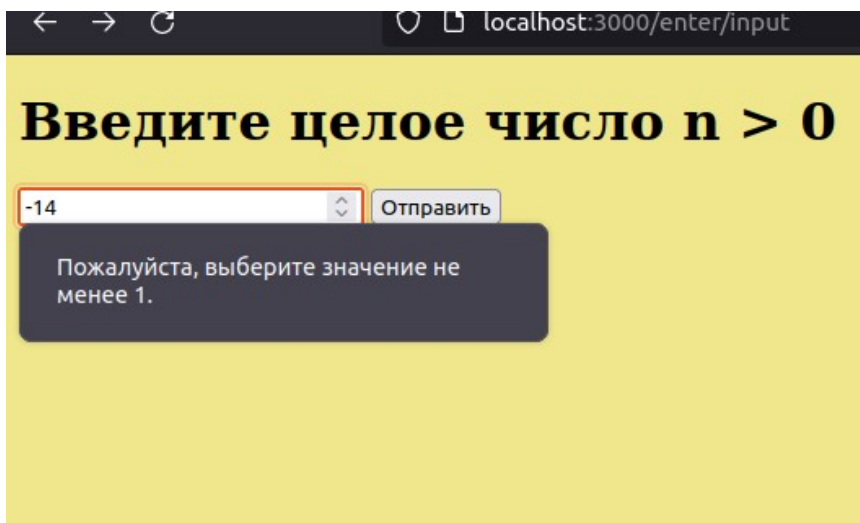
Ввод пустого значений



Ввод не чисел через форму



Ввод отрицательных чисел через форму



Ввод не чисел через адресную строку

localhost:3000/enter/compute?inpt=фцвфцв&commit=Отправить

Введите целое число $n > 0$

Введено не число

Ввод отрицательных чисел через адресную строку





localhost:3000/enter/compute?inpt=-123&commit=Отправить

Введите целое число $n > 0$

Введено отрицательное число

Измененные файлы

app/assets/stylesheets/application.css

```
laboratories > 8 > lab8_ > app > assets > stylesheets > # application.css > table
1  table, tr, td{
2      border: 1px  black dashed;
3      font-family: Arial;
4  }
5
6  .right{
7      color:  green;
8  }
9
10 .wrong{
11     color:  red;
12 }
13
14 body{
15     background-color:  khaki;
16 }
```

app/assets/controllers/enter_controller.rb

```
1  # frozen_string_literal: true
2
3  require 'prime'
4
5  # Controller for app
6  class EnterController < ApplicationController
7    def input; end
8
9    def compute
10     @input = params[:input]
11     if !/^-\?\d+$/.match(@input)
12       flash[:notice] = 'Введено не число'
13     elsif !(@input = @input.to_i).positive?
14       flash[:notice] = 'Введено отрицательное число'
15     else
16       @result = twins(@input)
17       return @prime_arr = (@input..2 * @input).select(&:prime?)
18     end
19     redirect_to :input
20   end
21
22   private
23
24   # rubocop:disable Naming/MethodParameterName
25   def twins(n)
26     arr = []
27     primes = (n..2 * n).to_a.select(&:prime?)
28     (primes.length - 1).times do |i|
29       arr << [primes[i], primes[i + 1]] if primes[i + 1] - primes[i] == 2
30     end
31     return arr unless arr.length.zero?
32
33     nil
34   end
35   # rubocop:enable Naming/MethodParameterName
36 end
37
```

Ограничения rubosor на имена переменных выключена, так как вводимое число n описано в задании

app/assets/views/enter/compute.html.erb

```
laboratories > 8 > lab8_ > app > views > enter > <> compute.html.erb
1 |<h2>Вы ввели</h2>
2 |<p><%= @input %></p>
3 |
4 |<%unless @result.nil?%>
5 |
6 |<h2>Простые числа в диапазоне от <%= "#{@input} до #{@input * 2}" %></h2>
7 |<table>
8 |   <tr>
9 |     <td colspan = 2 >Соседние числа</td>
10 |    <td>Разница <br/> между ними</td>
11 |  </tr>
12 |  <%( @prime_arr.length - 1).times do |i| %>
13 |    <tr>
14 |      <td><%= @prime_arr[i] %></td>
15 |      <td><%= @prime_arr[i + 1] %></td>
16 |      <%= content_tag(:td, "#{diff = @prime_arr[i + 1] - @prime_arr[i]}", :class => ( (diff) == 2 ? 'right' : 'wrong') ) %>
17 |    </tr>
18 |  <%end%>
19 |
20 |</table>
21 |
22 |<h2>Результат</h2>
23 |<table>
24 |  <tr>
25 |    <td colspan = 2>Числа близнецы</td>
26 |  </tr>
27 |  <%= @result.each do |i| %>
28 |    <tr>
29 |      <td><%= i[0] %></td>
30 |      <td><%= i[1] %></td>
31 |    </tr>
32 |  <%end%>
33 |</table>
34 |<%else%><p>Для данного n нет близнецов в диапазоне от n до 2n</p>
35 |<%end%>
36 |<br/>
37 |<%= link_to 'Вернуться', 'input' %>
```

app/assets/views/enter/input.html.erb

```
laboratories > 8 > lab8_ > app > views > enter > <> input.html.erb
1 |
2 |<h1>Введите целое число n > 0</h1>
3 |<%unless flash[:notice].blank?%>
4 |  <h2 style = 'color:red'><%= flash[:notice] %></h2>
5 |
6 |<%end%>
7 |<%= form_with url: :compute, method: :get do |form| %>
8 |  <%= form.number_field :inpt, in: 1..Float::INFINITY %>
9 |  <%= form.submit "Отправить" %>
10 |<% end %>
11 |
12 |
```


spec/requests/specs.rb

```
laboratories > 8 > lab8_ > spec > requests > specs.rb
2  require 'spec_helper'
3
4  RSpec.describe 'Enters', type: :request do
5    before do
6      @cont = EnterController.new
7    end
8
9    describe 'GET /input' do
10     it 'returns http success' do
11       get '/enter/input'
12       expect(response).to have_http_status(:success)
13     end
14   end
15
16   describe 'GET /compute' do
17     it 'returns http redirection' do
18       get '/enter/compute'
19       expect(response).to have_http_status(:redirect)
20     end
21   end
22
23   describe 'Rendering templates' do
24     it 'return input template' do
25       get '/enter/input'
26       expect(response).to render_template('input')
27     end
28
29     it 'return compute template' do
30       get '/enter/compute?inpt=5'
31       expect(response).to render_template('compute')
32     end
33   end
34
35   describe 'Check returning params' do
36     it 'populates an array of articles' do
37       get '/enter/compute/?inpt=5'
38       expect(assigns(:result)).to eq([[5, 7]])
39     end
40
41     it 'populates an array of articles' do
42       get '/enter/compute/', params: { inpt: '1' }
43       expect(assigns(:result)).to be_nil
44     end
45   end
46 end
47
```

```
● kiryu@kiryu-UPC:~/VUZ/WPL/laboratories/8/lab8_$ rspec spec/requests/specs.rb
```

```
.....
Finished in 0.11676 seconds (files took 1.06 seconds to load)
6 examples, 0 failures
```


Проверка контроллера рубокопом

```
Exiting
● kiryu@kiryu-UPC:~/VUZ/WPL/laboratories/8/lab8$ rubocop app/controllers/enter_controller.rb
Inspecting 1 file
.

1 file inspected, no offenses detected
○ kiryu@kiryu-UPC:~/VUZ/WPL/laboratories/8/lab8$
```

Вывод: в данной лабораторной работе я научился создавать простейшие Rails приложения, работать с контроллерами и представлениями