



**«Московский государственный технический университет
имени Н.Э. Баумана»
(МГТУ им. Н.Э. Баумана)**

ФАКУЛЬТЕТ Информатика и системы управления
КАФЕДРА Компьютерные системы и сети

**О т ч е т
по Лабораторной работе №10**

Дисциплина: Языки интернет программирования

**Название лабораторной работы: Формирование XML. Отображение XML в
HTML средствами сервера и клиента. XSLT. Тесты.**

Вариант 24

Студент гр. ИУ6-32 _____
(Подпись, дата)

Скударнов И. В.
(И.О. Фамилия)

Преподаватель _____
(Подпись, дата)

Самарёв Р. С.
(И.О. Фамилия)

Задание:

Модифицировать код лабораторной работы 8 таким образом, чтобы по запросу с указанными параметрами выдавался результат в формате XML. Разработать второе приложение, являющееся посредником между клиентом и первым приложением, задачей которого является преобразование XML в HTML или передача в неизменном виде браузеру для отображения браузером.

Код:

palind_controller.rb – контроллер первого приложения

```
class PalindController < ApplicationController
  def input
    end

  def palind(n)
    mas = Array.new(n+1)
    mas[0] = 0
    for i in 11..n
      ik = i**2
      if ik.to_s == ik.to_s.reverse
        mas[i] = ik
        mas[0] += 1
        puts "Число #{i}, его квадрат #{mas[i]}"
      else
        mas[i]=nil
      end
    end
    return mas
  end

  def view
    @n1 = params[:n1].to_i
    @result = palind(@n1)
    format_answer(@n1, @result)
  end

  def format_answer(n1, result)
    respond_to do |format|
      format.html
      format.xml do
        render xml: {
          result: {
            description: 'Result Array',
            type: 'Array', value: result
          }
        }
      end
    end
  end
end
```

palind_controller_test.rb – функциональные тесты контроллера

```
require 'test_helper'
```

```
class PalindControllerTest < ActionDispatch::IntegrationTest
```

```
  test "should get input" do
    get palind_input_url
    assert_response :success
  end
```

```
  test "should get view" do
    get palind_view_url
    assert_response :success
  end
```

```
  test 'should get 3' do
    get palind_view_url, params: {n1: 30}
    answer = 3
    result = controller.instance_variable_get(:@result)[0]
    assert_equal answer, result
  end
```

```
  test 'should get 5' do
    get palind_view_url, params: {n1: 112}
    answer = 5
    result = controller.instance_variable_get(:@result)[0]
    assert_equal answer, result
  end
```

```
  test 'should correct format xml to hash' do
    get '/palind/view.xml', params: {n1: 50}
    answer = {
      'hash' => {
        'result' => {
          'description' => "Result Array",
          'type' => 'Array', 'value' => assigns[:result]
        }
      }
    }
    assert_equal answer, Hash.from_xml(response.body)
  end
```

```
  test 'should have correct result tag' do
    get '/palind/view.xml', params: {n1: 50}
    assert_select 'result' do
      assert_select 'type', 'Array'
      result_array_value_answer = { 'value' => assigns[:result]}
      assert_select 'value>value', result_array_value_answer
    end
  end
end
```

transformer.xslt – код XSLT-программы

```
<?xml version="1.0" encoding="UTF-8"?>
<xsl:stylesheet version="1.0" xmlns:xsl="http://www.w3.org/1999/XSL/Transform">
  <xsl:template match="/">
    <html>
      <body>
        <p>palindroms were found<p>
        <div class='table'>
          <div class='row header'>
            <div class='cell'>Number</div>
            <div class='cell'>Squared Number</div>
          </div>
          <xsl:for-each select="//result/value/value">
            <xsl:choose>
              <xsl:when test="not( = '' ) and not(position() = '1')">
                <div class='row'>
                  <div class='cell'><xsl:value-of select="position()"/></div>
                  <div class='cell'><xsl:value-of select="."/></div>
                </div>
              </xsl:when>
            </xsl:choose>
          </xsl:for-each>
        </div>
      </body>
    </html>
  </xsl:template>
</xsl:stylesheet>
```

xml_transformer_controller.rb – контроллер второго приложения

```
require 'nokogiri'
```

```
class XmlTransformerController < ApplicationController
```

```
  def input
    end
```

```
  def answer
```

```
    page =
```

```
Nokogiri::XML(open("http://localhost:3000/palind/view.xml?utf8=%E2%9C%93&n1=#{params[:n1]}&commit=Show+palindroms"))
```

```
    xslt = Nokogiri::XSLT(File.read('public/transformer.xslt'))
```

```
    xslt_href = Nokogiri::XML::ProcessingInstruction.new(page, 'xml:stylesheet', 'type="text/xsl" href="transformer.xslt"')
```

```
    request.format = params[:format].to_sym
```

```
    respond_to do |format|
```

```
      format.html { render html: xslt.transform(page).to_s.html_safe }
```

```
      format.xml do
```

```
        page.at('hash').add_previous_sibling xslt_href
```

```
        render xml: page
```

```
      end
```

```
    end
```

```
  end
```

```
end
```

input.html.erb

```
<h1>XmlTransformer#input</h1>
<p>Find me in app/views/xml_transformer/input.html.erb</p>
<div>
  <%= form_tag('/palind/view', :method => 'get') do %>
    <div>
      <%= label_tag('Enter n', nil, for: 'input_n1') %>
      <%= text_field_tag(:n1, nil, id: 'input_n1') %>
    </div>

    <br/>

    <div>
      <%= radio_button_tag(:format, 'html', true, id: 'choose_html') %>
      <%= label_tag('html_answer', ('HTML answer').html_safe, for: 'choose_html') %>
      <%= radio_button_tag(:format, 'xml', false, id: 'choose_xml') %>
      <%= label_tag('xml_answer', ('XML answer').html_safe, for: 'choose_xml') %>
    </div>

    <br/>

    <div>
      <%= submit_tag("Show palindroms") %>
    </div>
  <% end %>
</div>
```

xml_transformer_controller_test.rb – тесты контроллера

require 'test_helper'

```
class XmlTransformerControllerTest < ActionDispatch::IntegrationTest
  test "should get input" do
    get xml_transformer_input_url
    assert_response :success
  end

  test 'should get answer html' do
    get '/palind/view', params: { format: 'html' }
    assert_response :success
  end

  test 'should get answer xml' do
    get '/palind/view', params: { format: 'xml' }
    assert_response :success
  end

  test 'result from first app shold be correct' do
    n1_value = rand(1000)
    page = Nokogiri::XML(open("http://localhost:3000/palind/view.xml?n1=#{n1_value}"))
    assert_equal palind(n1_value), page.at('result>value').elements.map { |el| el.children.text.to_i }
  end

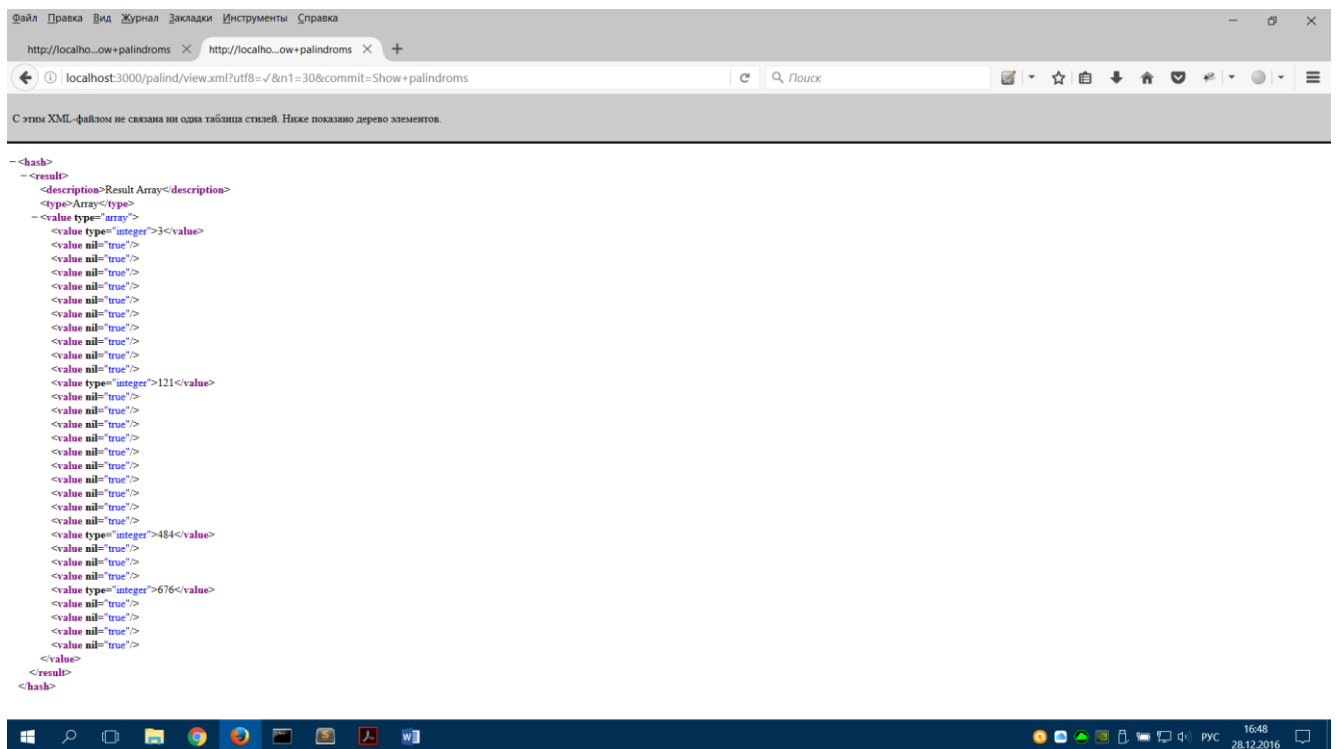
  def palind(n)
    mas = Array.new(n+1)
```

```

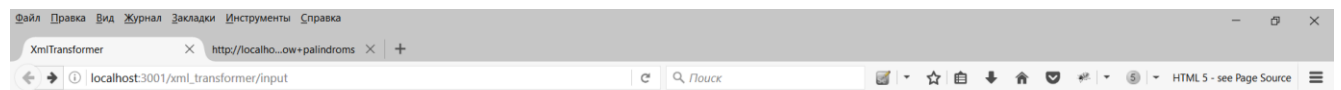
mas[0] = 0
for i in 1..10
    mas[i]=0
end
for i in 11..n
    ik = i**2
    if ik.to_s == ik.to_s.reverse
        mas[i] = ik
        mas[0] += 1
        puts "Число #{i}, его квадрат #{mas[i]}"
    else
        mas[i]=0
    end
end
end
return mas
end
end

```

Скриншоты:



```
C:\ruby\lr10\xml>ruby transformation.rb
<html><body><p>palindroms were found<p><div class="table">
<div class="row header">
<div class="cell">Number</div>
<div class="cell">Squared Number</div>
</div>
<div class="row">
<div class="cell">12</div>
<div class="cell">121</div>
</div>
<div class="row">
<div class="cell">23</div>
<div class="cell">484</div>
</div>
<div class="row">
<div class="cell">27</div>
<div class="cell">676</div>
</div>
</div></p></p></body></html>
```



XmlTransformer#input

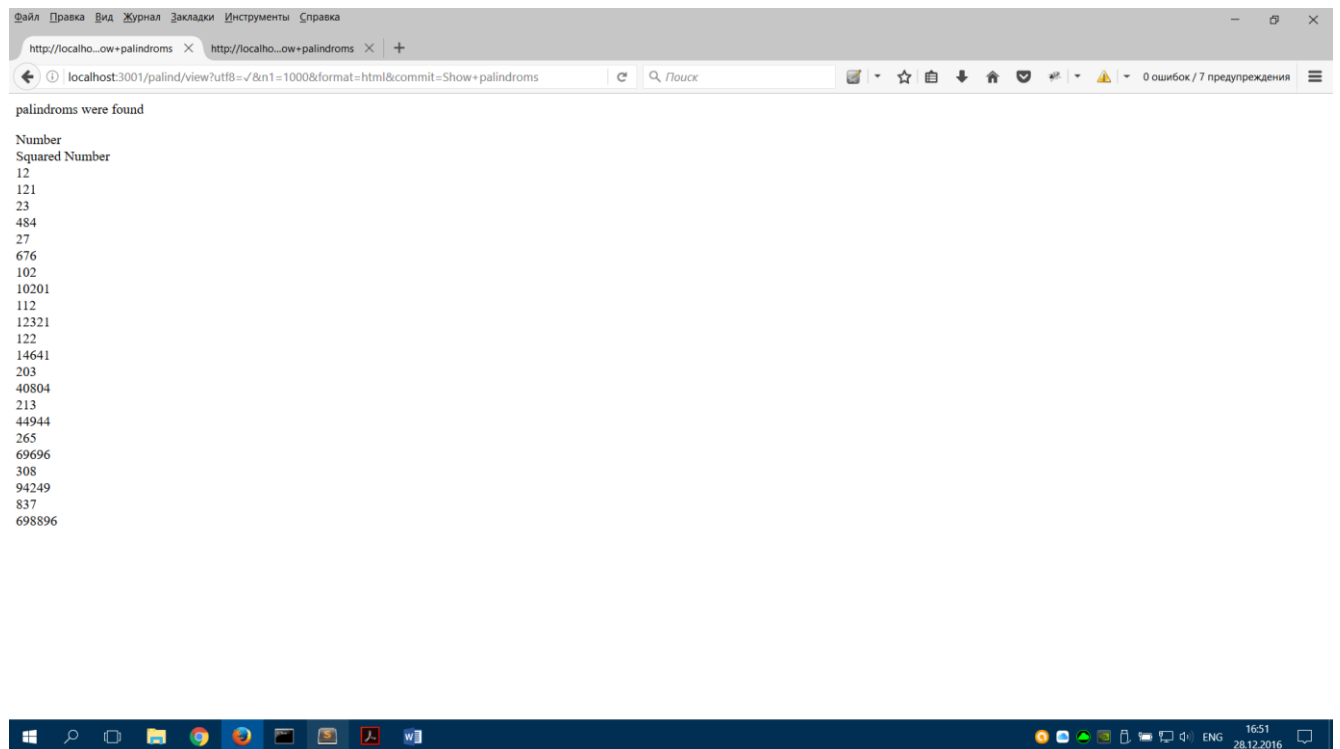
Find me in app/views/xml_transformer/input.html.erb

Enter n

☒ HTML answer ☐ XML answer

[Show palindroms](#)





Вывод: научился организовывать вывод информации в формате XML. Проверять формирование XML. Разрабатывать XSLT-программу преобразования XML в HTML.