

*Федеральное государственное бюджетное образовательное учреждение
высшего профессионального образования*



*«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

Лабораторная работа 11

Студент гр.ИУ6-32 _____ Липко Д.Н.
(Подпись, дата) (И.О. Фамилия)

Преподаватель _____
(Подпись, дата) (И.О. Фамилия)

Модифицировать код ЛР 8 таким образом, чтобы запросы, которые были ранее выполнены, сохранялись в БД и при следующем запросе не требовали повтора вычислений.

- Сформировать модель в соответствии с потребностями хранения данных. Входные параметры являются ключами, по которым извлекается результат.
- Выполнить создание БД и миграцию соответствующими запросами rake.
- Написать тест на добавление и поиск данных с помощью модели. Проверить выполнение теста.
- Модифицировать код приложения таким образом, чтобы результат вычислений преобразовывался в строковый или бинарный формат (на выбор: json, xml, и пр.). Проверить через отладочную печать в консоль, что преобразование выполняется корректно.
- Вставить код для сохранения данных в БД и запрос на поиск предыдущего результата вычислений.
- Добавить действие в контроллер, позволяющее определить, что хранится в БД через сериализацию в XML.
- Проверить, что при выполнении запроса, данные добавляются в БД.
- При помощи консоли сообщений Webrick определить, производится ли поиск результата предыдущего запроса в БД и не повторяются ли одни и те же вычисления.
- Модифицировать модель таким образом, чтобы добавление записей с одинаковыми параметрами было невозможно.
- Реализовать тест модели, проверяющий невозможность повторного добавления одних и тех же результатов вычислений.
- Реализовать функциональный тест, проверяющий, что результаты вычислений различны при различных входных параметрах.
- Проверить маршруты приложения с помощью rake routes и убрать лишние. Обеспечить доступ при обращении по адресу /.

Polin_controller.rb

```
class CalcController < ApplicationController
  def input
  end

  def ifPalindrom(num)
    #mas = num.split(/[1-9]/)
    mas = num.split("")
    #p mas
    n = mas.length / 2
    #1234321
    ifpolindrom = true
    for i in 0..n
      if mas[i] != mas[mas.length - 1 - i]
        ifpolindrom = false
      end
    end
    ifpolindrom
  end

  def Change(num)
    mas = num.split("")
    n = mas.length / 2
    n -= 1
    for i in 0..n
      c = mas[i]
      mas[i] = mas[mas.length - 1 - i]
      mas[mas.length - 1 - i] = c
    end

    (mas.join.to_i + num.to_i).to_s
  end

  def view
    calc = Calculation.find_by(input: params[:v1])

    if calc.blank?
      # record not found
      str = params[:v1].to_s

      resul_array = []
      count = 0
      resul_array[count] = str
      while !ifPalindrom(str)
```

```

        str = Change(str)
        count += 1
        resul_array[count] = str
        #p str
    end

    #v1 = params[:v1].to_s
    @result = resul_array
    Calculation.create(input: params[:v1], output: @result.last,
intermediate: resul_array.join(","))
    else
        @result = calc.intermediate_to_array
    end
    @result
    respond_to do |format|
        format.html
        format.json do
            render json: @result
        end
    end
end

def all
    res = Calculation.all

    render xml: res
end
end

```

input.html.erb

```
<h1>Calc#input</h1>
<p>Find me in app/views/calc/input.html.erb</p>
<%= form_tag("/calc/view", :method => "get") do %>
  <%= label_tag("Value number:") %>
  <%= text_field_tag(:v1) %> <br/>
<br/>
  <%= submit_tag("Calc result") %>
<% end %>
```

view.html.erb

```
<h1>Calc#view</h1>
<p>Find me in app/views/calc/view.html.erb</p>

<table border="1">
<caption>Table of result</caption>
  <tr>
    <th>i</th>
    <th>number</th>
  </tr>
  <% for i in 0..@result.length-1 %>
    <tr><td><%=i+1%></td><td><%=@result[i]%></td></tr>
  <%end%>

</table>

<%= link_to "Repeat calculation", :calc_input %>
```

Migrate

```
class CreateCalculations < ActiveRecord::Migration[5.0]
  def change
    create_table :calculations do |t|
      t.integer :input
      t.text :intermediate
      t.integer :output
      t.timestamps
    end
  end
end
```

Calc_controller_test.rb

```
require 'test_helper'
```

```
class CalcControllerTest < ActionDispatch::IntegrationTest
```

```
  test "should get input" do
```

```
    get calc_input_url
```

```
    assert_response :success
```

```
  end
```

```
  test "should get view db" do
```

```
    get calc_view_url, params: {v1: 12345}
```

```
    assert_equal ["12345", "66666"], assigns[:result]
```

```
  end
```

```
end
```

Проверка тестов

```
Администратор: C:\Windows\system32\cmd.exe
C:\Ruby\Laba_11\calc>rake test
Run options: --seed 1424

# Running:
...

Finished in 1.217729s, 2.4636 runs/s, 1.6424 assertions/s.
3 runs, 2 assertions, 0 failures, 0 errors, 0 skips
C:\Ruby\Laba_11\calc>
```

Работа веб-приложения

```
Администратор: C:\Windows\system32\cmd.exe - rails s
Use Ctrl-C to stop
Started GET "/" for ::1 at 2016-12-19 20:57:29 +0300
ActiveRecord::SchemaMigration Load (1.0ms) SELECT "schema_migrations".* FROM "schema_migrations"
Processing by CalcController#input as HTML
Rendering calc/input.html.erb within layouts/application
Rendered calc/input.html.erb within layouts/application (11.0ms)
Completed 200 OK in 850ms (Views: 818.7ms | ActiveRecord: 0.0ms)

Started GET "/calc/view?utf8=%E2%9C%93&v1=98764&commit=Calc+result" for ::1 at 2016-12-19 20:58:23 +0300
Processing by CalcController#view as HTML
Parameters: {"utf8"=>"?", "v1"=>"98764", "commit"=>"Calc result"}
Calculation Load (0.0ms) SELECT "calculations".* FROM "calculations" WHERE "calculations"."input" = ? LIMIT ? [{"input", 98764}, [{"LIMIT", 1}]]
(0.0ms) begin transaction
Calculation Exists (0.0ms) SELECT 1 AS one FROM "calculations" WHERE "calculations"."input" = ? LIMIT ? [{"input", 98764}, [{"LIMIT", 1}]]
SQL (1.0ms) INSERT INTO "calculations" ("input", "intermediate", "output", "created_at", "updated_at") VALUES (?, ?, ?, ?, ?) [{"input", 98764}, [{"intermediate", "98764,145553,501094,991199"}, [{"output", 991199}, [{"created_at", 2016-12-19 17:58:23 UTC}, [{"updated_at", 2016-12-19 17:58:23 UTC}]]
(111.0ms) commit transaction
Rendering calc/view.html.erb within layouts/application
Rendered calc/view.html.erb within layouts/application (0.0ms)
Completed 200 OK in 352ms (Views: 125.8ms | ActiveRecord: 113.0ms)

Started GET "/" for ::1 at 2016-12-19 20:58:46 +0300
Processing by CalcController#input as HTML
Rendering calc/input.html.erb within layouts/application
Rendered calc/input.html.erb within layouts/application (0.0ms)
Completed 200 OK in 145ms (Views: 126.7ms | ActiveRecord: 0.0ms)

Started GET "/calc/view?utf8=%E2%9C%93&v1=98764&commit=Calc+result" for ::1 at 2016-12-19 20:58:58 +0300
Processing by CalcController#view as HTML
Parameters: {"utf8"=>"?", "v1"=>"98764", "commit"=>"Calc result"}
Calculation Load (0.0ms) SELECT "calculations".* FROM "calculations" WHERE "calculations"."input" = ? LIMIT ? [{"input", 98764}, [{"LIMIT", 1}]]
Rendering calc/view.html.erb within layouts/application
Rendered calc/view.html.erb within layouts/application (1.0ms)
Completed 200 OK in 139ms (Views: 118.5ms | ActiveRecord: 0.0ms)
```

Вывод: мы научились формировать модель в соответствии с потребностями хранения данных(входные параметры являются ключами, по которым извлекается результат) Выполнили создание БД и миграцию соответствующими запросами rake. Написали тест на добавление и поиск данных с помощью модели. Проверили выполнение теста. Модифицировали код приложения таким образом, чтобы результат вычислений преобразовывался в строковый формат

(xml.).