

НАПРАВЛЕНИЕ ПОДГОТОВКИ 09.03.03 «Прикладная информатика»

по лабораторной работе № 11

Дисциплина: Языки Интернет-программирования

Д. В. Малахов
(И.О. Фамилия)

Москва, 2022

Цель работы: Освоить принципы применения средств объектно-реляционного преобразования в составе Ruby on Rails.

Задание:

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 Puma/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 /.

Тексты измененных файлов

seq_controller.rb

```
# frozen_string_literal: true
```

```

# Top-level class
class SeqController < ApplicationController
  def input; end

  def last
    @str = Sequence.last
  end

  def base
    @str = Sequence.all
  end

  def xml_str
    str = Sequence.all
    res = str.map { |s| [JSON.parse(s.given).to_s, JSON.parse(s.max_seq).to_s,
JSON.parse(s.all_seq).to_s] }
    render xml: res.to_xml
  end

  def show
    params[:str] ? @str = params[:str] : nil

    res = Sequence.find_by_given(ActiveSupport::JSON.encode(@str))

    if res
      @source = 'DataBase'
      @max_found = ActiveSupport::JSON.decode(res.max_seq)
      @found = ActiveSupport::JSON.decode(res.all_seq)
    else
      @source = 'Calculation'

      arr = @str&.split&.map(&:to_i)

      if arr && !arr.empty?
        tmp = helpers.find_seqs(arr)
        @found = tmp.map { |x| x.join(' ') }.join(' | ')
        @max_found = helpers.find_max(tmp)
      else
        @max_found = @str
        @found = @str
      end

      res = Sequence.create given: ActiveSupport::JSON.encode(@str), max_seq:
ActiveSupport::JSON.encode(@max_found),
                           all_seq: ActiveSupport::JSON.encode(@found)
      res.save
    end
  end
end

```

seq_helper.rb

```

# frozen_string_literal: true

# top-level documentation
module SeqHelper
  def find_seqs(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
  end
end

```

```

        end
      end

      tmp_found << arr[-1]
      found << tmp_found
      p found
      found
    end
  end

  def find_max(alr_found)
    sizes = alr_found.map(&:size)
    alr_found[sizes.index(sizes.max)].join(' ')
  end
end

```

sequence.rb

```
# frozen_string_literal: true
```

```

class Sequence < ApplicationRecord
  validates_uniqueness_of :given
end

```

application.html.erb

```

<!DOCTYPE html>
<html>
  <head>
    <title>Lab11</title>
    <meta name="viewport" content="width=device-width,initial-scale=1">
    <%= csrf_meta_tags %>
    <%= csp_meta_tag %>
    <link
href="https://cdn.jsdelivrivr.net/npm/bootstrap@5.2.2/dist/css/bootstrap.min.css"
rel="stylesheet" integrity="sha384-
Zenh87qX5JnK2Jl0vWa8Ck2rdkQ2Bzep5IDxbcnCeu0xjzrPF/et3URy9Bv1WTRi"
crossorigin="anonymous">

    <script
src="https://cdn.jsdelivrivr.net/npm/bootstrap@5.2.2/dist/js/bootstrap.bundle.min.js"
integrity="sha384-0ERcA2EqjJCMA+/3y+gxIOqMEjwtxJY7qPCqsdltbNJua0e923+mo//
f6V8Qbsw3" crossorigin="anonymous"></script>

    <%= stylesheet_link_tag "application", "data-turbo-track": "reload" %>
    <%= javascript_importmap_tags %>
  </head>

  <body class="bg-light">
    <div class="col-md-10 mx-auto my-4 p-5 shadow">
      <%= yield %>
    </div>
  </body>
</html>

```

input.html.erb

```

<h1 class="h1">Поиск возрастающих последовательностей</h1>
  <p>Вывести введенную цепочку, все найденные последовательности и наиболее
длинную из них.</p>

<%= form_tag("/seq/show", :method => "get") do %>
  <%= label_tag("Введите последовательность:") %>
  <%= text_field_tag(:str) %> <br/>
  <br/>
  <%= submit_tag("Найти", class: 'btn btn-primary') %>
<% end %>

```

```

</br>
<%= form_tag("seq/base", :method => "get") do%>
  <%= submit_tag("Посмотреть базу данных", class: 'btn btn-primary')%>
<% end%>
</br>
<%= form_tag("seq/xml_str", :method => "get", data:{turbo: false}) do%>
  <%= submit_tag("xml базы данных", class: 'btn btn-primary')%>
<% end%>
</br>
<%= form_tag("seq/last", :method => "get") do%>
  <%= submit_tag("Последнее добавление", class: 'btn btn-primary')%>
<% end%>

```

show.html.erb

```

<h1 class="h1">Результат</h1>
<p>Найдены последовательности</p>

<p id="str">Введенная строка: <%= @str %></p>
<p id="source">Источник: <%= @source %></p>
<p id="max found: ">Максимальная последовательность: <%= @max_found %></p>
<p id="found: ">Все последовательности: <%= @found %></p>

<%= link_to "Repeat input", '/' %>

```

last.html.erb

```

<h1 class="h1">Последнее вычисление:</h1>
<h1 class="h1">Последнее вычисление:</h1>
<div>
  <p>Промежуток <%= @str.given %></p>
  <p>Макс. последовательность: <%= @str.max_seq %> </p>
  <p>Все последовательности: <%= @str.all_seq %> </p>
  <p>Создано: <%= @str.created_at %></p>
  <%= link_to "Назад", "/" %>
</div>

```

base.html.erb

```

<h1>Содержимое базы данных</h3>
<table >
  <tr>
    <th> Данная строка </th>
    <th> Макс. последовательность </th>
    <th> Все последовательности </th>
    <th> Дата создания </th>
    <th> Время создания </th>
  </tr>

  <% @str.each do |i| %>
    <tr>
      <td> <%= i.given %> </td>
      <td> <%= i.max_seq %> </td>
      <td> <%= i.all_seq %> </td>
      <td> <%= i.created_at %> </td>
      <td> <%= i.updated_at %> </td>
    </tr>
  <tr></tr>
  <% end %>
</table>
<%= link_to "Назад", "/" %>

```

routes.rb

```

# frozen_string_literal: true

```

```

Rails.application.routes.draw do
  root 'seq#input', as: :home
  get 'seq/show'
  get 'seq/base'
  get 'seq/last'
  get 'seq/xml_str'
  # 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'

```

```

TEST_DATA = [
  {
    input: { str: '1 2 3 4 3 4 5 1 1 1 5 3 4 5 6 10 0 -1 -3' },
    output: ['1 2 3 4 3 4 5 1 1 1 5 3 4 5 6 10 0 -1 -3', '3 4 5 6 10',
            '1 2 3 4 | 3 4 5 | 1 | 1 | 1 5 | 3 4 5 6 10 | 0 | -1 | -3'],
  },
  {
    input: { str: '1 2 3 4 1 2 3 1 2' },
    output: ['1 2 3 4 1 2 3 1 2', '1 2 3 4', '1 2 3 4 | 1 2 3 | 1 2']
  },
  {
    input: { str: '38 23 20 28 49 22 39 9 3 56 4' },
    output: ['38 23 20 28 49 22 39 9 3 56 4', '20 28 49', '38 | 23 | 20 28 49 | 22
39 | 9 | 3 56 | 4']
  }
].freeze

```

```

# Top-level class

```

```

class SeqControllerTest < ActionDispatch::IntegrationTest

```

```

  test 'should get show' do
    get '/seq/show'
    assert_response :success
  end

```

```

  test 'should get input' do
    get '/'
    assert_response :success
  end

```

```

  test 'should return empty result' do
    input = { str: '' }
    get "/seq/show?#{input.to_query}"
    assert_equal('', assigns(:str))
    assert_equal('', assigns(:max_found))
    assert_equal('', assigns(:found))
  end

```

```

  test 'should return correct data' do
    TEST_DATA.each do |data|
      input = data[:input]
      output = data[:output]
      get "/seq/show?#{input.to_query}"
      assert_equal(output[0], assigns(:str))
      assert_equal(output[1], assigns(:max_found))
      assert_equal(output[2], assigns(:found))
    end
  end

```

```

end

test 'different data' do
  input1 = TEST_DATA[0][:input]
  input2 = TEST_DATA[1][:input]
  get "/seq/show?#{input1.to_query}"
  a = [assigns(:str), assigns(:max_found), assigns(:found)]
  # aa = assigns(:max_found)
  get "/seq/show?#{input2.to_query}"
  b = [assigns(:str), assigns(:max_found), assigns(:found)]
  # bb = assigns(:max_found)
  # assert_not_equal(aa, bb)

  a.each_index do |i|
    assert_not_equal(a[i], b[i])
  end
end

# test "should get 11 for view with with 1+10" do
#   get '/seq/show', params: {v1: 1, v2: 10, op: '+'}
#   assert_equal(assigns[:result], 11)
# end

# test "should get Unknown! for incorrect params" do
#   get '/seq/show'
#   assert_equal(assigns[:result], 'Unknown!')
# end
end

```

sequence_test.rb

```
# frozen_string_literal: true
```

```
require 'test_helper'
```

```
# top-level documentation
```

```
class SequenceTest < ActiveSupport::TestCase
```

```
  test 'cannot save twice' do
```

```
    res = Sequence.create given: '1 2 3', max_seq: '1 2 3', all_seq: '1 2 3'
```

```
    res.save
```

```
    res = Sequence.create given: '1 2 3', max_seq: '0 1 2 3', all_seq: '1 2 3'
```

```
    res.save
```

```
    res = Sequence.find_by_given('1 2 3')
```

```
    assert_equal(res.max_seq, '1 2 3')
```

```
  end
```

```
end
```

20221212175731_create_sequences.rb

```
# frozen_string_literal: true
```

```
# top-level documentation
```

```
class CreateSequences < ActiveRecord::Migration[7.0]
```

```
  def change
```

```
    create_table :sequences do |t|
```

```
      t.string :given
```

```
      t.string :max_seq
```

```
      t.string :all_seq
```

```
      t.timestamps
```

```
    end
```

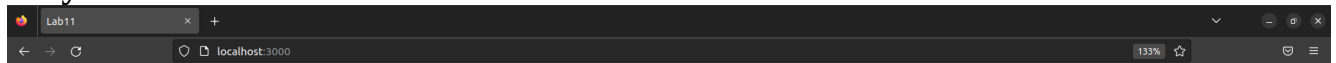
```
  end
```

```
end
```

schema.rb

```
ActiveRecord::Schema[7.0].define(version: 20_221_212_175_731) do
  create_table 'sequences', force: :cascade do |t|
    t.string 'given'
    t.string 'max_seq'
    t.string 'all_seq'
    t.datetime 'created_at', null: false
    t.datetime 'updated_at', null: false
  end
end
```

Результаты выполнения:



Поиск возрастающих последовательностей

Вывести введенную цепочку, все найденные последовательности и наиболее длинную из них.

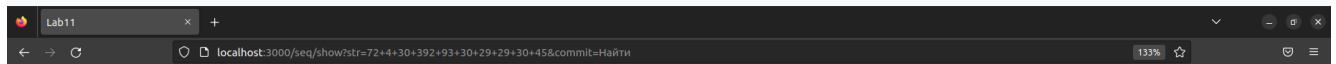
Введите последовательность:

Найти

Посмотреть базу данных

xml базы данных

Последнее добавление



Результат

Найдены последовательности

Введенная строка: 72 4 30 392 93 30 29 29 30 45

Источник: Calculation

Максимальная последовательность: 4 30 392

Все последовательности: 72 | 4 30 392 | 93 | 30 | 29 | 29 30 45

[Repeat input](#)

Содержимое базы данных

Данная строка	Макс. последовательность	Все последовательности	Дата создания	Время создания
"0 1 5 5 5 10 20 25 -5 1 5 25 125 625"	"-5 1 5 25 125 625"	"0 1 5 5 5 10 20 25 -5 1 5 25 125 625"	2022-12-16 14:50:30 UTC	2022-12-16 14:50:30 UTC
"2 2 3 4 1"	"2 3 4"	"2 2 3 4 1"	2022-12-16 14:51:31 UTC	2022-12-16 14:51:31 UTC
"1 2 3 4 2"	"1 2 3 4"	"1 2 3 4 2"	2022-12-16 15:02:47 UTC	2022-12-16 15:02:47 UTC
""	""	""	2022-12-16 15:05:10 UTC	2022-12-16 15:05:10 UTC
"1 2 3 6 2"	"1 2 3 6"	"1 2 3 6 2"	2022-12-16 15:53:33 UTC	2022-12-16 15:53:33 UTC
"1 2 3 4 3 4 5 1 1 1 5 3 4 5 6"	"1 2 3 4"	"1 2 3 4 3 4 5 1 1 1 5 3 4 5 6"	2022-12-20 17:31:26 UTC	2022-12-20 17:31:26 UTC
"72 4 30 392 93 30 29 29 30 45"	"4 30 392"	"72 4 30 392 93 30 29 29 30 45"	2022-12-20 17:32:05 UTC	2022-12-20 17:32:05 UTC

[Назад](#)

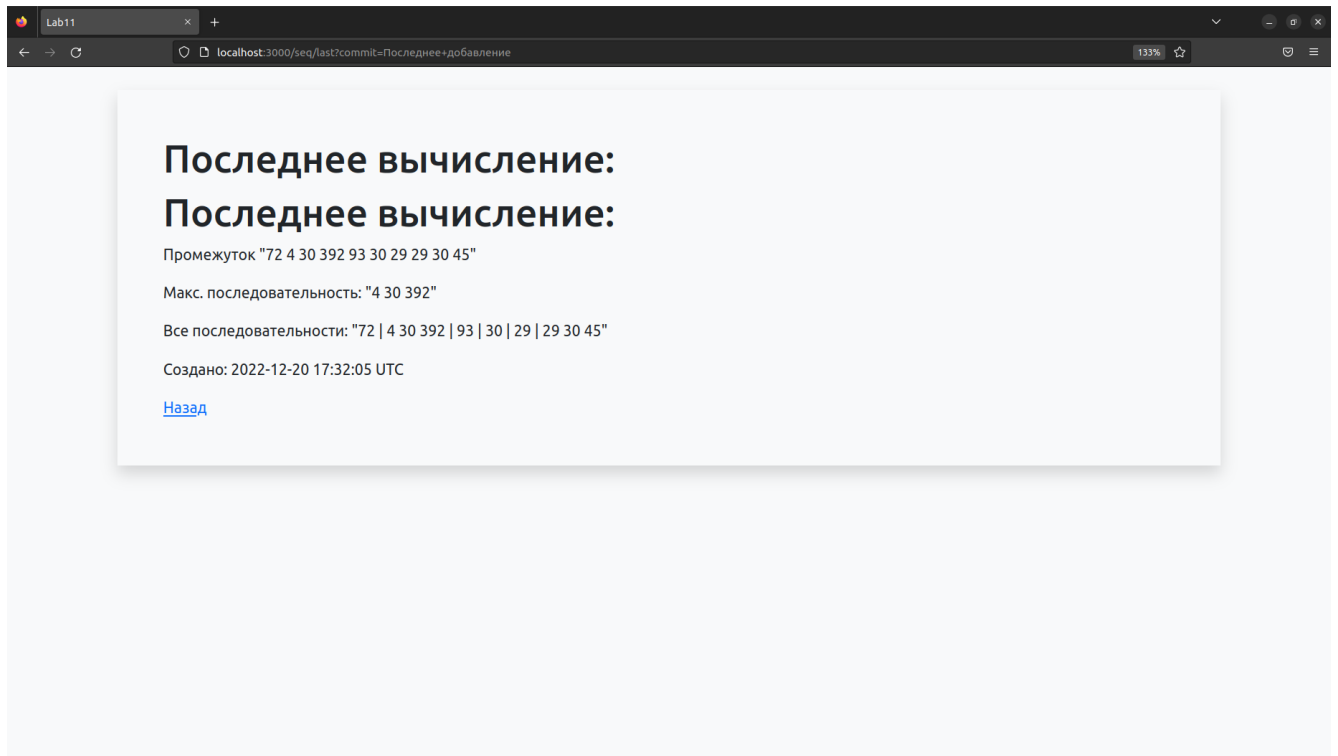


localhost:3000/seq/xml_str?×



localhost:3000/seq/xml_str?commit=xml+базы+данных

```
-<array type="array">
  <array>0 1 5 5 5 10 20 25 -5 1 5 25 125 625</array>
  <array>-5 1 5 25 125 625</array>
  <array>0 1 5 | 5 | 5 10 20 25 | -5 1 5 25 125 625</array>
</array>
-<array type="array">
  <array>2 2 3 4 1</array>
  <array>2 3 4</array>
  <array>2 | 2 3 4 | 1</array>
</array>
-<array type="array">
  <array>1 2 3 4 2</array>
  <array>1 2 3 4</array>
  <array>1 2 3 4 | 2</array>
</array>
-<array type="array">
  <array/>
  <array/>
  <array/>
</array>
-<array type="array">
  <array>1 2 3 6 2</array>
  <array>1 2 3 6</array>
  <array>1 2 3 6 | 2</array>
</array>
-<array type="array">
  <array>1 2 3 4 3 4 5 1 1 1 5 3 4 5 6</array>
  <array>1 2 3 4</array>
  <array>1 2 3 4 | 3 4 5 | 1 | 1 | 1 5 | 3 4 5 6</array>
</array>
-<array type="array">
  <array>72 4 30 392 93 30 29 29 30 45</array>
  <array>4 30 392</array>
  <array>72 | 4 30 392 | 93 | 30 | 29 | 29 30 45</array>
</array>
</arrays>
```



Консоль при добавлении результатов:

```
Терминал

Started GET "/seq/show?str=72+4+30+392+93+30+29+29+30+45&commit=%D0%9D%D0%B0%D0%B9%D1%82%D0%B8" for 127.0.0.1 at 2022-12-20 20:32:05 +0300
Processing by SeqController#show as HTML
Parameters: {"str"=>"72 4 30 392 93 30 29 29 30 45", "commit"=>"Найти"}
Sequence Load (0.2ms) SELECT "sequences".* FROM "sequences" WHERE "sequences"."given" = ? LIMIT ? [["given", "\"72 4 30 392 93 30 29 29 30 45\""], ["LIMIT", 1]]
↳ app/controllers/seq_controller.rb:24:in `show'
[[72], [4, 30, 392], [93], [30], [29], [29, 30, 45]]
TRANSACTION (0.1ms) begin transaction
↳ app/controllers/seq_controller.rb:44:in `show'
Sequence Exists? (0.4ms) SELECT 1 AS one FROM "sequences" WHERE "sequences"."given" = ? LIMIT ? [["given", "\"72 4 30 392 93 30 29 29 30 45\""], ["LIMIT", 1]]
↳ app/controllers/seq_controller.rb:44:in `show'
Sequence Create (0.6ms) INSERT INTO "sequences" ("given", "max_seq", "all_seq", "created_at", "updated_at") VALUES (?, ?, ?, ?, ?) [["given", "\"72 4 30 392 93 30 29 29 30 45\"", ["max_seq", "\"4 30 392\"", ["all_seq", "\"72 | 4 30 392 | 93 | 30 | 29 | 29 30 45\"", ["created_at", "2022-12-20 17:32:05.332763"], ["updated_at", "2022-12-20 17:32:05.332763"]]]
↳ app/controllers/seq_controller.rb:44:in `show'
TRANSACTION (5.1ms) commit transaction
↳ app/controllers/seq_controller.rb:44:in `show'
TRANSACTION (0.2ms) begin transaction
↳ app/controllers/seq_controller.rb:46:in `show'
Sequence Exists? (0.4ms) SELECT 1 AS one FROM "sequences" WHERE "sequences"."given" = ? AND "sequences"."id" != ? LIMIT ? [["given", "\"72 4 30 392 93 30 29 29 30 45\"", ["id", 7], ["LIMIT", 1]]
↳ app/controllers/seq_controller.rb:46:in `show'
TRANSACTION (0.1ms) commit transaction
↳ app/controllers/seq_controller.rb:46:in `show'
Rendering layout layouts/application.html.erb
Rendering seq/show.html.erb within layouts/application
Rendered seq/show.html.erb within layouts/application (Duration: 0.1ms | Allocations: 32)
```

Консоль при получении результата из базы данных:

```
Терминал

Started GET "/seq/show?str=0+1+5+5+5+10+20+25+-5+1+5+25+125+625&commit=%D0%9D%D0%B0%D0%B9%D1%82%D0%B8" for 127.0.0.1 at 2022-12-20 20:35:20 +0300
Processing by SeqController#show as HTML
  Parameters: {"str"=>"0 1 5 5 5 10 20 25 -5 1 5 25 125 625", "commit"=>"Найти"}
  Sequence Load (0.3ms)  SELECT "sequences".* FROM "sequences" WHERE "sequences"."given" = ? LIMIT ? [{"given", "\"0 1 5 5 5 10 20 25 -5 1 5 25 125 625\""}, ["LIMIT", 1]]
    ↳ app/controllers/seq_controller.rb:24:in `show'
  Rendering layout layouts/application.html.erb
  Rendering seq/show.html.erb within layouts/application
  Rendered seq/show.html.erb within layouts/application (Duration: 0.3ms | Allocations: 32)
  Rendered layout layouts/application.html.erb (Duration: 10.0ms | Allocations: 2362)
Completed 200 OK in 15ms (Views: 11.2ms | ActiveRecord: 0.3ms | Allocations: 3155)
```

Тестирование:

для контроллера:

```
Терминал

s3r6anita@s3r6anita-Modern-14-B10MW:~/LoIP/labs/lab11$ rails test:controllers
Running 5 tests in a single process (parallelization threshold is 50)
Run options: --seed 42385

# Running:

..[[1, 2, 3, 4], [3, 4, 5], [1], [1], [1, 5], [3, 4, 5, 6, 10], [0], [-1], [-3]]
[[1, 2, 3, 4], [1, 2, 3], [1, 2]]
[[38], [23], [20, 28, 49], [22, 39], [9], [3, 56], [4]]
.[[1, 2, 3, 4], [3, 4, 5], [1], [1], [1, 5], [3, 4, 5, 6, 10], [0], [-1], [-3]]
[[1, 2, 3, 4], [1, 2, 3], [1, 2]]
..

Finished in 0.227772s, 21.9518 runs/s, 74.6361 assertions/s.
5 runs, 17 assertions, 0 failures, 0 errors, 0 skips
s3r6anita@s3r6anita-Modern-14-B10MW:~/LoIP/labs/lab11$
```

для модели

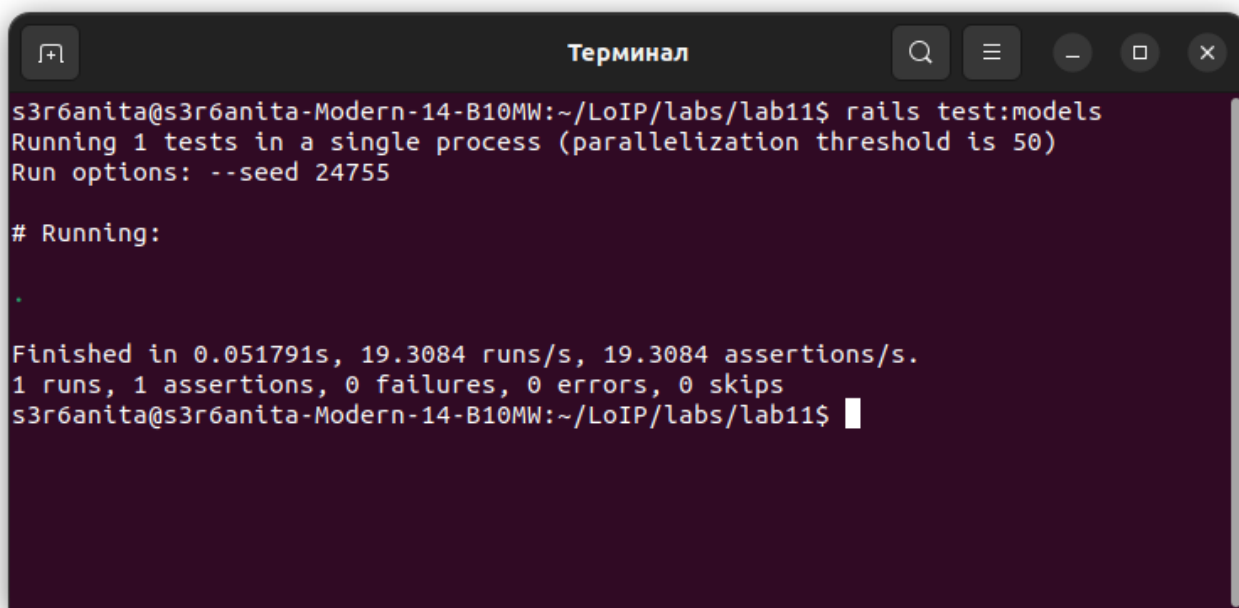


```
s3r6anita@s3r6anita-Modern-14-B10MW:~/LoIP/labs/lab11$ rails test:models
Running 1 tests in a single process (parallelization threshold is 50)
Run options: --seed 3836

# Running:

.

Finished in 0.026977s, 37.0690 runs/s, 37.0690 assertions/s.
1 runs, 1 assertions, 0 failures, 0 errors, 0 skips
s3r6anita@s3r6anita-Modern-14-B10MW:~/LoIP/labs/lab11$
```



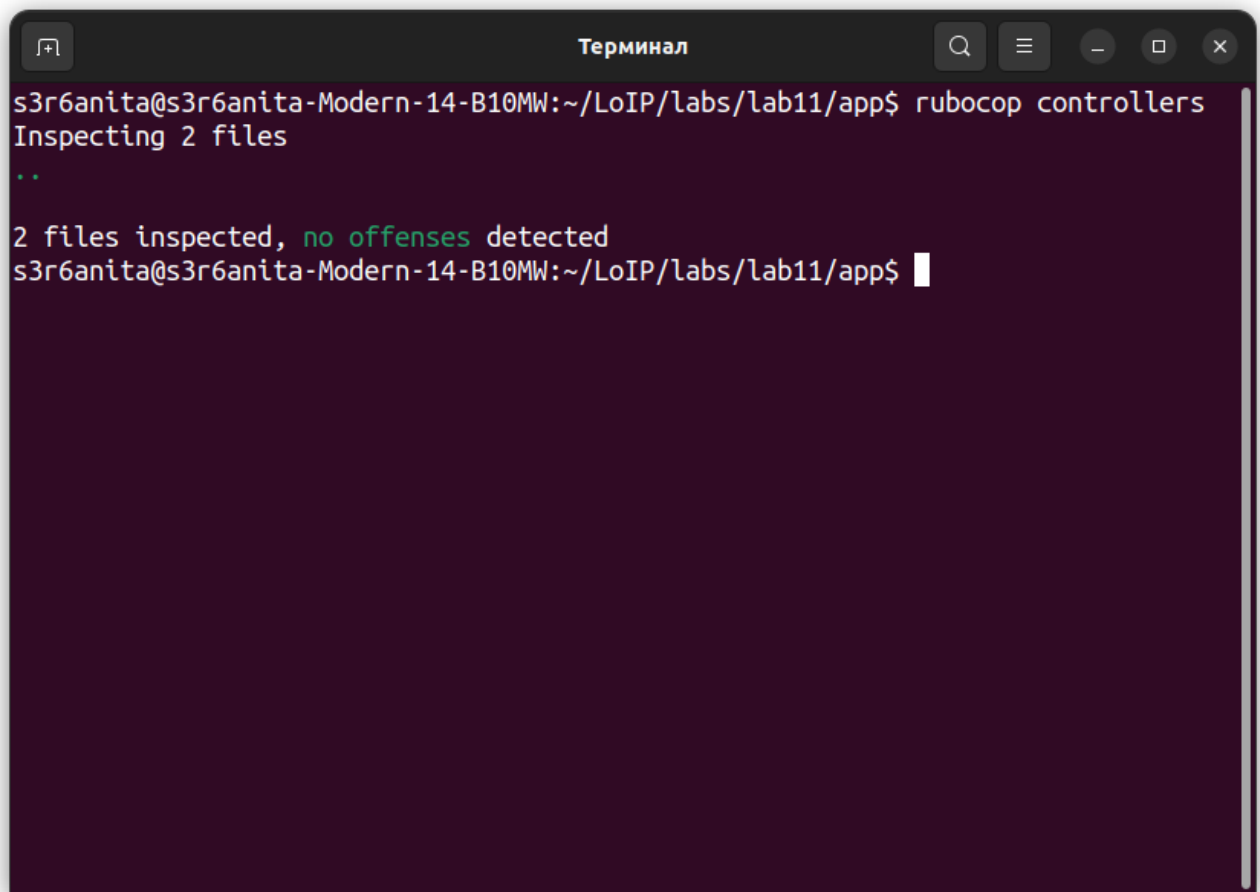
```
s3r6anita@s3r6anita-Modern-14-B10MW:~/LoIP/labs/lab11$ rails test:models
Running 1 tests in a single process (parallelization threshold is 50)
Run options: --seed 24755

# Running:

.

Finished in 0.051791s, 19.3084 runs/s, 19.3084 assertions/s.
1 runs, 1 assertions, 0 failures, 0 errors, 0 skips
s3r6anita@s3r6anita-Modern-14-B10MW:~/LoIP/labs/lab11$
```

Результаты проверки анализатором rubocop:

A terminal window titled "Терминал" (Terminal) with a dark background. The prompt is "s3r6anita@s3r6anita-Modern-14-B10MW:~/LoIP/labs/lab11/app\$". The command "rubocop controllers" has been executed. The output shows "Inspecting 2 files", followed by two dots on separate lines, and then "2 files inspected, no offenses detected". The prompt "s3r6anita@s3r6anita-Modern-14-B10MW:~/LoIP/labs/lab11/app\$" is shown again with a cursor.

```
s3r6anita@s3r6anita-Modern-14-B10MW:~/LoIP/labs/lab11/app$ rubocop controllers
Inspecting 2 files
..

2 files inspected, no offenses detected
s3r6anita@s3r6anita-Modern-14-B10MW:~/LoIP/labs/lab11/app$
```

Вывод: в результате выполнения были освоены принципы применения средств объектно-реляционного преобразования в составе Ruby on Rails.