|  |  |
| --- | --- |
| 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 **09.03.01 Информатика и вычислительная техника**

**Отчет**

|  |  |
| --- | --- |
| **по лабораторной работе №** | 11 |

**Дисциплина:** Языки интернет-программирования

**Вариант:** 23

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Студент | ИУ6-33Б |  |  | Д. И. Мироненко |
|  | (Группа) |  | (Подпись, дата) | (И.О. Фамилия) |
|  |  |  |  |  |
| Преподаватель |  |  |  | В. Д. Шульман |
|  |  |  | (Подпись, дата) | (И.О. Фамилия) |

Москва, 2023

**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 /.

**Код программы**

**Файл “app/controllers/calc\_controller.rb”**

|  |
| --- |
| # frozen\_string\_literal: true  class CalcController < ApplicationController  def index; end  def all  respond\_to do |format|  format.xml { render xml: Calc.all.to\_xml }  format.html { @calcs = Calc.order(clicks: :desc) }  end  end  def result  raise ArgumentError if params[:value].empty?  array = params[:value].split  array.length.times do |i|  value = Float(array[i])  value = value.to\_i if (value % 1).zero?  array[i] = value  end  @calc = Calc.find\_by(input\_sequence: array.join(' '))  @calc = Calc.save\_sequence(array) if @calc.nil?  redirect\_to calc\_show\_url(id: @calc.id)  rescue ArgumentError  redirect\_to root\_path  end  def show  id = Integer(params[:id])  if id && Calc.find\_by(id: id)  @calc = Calc.find(id)  end  end  end |

**Файл “app/views/calc/all.html.erb”**

|  |
| --- |
| <div class="container mt-4">  <ul>  <% @calcs.length.times do |i| %>  <div class="card mt-4">  <div class="card-body">  <h3 class="card-title">Результат <%= i + 1 %></h3><br>  <% sequences = @calcs[i].output\_sequences.split('\n') %>  <% (sequences.length - 1).times.each do |i| %>  <p><strong>Sequence</strong> <%= i + 1 %>: <%= sequences[i] %></p>  <% end %>  <p><strong>Longest Sequence</strong>: <%= sequences[sequences.length - 1] %></p>  </div>  </div>  <% end %>  </ul>  </div> |

**Файл “app/views/calc/index.html.erb”**

|  |
| --- |
| <div class="container">  <div class="card mt-5">  <div class="card-body">  <%= bootstrap\_form\_with url: calc\_result\_url, method: :post, local: true, remote: false do |f| %>  <strong>Задание:</strong>  <p>Написать программу, которая вводит цепочку целых чисел (количество чисел не менее 10) и определяет наиболее длинную монотонно возрастающую их последовательность. Вывести на печать введенную цепочку, все найденные последовательности и наиболее длинную из них. При программировании использовать функцию.</p>  <%= f.text\_field :value, required: true, label: 'Массив: ' %>  <br>  <%= f.submit "Вычислить", class: 'btn btn-primary' %>  <% end %>  </div>  </div>  <br>  <div>  <p>  <%= link\_to "БД в HTML формате", calc\_all\_url, class: 'btn btn-secondary' %>  <%= link\_to "БД в XML формате", calc\_all\_url(format: :xml), class: 'btn btn-secondary' %>  </p>  </div>  </div> |

**Файл “app/views/calc/show.html.erb”**

|  |
| --- |
| <main class="container mt-5">  <% if @error %>  <div class="alert alert-danger" role="alert">  <p><%= @error %></p>  </div>  <% else %>  <div class="card">  <div class="card-body">  <h3 class="card-title">Результат:</h3><br>  <% sequences = @calc.output\_sequences.split('\n') %>  <% (sequences.length - 1).times.each do |i| %>  <p><strong>Sequence</strong> <%= i + 1 %>: <%= sequences[i] %></p>  <% end %>  <p><strong>Longest Sequence</strong>: <%= sequences[sequences.length - 1] %></p>  </div>  </div>  <% end %>  <br>  <%= link\_to "Назад", root\_url, class: "btn btn-primary" %>  </main> |

**Работа программы**

A screenshot of a computer

Description automatically generated

**Рисунок 1** – ввод значений

**A white rectangular object with a black border

Description automatically generated**

**Рисунок 2** – вывод результатов

**A screenshot of a computer

Description automatically generated**

**Рисунок 3** – сериализация БД в HTML

**A screenshot of a computer

Description automatically generated**

**Рисунок 4** – сериализация БД в XML

**A screenshot of a computer

Description automatically generated**

**Рисунок 5** – проверка кода rubocop’ом

**Вывод**

Изучена технология ORM для взаимодействия между Rails и базой данных; полученные знания отработаны при создании простого web-сайта; исходный код проверен rubocop’ом.