«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 №12

Дисциплина: Языки интернет программирования

Название лабораторной работы: Сессии. Выполнение авторизации. Хранение состояния средствами браузера и сервера. Интеграционные тесты.

Вариант 24

Студент гр. ИУ6-32	_____	**Скударнов И. В.**
	(Подпись, дата)	(И.О. Фамилия)
Преподаватель	_____	**Самарёв Р. С.**
	(Подпись, дата)	(И.О. Фамилия)

Москва, 2016

Задание:

Модифицировать код лабораторной работы 8. Релизовать в приложении возможность регистрации и авторизации, запретить доступ до вычислений без авторизации.

Код:

users_controller.rb
```
class UsersController < ApplicationController
 before_action :set_user, only: [:show, :edit, :update, :destroy]

 def index
  @users = User.all
  respond_to do |format|
   format.html
   format.xml { render xml: @users.map { |user| { user: {
                                    id: user.id,
                                    login: user.login,
                                    password: user.password_filtered,
                                    created_at: user.created_at,
                                    updated_at: user.updated_at
                                    } } } }
  end
 end

 def show
  set_user
 end

 def new
  @user = User.new
 end

 def edit
  set_user
 end

 include SessionHelper

 def create
  @user = User.new(user_params)
  if @user.save_new_user
   sign_in @user
   redirect_to root_path
  else
   redirect_to signin_path, notice: 'user' + user_params[:login] + 'exists'
  end
 end

 def update
  set_user
  if @user.update_exist_user(user_params)
   redirect_to user_path, notice: 'user' + @user.login + 'updated'
```

```ruby
    else
      redirect_to user_edit_path, notice: 'user' + user_params[:login] + 'exists'
    end
  end

  def destroy
    set_user
    @user.destroy
    redirect_to root_path
  end

  private

  def set_user
    @user = User.find session[:current_user_id]
  end

  def user_params
    params.require(:user).permit(:login, :password)
  end
end
```

user.rb– модель
```ruby
class User < ApplicationRecord
  def password_filtered
    password.size.times.to_a.map { '*' }.join
  end

  def has_password?(submitted_password)
    password == submitted_password
  end

  def self.authenticate(login, submitted_password)
    user = find_by_login(login)
    return nil  if user.nil?
    return user if user.has_password?(submitted_password)
  end

  def save_new_user
    save unless User.find_by_login login
  end

  def update_exist_user(params)
    update(params) if User.find_by_login(params[:login]).nil? || User.find_by_login(params[:login]) ==
self
  end

  def password
    super
  end

  def save
    super
  end
```

```ruby
  def update(params)
    super(params)
  end
end
```

session_controller.rb

```ruby
class SessionController < ApplicationController
  def new
  end

  def create
    user = User.authenticate params[:session][:login], params[:session][:password]

    if user.nil?
      redirect_to signin_path, notice: 'invalid_login_password'
    else
      sign_in user
      redirect_to root_path
    end
  end

  def destroy
    sign_out
    redirect_to root_path
  end
end
```

new.html.erb – представление контроллера Session

```erb
<h1>Session#new</h1>
<p>Find me in app/views/session/new.html.erb</p>
<div class='container'>
  <div class='headline'><%= 'signin' %></div>
  <div>
    <%= form_for(:session, :url => session_path) do |f| %>
      <div class='field'>
        <%= f.label 'login', class: 'user_input_label', for: 'session_login' %>
        <%= f.text_field :login %>
      </div>
      <div class='field'>
        <%= f.label 'password', class: 'user_input_label', for: 'session_password' %>
        <%= f.password_field :password %>
      </div>
      <div class='field'>
        <%= f.submit 'signin' %>
      </div>
    <% end %>
  </div>

  <div id='notice'><%= notice %></div>
  <div>
    <span id='new_user_span'><%= 'new_user?' %>  </span>
    <%= link_to 'signup_now', signup_path %>
```

```
  </div>
</div>
```

session_helper.rb

```ruby
module SessionHelper
  def sign_in(user)
    session[:current_user_id] = user.id
  end

  def sign_out
    session[:current_user_id] = nil
  end
end
```

application_controller.rb

```ruby
class ApplicationController < ActionController::Base
  protect_from_forgery with: :exception

  include SessionHelper

  before_action :authenticate, except: [:new, :create, :signup, :create_user]

  def current_user
    session[:current_user_id] && User.find_by_id(session[:current_user_id])
  end

  def authenticate
    puts "Current user is #{current_user}, id is #{session[:current_user_id]}"
    unless current_user
      redirect_to signin_path
    end
  end

  before_action :set_locale

  def set_locale
    I18n.locale = browser_locales.first || I18n.default_locale
  end

  def browser_locales
    request.env['HTTP_ACCEPT_LANGUAGE'].split(/;|,/).reject do |x|
      x.start_with?('q')
    end & I18n.available_locales.map(&:to_s)
  end
end
```

routes.rb

```ruby
Rails.application.routes.draw do
  root 'palind#input'
  get 'session/new'

  get 'palind/input'
```

```ruby
  get 'palind/view', to: 'palind#view'

  get 'signup',  to: 'users#new'
  post 'create_user', to: 'users#create'
  get 'user', to: 'users#show'
  post 'user', to: 'users#update'
  get 'user_edit', to: 'users#edit'
  get 'users', to: 'users#index'
  get 'destroy_user', to: 'users#destroy'

  get 'signin',  to: 'session#new'
  get 'signout', to: 'session#destroy'
  post 'session', to: 'session#create'

  # For details on the DSL available within this file, see http://guides.rubyonrails.org/routing.html
end
```

user_flows_test.rb

```ruby
require 'test_helper'

class UserFlowsTest < ActionDispatch::IntegrationTest
  test 'registration and calculation'do

    get signup_path
    assert_response :success

    post create_user_path, params: { user:{ login: 'login', password: 'password' } }
    assert_redirected_to controller: 'palind', action: 'input'

    follow_redirect!
    get '/palind/view', params: { n1: 50 }

    answer = 3
    result = assigns[:result].first
    assert_in_delta answer, result
  end

  test 'registration and walk through pages'do

    get signup_path
    assert_response :success

    post create_user_path, params: { user:{ login: 'login', password: 'password' } }
    assert_redirected_to controller: 'palind', action: 'input'

    follow_redirect!
    get '/palind/view', params: { n1: 50 }
    assert_response :success

    get user_path
    assert_response :success

    get users_path
    assert_response :success
```

```
    get user_edit_path
    assert_response :success

    post user_path, params: { user:{ login: 'login', password: 'password' } }
    assert_redirected_to controller: 'users', action: 'show'

    get destroy_user_path
    assert_redirected_to root_path
  end
end
```

Результаты тестирования:

```
C:\ruby\lr12\palind>rake
Run options: --seed 24800

# Running:

Current user is #<User:0x0000000b1ea900>, id is 980190963
Current user is #<User:0x0000000b24a260>, id is 980190963
Число 11, его квадрат 121
Число 22, его квадрат 484
Число 26, его квадрат 676
Current user is #<User:0x0000000b2888a8>, id is 980190963
Current user is #<User:0x0000000b2c9308>, id is 980190963
Current user is #<User:0x0000000a1cc0f0>, id is 980190963
Current user is #<User:0x0000000a0f8c78>, id is 980190963
Current user is #<User:0x0000000b3f5650>, id is 980190963
.Current user is #<User:0x0000000b495f60>, id is 980190963
Current user is #<User:0x00000009ef1470>, id is 980190963
Число 11, его квадрат 121
Число 22, его квадрат 484
Число 26, его квадрат 676

.

Finished in 16.064156s, 0.1245 runs/s, 0.9338 assertions/s.

2 runs, 15 assertions, 0 failures, 0 errors, 0 skips
```
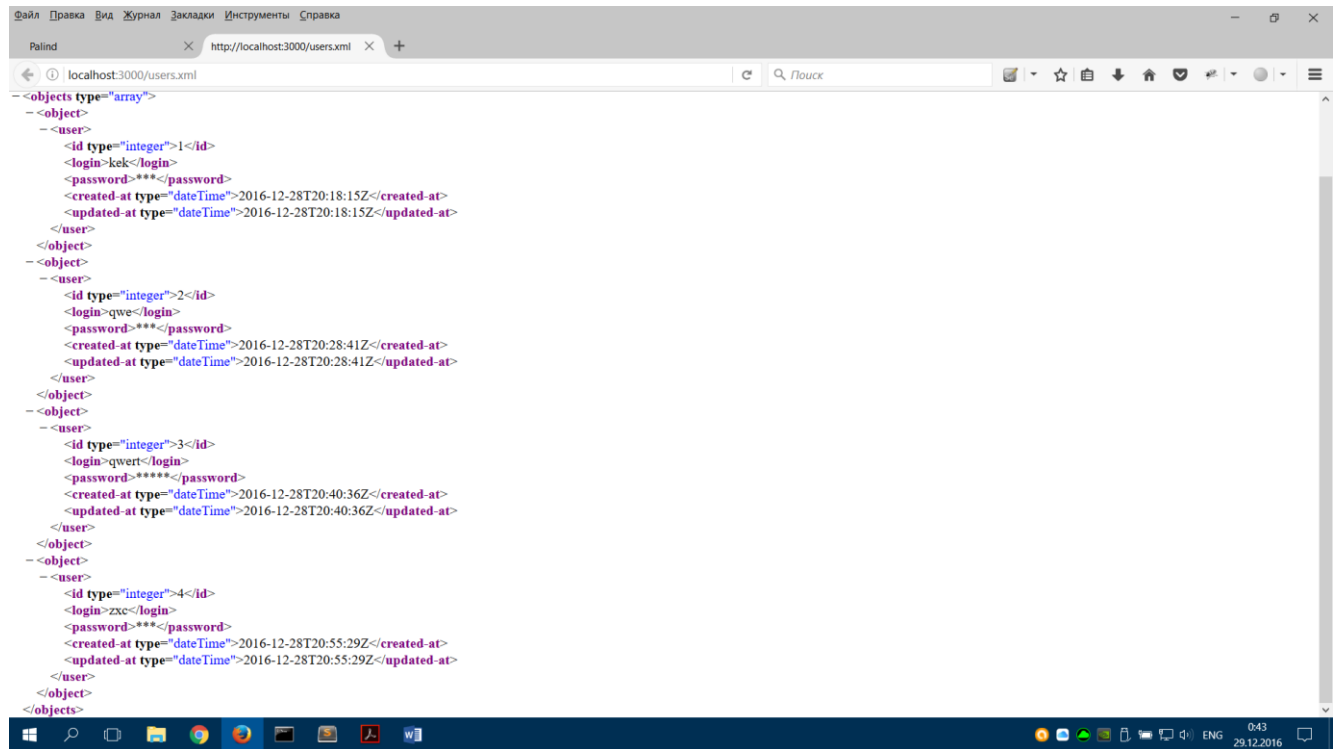
XML-распечатка базы данных:

Вывод: научился работать с сессиями, делать регистрацию и авторизацию пользователей.