ГУАП

КАФЕДРА № 43

ОТЧЕТ   
ЗАЩИЩЕН С ОЦЕНКОЙ

ПРЕПОДАВАТЕЛЬ

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Старший преподаватель |  |  |  | П.А.Степанов |
| должность, уч. степень, звание |  | подпись, дата |  | инициалы, фамилия |

|  |
| --- |
| ОТЧЕТ О ЛАБОРАТОРНОЙ РАБОТЕ №4 |
| «Разработка формы логина» |
| по курсу: Технологии разработки серверных информационных систем |
|  |
|  |

РАБОТУ ВЫПОЛНИЛ

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| СТУДЕНТ ГР. № | 4831 |  | 10.10.2020 |  | К.А.Корнющенков |
|  |  |  | подпись, дата |  | инициалы, фамилия |

Санкт-Петербург 2020

1. **Цель работы**

Разработать форму логина с использованием Spring Security

1. **Вариант задания**

Коллекционирование (нумизматика, филателия и пр).

1. **Код программы**

ControllersConfig

@Configuration  
public class ControllersConfig extends WebMvcConfigurerAdapter {  
  
 @Override  
 public void addViewControllers(ViewControllerRegistry registry) {  
 registry.addViewController("/login").setViewName("login");  
 registry.addViewController("/error").setViewName("error");  
 registry.addViewController("/").setViewName("money");  
 }  
}

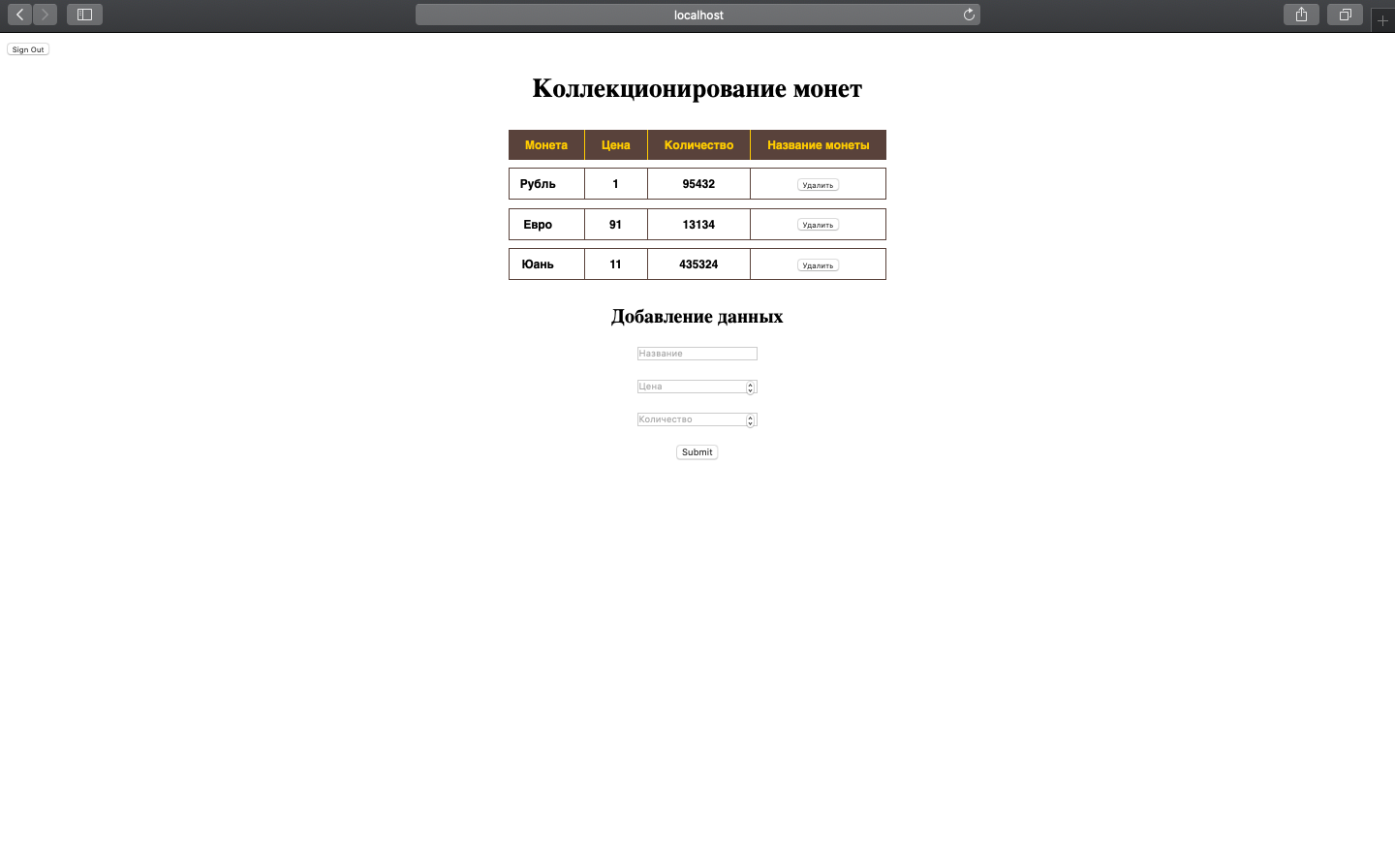
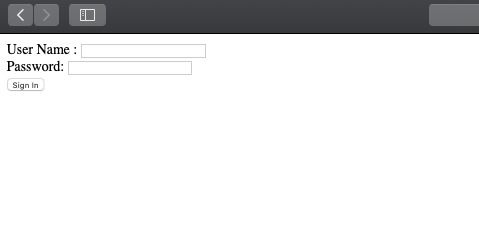
WebSecurityConfig

@Configuration  
public class WebSecurityConfig extends WebSecurityConfigurerAdapter {  
  
 @Autowired  
 private UserDetailsService userDetailsService;  
  
 @Autowired  
 public void configAuthentication(AuthenticationManagerBuilder auth) throws Exception {  
 auth.userDetailsService(userDetailsService).passwordEncoder(new BCryptPasswordEncoder());  
 }  
  
 @Bean  
 @Override  
 public AuthenticationManager authenticationManagerBean() throws Exception {  
 return super.authenticationManagerBean();  
 }  
  
 @Override  
 protected void configure(AuthenticationManagerBuilder auth) throws Exception {  
 PasswordEncoder encoder = PasswordEncoderFactories.*createDelegatingPasswordEncoder*();  
 auth.inMemoryAuthentication()  
 .withUser("kirill")  
 .password(encoder.encode("qwerty"))  
 .roles("ALL");  
 }  
  
 @Override  
 public void configure(WebSecurity web) throws Exception {  
 web.ignoring().antMatchers("/resources/\*\*");  
 }  
  
 @Override  
 protected void configure(HttpSecurity http) throws Exception {  
 http  
 .authorizeRequests()  
 .antMatchers("/money", "/css/\*\*", "/js/\*\*", "/images/\*\*", "/public/rest/\*\*")  
 .permitAll()  
 .antMatchers("/")  
 .authenticated()  
 .and()  
 .exceptionHandling()  
 .accessDeniedPage("/403")  
 .and()  
 .formLogin()  
 .loginPage("/login")  
 .failureUrl("/login?error")  
 .permitAll()  
 .and()  
 .logout()  
 .logoutUrl("/logout")  
 .logoutSuccessUrl("/login?logout")  
 .permitAll()  
// .and()  
// .exceptionHandling()  
// .accessDeniedPage("/forbidden")  
 .and()  
 .csrf()  
 .csrfTokenRepository(CookieCsrfTokenRepository.*withHttpOnlyFalse*());;  
 //.and().csrf().disable();  
 }  
}

Login.html

<!DOCTYPE html>  
<html xmlns="http://www.w3.org/1999/xhtml" xmlns:th="https://www.thymeleaf.org"  
 xmlns:sec="https://www.thymeleaf.org/thymeleaf-extras-springsecurity3">  
<head>  
 <script src="//ajax.googleapis.com/ajax/libs/jquery/3.3.1/jquery.min.js"></script>  
  
 <meta name="\_csrf" content="${\_csrf.token}"/>  
 <meta name="\_csrf\_header" content="${\_csrf.headerName}"/>  
 <title>Spring Security Example </title>  
</head>  
<body>  
 <div th:if="${param.error}">  
 Invalid username and password.  
 </div>  
 <div th:if="${param.logout}">  
 You have been logged out.  
 </div>  
 <form th:action="@{/login}" method="post">  
 <div><label> User Name : <input type="text" name="username"/> </label></div>  
 <div><label> Password: <input type="password" name="password"/> </label></div>  
 <div><input type="submit" value="Sign In"/></div>  
 </form>  
</body>  
</html>

1. **Результаты работы программы**



1. **Вывод**

В ходе выполнения лабораторной работы была разработана форма логина с использованием Spring Security.