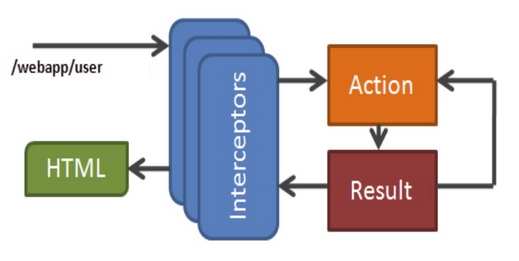
**Руководство Spring Boot Interceptor**

**Будет использовано 3 перехватчика**

1- Что такое Spring Interceptor?

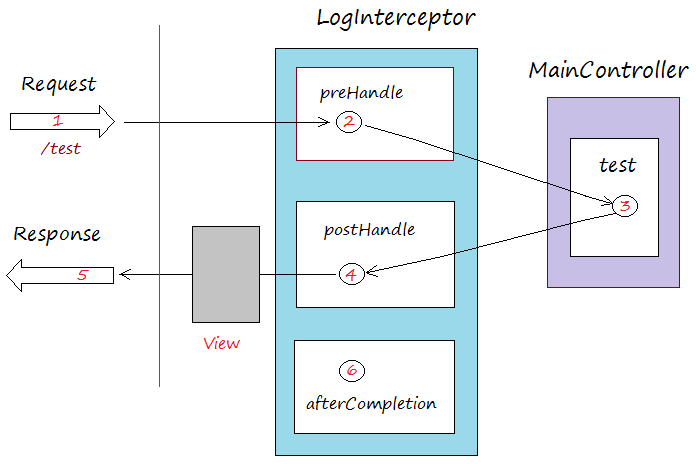
Когда вы приходите в компанию и хотите встретиться с менеджером этой компании, вам нужно пройти через перехватчики (Interceptor), здесь ими могут являться охранник, ресепшн,..  
  
В **Spring**, когда запрос отправляется к controller, перед тем как запрос обрабатывается  **Controller-ом**, он должен перейти через перехватчики  **Interceptor**(0 или более).

**Spring Interceptor** это понятие довольно схожее с  **Servlet Filter** но слабее.  
  
**Spring Interceptor** применяется к запросам отправленным к **Controller**.



Вы можете использовать  **Interceptor** для выполнения таких задач, как запись  **Log**, добавить или обновить конфигурацию перед тем как запрос обрабатывается **Controller-ом**, ...

* *Одно из приложений****Spring Boot MVC****использует****Interceptor****как "Многоязычное веб приложение". Вы можете посмотреть больше про это приложение по этой ссылке:*[*Создание веб-приложения с несколькими языками с помощью Spring Boot*](https://o7planning.org/ru/11691/create-a-multi-language-web-application-with-spring-boot)



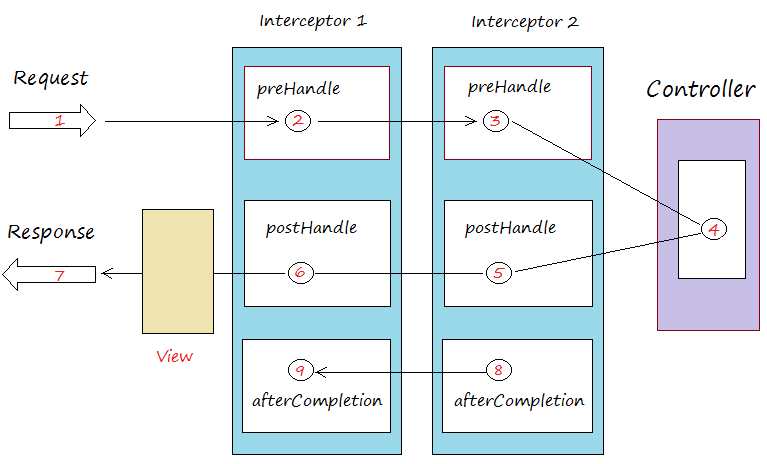
Ваш класс  **Interceptor**должен выполнить интерфейс  *org.springframework.web.servlet.****HandlerInterceptor*** или расшириться из класс  org.springframework.web.servlet.handler**.HandlerInterceptorAdapter**.

Вам нужно выполнить 3 абстрактных метода:

|  |  |
| --- | --- |
| 1  2  3  4  5  6  7  8  9  10  11  12  13  14  15 | public boolean **preHandle**(HttpServletRequest request,                           HttpServletResponse response,                           Object handler)      public void **postHandle**(HttpServletRequest request,                         HttpServletResponse response,                         Object handler,                         ModelAndView modelAndView)      public void **afterCompletion**(HttpServletRequest request,                              HttpServletResponse response,                              Object handler,                              Exception ex) |

*Примечание: метод****preHandle****возвращает****true****или****false****. Если возвращает****true****значит запрос продолжает идти до его цели (т.е. к****Controller-у****) .*

Каждый запрос может пройти через многие  **Interceptor**. Изображение ниже иллюстрирует это.



2- Создать проект Spring Boot

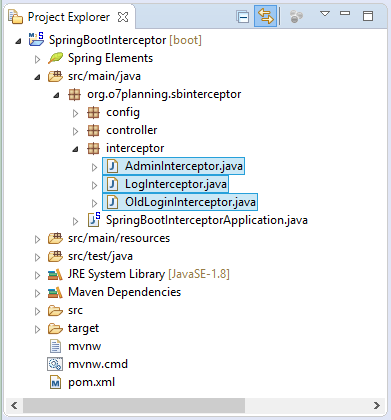
**pom.xml**

|  |  |
| --- | --- |
| 1  2  3  4  5  6  7  8  9  10  11  12  13  14  15  16  17  18  19  20  21  22  23  24  25  26  27  28  29  30  31  32  33  34  35  36  37  38  39  40  41  42  43  44  45  46  47  48  49  50  51  52  53  54 | <?xml version="1.0" encoding="UTF-8"?>  <project xmlns="<http://maven.apache.org/POM/4.0.0>" xmlns:xsi="<http://www.w3.org/2001/XMLSchema-instance>"      xsi:schemaLocation="<http://maven.apache.org/POM/4.0.0> <http://maven.apache.org/xsd/maven-4.0.0.xsd>">      <modelVersion>4.0.0</modelVersion>        <groupId>org.o7planning</groupId>      <artifactId>SpringBootInterceptor</artifactId>      <version>0.0.1-SNAPSHOT</version>      <packaging>jar</packaging>        <name>SpringBootInterceptor</name>      <description>Spring Boot + Interceptor</description>        <parent>          <groupId>org.springframework.boot</groupId>          <artifactId>spring-boot-starter-parent</artifactId>          <version>2.0.0.RELEASE</version>          <relativePath/> <!-- lookup parent from repository -->      </parent>        <properties>          <project.build.sourceEncoding>UTF-8</project.build.sourceEncoding>          <project.reporting.outputEncoding>UTF-8</project.reporting.outputEncoding>          <java.version>1.8</java.version>      </properties>        <dependencies>          <dependency>              <groupId>org.springframework.boot</groupId>              <artifactId>spring-boot-starter-thymeleaf</artifactId>          </dependency>          <dependency>              <groupId>org.springframework.boot</groupId>              <artifactId>spring-boot-starter-web</artifactId>          </dependency>            <dependency>              <groupId>org.springframework.boot</groupId>              <artifactId>spring-boot-starter-test</artifactId>              <scope>test</scope>          </dependency>      </dependencies>        <build>          <plugins>              <plugin>                  <groupId>org.springframework.boot</groupId>                  <artifactId>spring-boot-maven-plugin</artifactId>              </plugin>          </plugins>      </build>      </project> |

**SpringBootInterceptorApplication.java - main**

|  |  |
| --- | --- |
| 1  2  3  4  5  6  7  8  9  10  11  12  13 | package org.o7planning.sbinterceptor;    import org.springframework.boot.SpringApplication;  import org.springframework.boot.autoconfigure.SpringBootApplication;    @SpringBootApplication  public class SpringBootInterceptorApplication {        public static void main(String[] args) {          SpringApplication.run(SpringBootInterceptorApplication.class, args);      }    } |

3- Interceptor classes

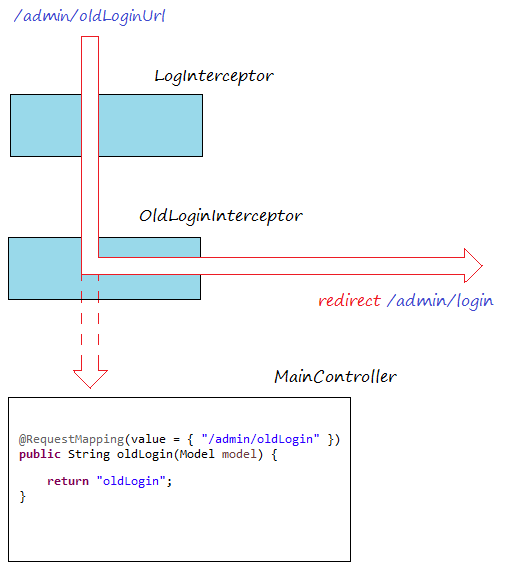


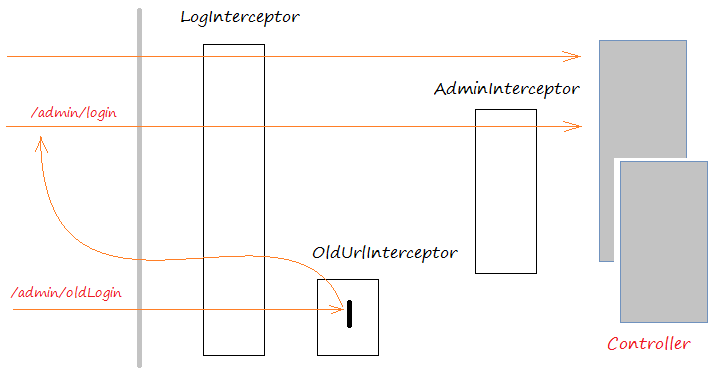
**LogInterceptor** применяется для всех запросов направляющихся к **Controller**. (Смотрите конфигурацию в  **WebMvcConfig**).

**LogInterceptor.java**

|  |  |
| --- | --- |
| 1  2  3  4  5  6  7  8  9  10  11  12  13  14  15  16  17  18  19  20  21  22  23  24  25  26  27  28  29  30  31  32  33  34  35  36  37  38  39  40  41  42  43  44  45  46  47  48 | package org.o7planning.sbinterceptor.interceptor;    import javax.servlet.http.HttpServletRequest;  import javax.servlet.http.HttpServletResponse;    import org.springframework.web.servlet.ModelAndView;  import org.springframework.web.servlet.handler.HandlerInterceptorAdapter;    public class LogInterceptor extends HandlerInterceptorAdapter {        @Override      public boolean **preHandle**(HttpServletRequest request, HttpServletResponse response, Object handler) throws Exception {          long startTime = System.currentTimeMillis();          System.out.println("\n-------- LogInterception.preHandle --- ");          System.out.println("Request URL: " + request.getRequestURL());          System.out.println("Start Time: " + System.currentTimeMillis());            request.setAttribute("startTime", startTime);            return true; //обязательная строка для этого метода      }        @Override      public void **postHandle**(HttpServletRequest request, HttpServletResponse response, Object handler, ModelAndView modelAndView) throws Exception {            System.out.println("\n-------- LogInterception.postHandle --- ");          System.out.println("Request URL: " + request.getRequestURL());            // Вы можете добавить атрибуты в modelAndView и использовать их на  //странице просмотра      }        @Override      public void **afterCompletion**(HttpServletRequest request, HttpServletResponse response, Object handler, Exception ex) throws Exception {          System.out.println("\n-------- LogInterception.afterCompletion --- ");            long startTime = (Long) request.getAttribute("startTime");          long endTime = System.currentTimeMillis();          System.out.println("Request URL: " + request.getRequestURL());          System.out.println("End Time: " + endTime);            System.out.println("Time Taken: " + (endTime - startTime));      }    } |

**OldLoginInterceptor** это перехватчик (interceptor), если пользователь вводит ссылку  ***/admin/oldLogin*** он переходит на новую ссылку это ***/admin/login***.





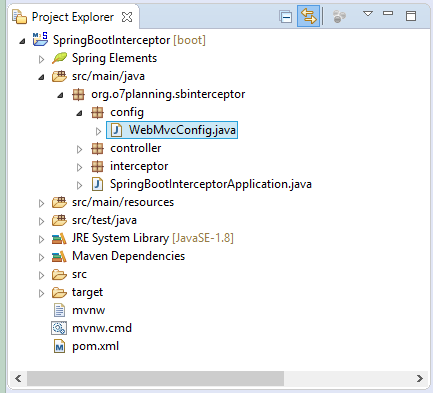
**OldLoginInterceptor.java**

|  |  |
| --- | --- |
| 1  2  3  4  5  6  7  8  9  10  11  12  13  14  15  16  17  18  19  20  21  22  23  24  25  26  27  28  29  30  31  32  33  34  35  36  37  38  39 | package org.o7planning.sbinterceptor.interceptor;    import javax.servlet.http.HttpServletRequest;  import javax.servlet.http.HttpServletResponse;    import org.springframework.web.servlet.ModelAndView;  import org.springframework.web.servlet.handler.HandlerInterceptorAdapter;    public class OldLoginInterceptor extends HandlerInterceptorAdapter {        @Override      public boolean preHandle(HttpServletRequest request, HttpServletResponse response, Object handler) throws Exception {            System.out.println("\n-------- OldLoginInterceptor.preHandle --- ");          System.out.println("Request URL: " + request.getRequestURL());          System.out.println("Sorry! This URL is no longer used, Redirect to /admin/login");            response.sendRedirect(request.getContextPath() + "/admin/login");          return false;      }        @Override      public void postHandle(HttpServletRequest request, HttpServletResponse response, Object handler, ModelAndView modelAndView) throws Exception {    **// This code will never be run.**          System.out.println("\n-------- OldLoginInterceptor.postHandle --- ");      }        @Override      public void afterCompletion(HttpServletRequest request, HttpServletResponse response, Object handler, Exception ex) throws Exception {    **// This code will never be run**.          System.out.println("\n-------- QueryStringInterceptor.afterCompletion --- ");      }    } |

**AdminInterceptor.java**

|  |  |
| --- | --- |
| 1  2  3  4  5  6  7  8  9  10  11  12  13  14  15  16  17  18  19  20  21  22  23  24  25  26  27  28  29  30  31  32  33 | package org.o7planning.sbinterceptor.interceptor;    import javax.servlet.http.HttpServletRequest;  import javax.servlet.http.HttpServletResponse;    import org.springframework.web.servlet.ModelAndView;  import org.springframework.web.servlet.handler.HandlerInterceptorAdapter;    public class AdminInterceptor extends HandlerInterceptorAdapter {        @Override      public boolean preHandle(HttpServletRequest request, HttpServletResponse response, Object handler)throws Exception {            System.out.println("\n-------- AdminInterceptor.preHandle --- ");          return true;      }        @Override      public void postHandle(HttpServletRequest request, HttpServletResponse response, Object handler, ModelAndView modelAndView) throws Exception {            System.out.println("\n-------- AdminInterceptor.postHandle --- ");      }        @Override      public void afterCompletion(HttpServletRequest request, HttpServletResponse response, Object handler, Exception ex) throws Exception {            System.out.println("\n-------- AdminInterceptor.afterCompletion --- ");      }    } |

4- Конфигурировать Interceptor



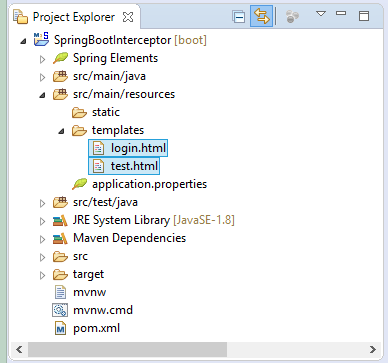
**WebMvcConfig.java**

|  |  |
| --- | --- |
| 1  2  3  4  5  6  7  8  9  10  11  12  13  14  15  16  17  18  19  20  21  22  23  24  25  26  27  28 | package org.o7planning.sbinterceptor.config;    import org.o7planning.sbinterceptor.interceptor.AdminInterceptor;  import org.o7planning.sbinterceptor.interceptor.LogInterceptor;  import org.o7planning.sbinterceptor.interceptor.OldLoginInterceptor;  import org.springframework.context.annotation.Configuration;  import org.springframework.web.servlet.config.annotation.InterceptorRegistry;  import org.springframework.web.servlet.config.annotation.WebMvcConfigurerAdapter;    @Configuration  public class WebMvcConfig implements WebMvcConfigurer {       @Override     public void addInterceptors(InterceptorRegistry registry) {        // LogInterceptor применяется ко всем URL-адресам.        registry.addInterceptor(new LogInterceptor());          // Старый URL для входа, больше не использовать.        // Использует OldURLInterceptor для перенаправления на новый URL.        registry.addInterceptor(new OldLoginInterceptor())              .addPathPatterns("/admin/oldLogin");          // Этот перехватчик добавляет шаблоны URL, к которым должен применяться зарегистрированный перехватчик (у нас URL-адрес /admin/\*)        // и исключает шаблоны URL, к которым не должен применяться зарегистрированный перехватчик (у нас URL-адрес /admin/oldLogin)        registry.addInterceptor(new AdminInterceptor())              .addPathPatterns("/admin/\*")//              .excludePathPatterns("/admin/oldLogin");     }    } |

5- Controllers & Views

**Controller.java**

|  |  |
| --- | --- |
| 1  2  3  4  5  6  7  8  9  10  11  12  13  14  15  16  17  18  19  20  21  22  23  24  25  26  27  28  29  30  31  32  33  34 | package org.o7planning.sbinterceptor.controller;    import org.springframework.stereotype.Controller;  import org.springframework.ui.Model;  import org.springframework.web.bind.annotation.RequestMapping;    @Controller  public class Controller {        @RequestMapping(value = { "/", "/test" })      public String test(Model model) {            System.out.println("\n-------- Controller.test --- ");          System.out.println(" \*\* You are in Controller \*\* ");            return "test";      }        // Этот путь больше не используется.      // Он будет перенаправлен OldLoginInterceptor      @Deprecated      @RequestMapping(value = { "/admin/oldLogin" })      public String oldLogin(Model model) {          // Следующий код никогда не запустится.          return "oldLogin";      }        @RequestMapping(value = { "/admin/login" })      public String login(Model model) {            System.out.println("\n-------- Controller.login --- ");          System.out.println(" \*\* You are in Controller \*\* ");          return "login";      }   } |



**test.html**

|  |  |
| --- | --- |
| 1  2  3  4  5  6  7  8  9  10  11  12  13  14  15  16  17  18  19  20  21  22  23  24 | <!DOCTYPE HTML>  <html xmlns:th="[http://www.thymeleaf.org](http://www.thymeleaf.org/)">       <head>        <meta charset="UTF-8" />        <title>Spring Boot Mvc Interceptor example</title>     </head>       <body>        <div style="border: 1px solid #ccc;padding: 5px;margin-bottom:10px;">           <a th:href="@{/}">Home</a>           &nbsp;&nbsp; | &nbsp;&nbsp;           <a th:href="@{/admin/oldLogin}">/admin/oldLogin (OLD URL)</a>        </div>          <h3>Spring Boot Mvc Interceptor</h3>          <span style="color:blue;">Testing LogInterceptor</span>        <br/><br/>          See Log in Console..       </body>  </html> |

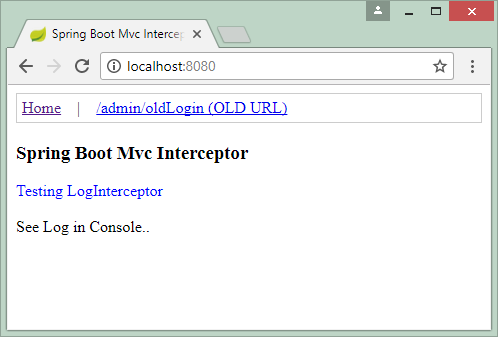
**login.html**

|  |  |
| --- | --- |
| 1  2  3  4  5  6  7  8  9  10  11  12  13  14  15  16  17  18  19  20  21  22  23 | <!DOCTYPE HTML>  <html xmlns:th="[http://www.thymeleaf.org](http://www.thymeleaf.org/)">     <head>        <meta charset="UTF-8" />        <title>Spring Boot Mvc Interceptor example</title>     </head>     <body>          <div style="border: 1px solid #ccc;padding: 5px;margin-bottom:10px;">           <a th:href="@{/}">Home</a>           &nbsp;&nbsp; | &nbsp;&nbsp;           <a th:href="@{/admin/oldLogin}">/admin/oldLogin (OLD URL)</a>        </div>          <h3>This is Login Page</h3>          <span style="color:blue">Testing OldLoginInterceptor &amp; AdminInterceptor</span>        <br/><br/>        See more info in the Console.       </body>    </html> |

## 6- Запуск приложения

1. Тест случая, когда пользователь заходит на страницу, **LogInterceptor** записывает связанную информацию (Адрес страницы, время доступа), и считает время, за которое  **Web Server** предоставил услугу для запроса.

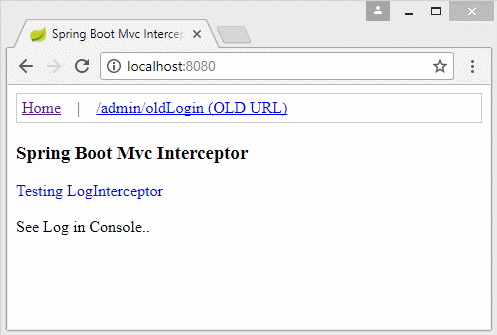
* [http://localhost:8080](http://localhost:8080/)



|  |  |
| --- | --- |
| 1  2  3  4  5  6  7  8  9  10  11  12  13  14  15  16 | -------- LogInterception.preHandle ---  Request URL: <http://localhost:8080/>  Start Time: 1512231713663    -------- Controller.test ---   \*\* You are in Controller \*\*    -------- LogInterception.postHandle ---  Request URL: <http://localhost:8080/>    -------- LogInterception.afterCompletion ---  Request URL: <http://localhost:8080/>  End Time: 1512231713665  Time Taken: 2 |

2. Тест случая когда пользователь заходит на старую страницу логина (Больше не используется), **OldLoginInterceptor** перенаправит запрос на новую страницу логина.

* <http://localhost:8080/admin/oldLogin>



|  |  |
| --- | --- |
| 1  2  3  4  5  6  7  8  9  10  11  12  13  14  15  16  17  18  19  20  21  22  23  24  25  26  27  28  29  30  31  32  33  34  35 | -------- LogInterception.preHandle ---  Request URL: <http://localhost:8080/admin/oldLogin>  Start Time: 1512231812219    -------- OldLoginInterceptor.preHandle ---  Request URL: <http://localhost:8080/admin/oldLogin>  Sorry! This URL is no longer used, Redirect to /admin/login    -------- LogInterception.afterCompletion ---  Request URL: <http://localhost:8080/admin/oldLogin>  End Time: 1512231812219  Time Taken: 1    -------- LogInterception.preHandle ---  Request URL: <http://localhost:8080/admin/login>  Start Time: 1512231812222    -------- AdminInterceptor.preHandle ---    -------- Controller.login ---   \*\* You are in Controller \*\*    -------- AdminInterceptor.postHandle ---    -------- LogInterception.postHandle ---  Request URL: <http://localhost:8080/admin/login>    -------- AdminInterceptor.afterCompletion ---    -------- LogInterception.afterCompletion ---  Request URL: <http://localhost:8080/admin/login>  End Time: 1512231812227  Time Taken: 5 |