

## مثال Page Redirection در Spring MVC

در این مثال چگونگی نوشتن یک برنامه ی تحت وب ساده نشان داده شده است که از طریق آن می توان جهت انتقال یک درخواست HTTP به صفحه ی دیگر از قابلیت `redirect` استفاده کرد. برای شروع Eclipse IDE را آماده کنید و جهت توسعه ی این برنامه ی وب مبتنی بر فرم پویا با استفاده از Spring Web Framework مراحل زیر را دنبال کنید.

مرحله	توضیحات
1	پروژه ای به نام <code>HelloWeb</code> در بسته ی <code>com.tutorialspoint</code> همان طور که در بخش <code>Spring MVC - Hello World</code> توضیح داده شده است، ایجاد کنید.
2	در بسته ی <code>com.tutorialspoint</code> یک <code>WebController</code> کلاس جاوا ایجاد کنید.
3	در زیرپوشه ی <code>jsp</code> فایل های ویوی <code>index.jsp</code> و <code>final.jsp</code> را ایجاد کنید.
4	مرحله ی آخر ایجاد محتوای فایل های محتوای پیکربندی و منبع و اکسپورت کردن برنامه به صورت زیر است.

### WebController.java

```
package com.tutorialspoint;

import org.springframework.stereotype.Controller;
import org.springframework.web.bind.annotation.RequestMapping;
import org.springframework.web.bind.annotation.RequestMethod;

@Controller
public class WebController {

    @RequestMapping(value = "/index", method = RequestMethod.GET)
```

```

public String index() {

    return "index";

}

@RequestMapping(value = "/redirect", method = RequestMethod.GET)

public String redirect() {

    return "redirect:finalPage";

}

@RequestMapping(value = "/finalPage", method = RequestMethod.GET)

public String finalPage() {

    return "final";

}
}

```

در ادامه می توانید محتوای فایل ویوی `index.jsp` را مشاهده کنید. این صفحه قرار است صفحه ی `landing` باشد. به این صورت که درخواستی را به متد سرویس `access-redirect` ارسال می کند که این درخواست به متد سرویس دیگری `redirect` می شود و در نهایت صفحه ی `final.jsp` نمایش داده می شود.

`index.jsp`

```

<%@taglib uri = "http://www.springframework.org/tags/form" prefix = "form"%>

<html>

<head>

<title>Spring Page Redirection</title>

</head>

```

```
<body>

  <h2>Spring Page Redirection</h2>

  <p>Click below button to redirect the result to new page</p>

  <form:form method = "GET" action = "/HelloWeb/redirect">

    <table>

      <tr>

        <td>

          <input type = "submit" value = "Redirect Page"/>

        </td>

      </tr>

    </table>

  </form:form>

</body>

</html>
```

### final.jsp

```
<%@taglib uri = "http://www.springframework.org/tags/form" prefix = "form"%>

<html>

  <head>

    <title>Spring Page Redirection</title>

  </head>

  <body>

    <h2>Redirected Page</h2>

  </body>
```

</html>

بعد از آن که کار فایل های پیکربندی و منبع تمام شد، برنامه ی خود را اکسپورت کنید. بر روی برنامه ی خود کلیک راست کنید، از گزینه ی Export → WAR File استفاده کنید و فایل HelloWeb.war را داخل پوشه ی webapps متعلق به Tomcat ذخیره کنید.

حالا سرور Tomcat را اجرا کنید و مطمئن شوید که از طریق پوشه ی webapps و با استفاده از یک مرورگر استاندارد می توانید به دیگر صفحات وب دسترسی پیدا کنید. حالا در صورت وارد کردن آدرس http://localhost:8080/HelloWeb/index و در صورت نبود مشکل در برنامه ی Spring Web صفحه ی زیر نمایش داده می شود.



حالا بر روی دکمه ی "Redirect Page" کلیک کنید تا فرم مورد نظرتان ارائه شود و بتوانید به صفحه ی Redirect شده ی نهایی بروید. اگر مشکلی وجود نداشته باشد، صفحه ی زیر نمایش داده می شود.

