**Handler Mapping**

Define how web request (URL) maps to the Controller handlers.  
  
**BeanNameUrlHandlerMapping**   
Maps the requested URL to the name of the cocntroller.  
**ControllerClassNameHandlerMapping**  
Uses convention to map the requested URL to Controller.  
**SimpleUrlHandlerMapping**   
Allow developer to specify the mapping of URL patterns and handler mappings explicitly.

In Spring MVC, BeanNameUrlHandlerMapping is the default handler mapping mechanism, which maps URL requests to the name of the beans. For example,

<beans ...="">

<bean class="org.springframework.web.servlet.handler.BeanNameUrlHandlerMapping">

<bean class="com.sb.web.controller.WelcomeController" name="/welcome.htm">

<bean class="com.sb.web.controller.StreetNameController" name="/streetName.htm">

<bean class="com.sb.web.controller.ProcessController" name="/process\*.htm">

</bean></bean></bean></bean></beans>

In above example, If URI pattern  
/welcome.htm is requested, DispatcherServlet will forward the request to the “WelcomeController“.  
/streetName.htm is requested, DispatcherServlet will forward the request to the “StreetNameController“.  
/processCreditCard.htm or /process{any thing}.htm is requested, DispatcherServlet will forward the request to the “ProcessController“.

In Spring MVC, ControllerClassNameHandlerMapping use convention to map requested URL to Controller (convention over configuration). It takes the Class name, remove the ‘Controller’ suffix if exists and return the remaining text, lower-cased and with a leading “/”.  
To enable the ControllerClassNameHandlerMapping, declared it in the bean configuration file, and now the controller’s bean’s name is no longer required.

<beans ...="">

<bean class="org.springframework.web.servlet.mvc.support.ControllerClassNameHandlerMapping">

<bean class="com.sb.web.controller.WelcomeController">

<bean class="com.sb.web.controller.HelloGuestController">

</bean></bean></bean></beans>

Now, Spring MVC is mapping the requested URL by following conventions :

WelcomeController -> /welcome\*

HelloGuestController -> /helloguest\*

/welcome.htm –> WelcomeController.

/welcomeHome.htm –> WelcomeController.

/helloguest.htm –> HelloGuestController.

/helloguest12345.htm –> HelloGuestController.

/helloGuest.htm, failed to map /helloguest\*, the “g” case is not match.

In Spring MVC application, the SimpleUrlHandlerMapping is the most flexible handler mapping class, which allow developer to specify the mapping of URL pattern and handlers explicitly.  
The SimpleUrlHandlerMapping can be declared in two ways.  
1. Method 1 – prop key  
The property keys are the URL patterns while the property values are the handler IDs or names.

<beans ...="">

<bean class="org.springframework.web.servlet.handler.SimpleUrlHandlerMapping">

<property name="mappings">

<props>

<prop key="/welcome.htm">welcomeController</prop>

<prop key="/\*/welcome.htm">welcomeController</prop>

<prop key="/helloGuest.htm">helloGuestController</prop>

</props>

</property>

</bean>

<bean class="com.sb.web.controller.WelcomeController" id="welcomeController">

<bean class="com.sb.web.controller.HelloGuestController" id="helloGuestController">

</bean></bean></beans>

2. Method 1 – value

The left side are the URL patterns while the right side are the handler IDs or names, separate by a equal symbol “=”.

<beans ...="">

<bean class="org.springframework.web.servlet.handler.SimpleUrlHandlerMapping">

<property name="mappings">

<value>

/welcome.htm=welcomeController

/\*/welcome.htm=welcomeController

/helloGuest.htm=helloGuestController

</value>

</property>

</bean>

<bean class="com.sb.web.controller.WelcomeController" id="welcomeController">

<bean class="com.sb.web.controller.HelloGuestController" id="helloGuestController">

</bean></bean></beans>