[Bài đọc] Interface ViewResolver

Controller do người dùng định nghĩa trả về tên của view và các thông tin chi tiết khác tới Front Controller. Tên của view được trả về có kiểu là String, nên cần xử lý tiếp bởi ViewResolver. Dưới đây là một vài ViewResolvers có thể sử dụng để trả lại view.

**XmlViewResolver**: Giúp xem file viết bằng XML. Nó được cấu hình mặc định trong WEB-INF/views.xml. File này chứa các thẻ <bean> của view có cùng định dạng DTD(Document Type Declaration) tương tự như các thẻ <bean> trong cấu hình của Spring. Cấu hình có thể được viết như sau:

<bean id="myHome" class="org.springframework.web.servlet.view.JstlView">  
 <property name="url" value="WEB-INF/jsp/home.jsp"/>  
<bean>

Với tên view là "myHome" thì file jsp tương ứng là "WEB-INF/jsp/myHome.jsp". Một bean cũng có thể có chuyển đến view bằng cách thực hiện ánh xạ tới bean khác. Cụ thể như dưới:

<bean id="logout" class="org.springframework.web.servlet.view.RenderView">  
 <property name="url" value="myHome"/>  
<bean>

Bean "logout" không ánh xạ tới bất kì file jsp nào. Tuy nhiên bean này thông qua bean myHome để lấy cùng file jsp.

**UrlBasedViewResolver**: Giúp ánh xạ trực tiếp URL tới tên logic của view. UrlBasedViewResolver được sử dụng nhiều hơn trong việc chỉ ra nguồn của view. Nó sử dụng một tiền tố và một hậu tố như là các thuộc tính trong việc lấy tên và vị trí của view. Để sử dụng URLBasedViewResolver cần cấu hình như dưới:

<bean id="viewResolver" class="org.springframework.web.servlet.view.UrlBasedViewResolver">  
 <property name="viewClass" value="org.springframework.web.servlet.view.JstlView"/>  
 <property name="prefix" value="WEB-INF/jsp/"/>  
 <property name="suffix" value=".jsp"/>  
<bean>

JstlView được sử dụng để tạo ra các trang hiển thị. Tên trang và vị trí của trang là: prefix+ view\_name\_from\_controller+suffix.

**InternalResourceViewResolver**: Là lớp con của lớp UrlBasedViewResolver dùng để giải quyết các vấn đề về tài nguyên bên trong. InternalResourceViewResolver sử dụng các thuộc tính như tiền tố, hậu tố giống như lớp cha của nó.AlwaysInclude, ExposeContextBeansAsAttributes, and ExposedContextBeanNames là các thuộc tính mở rộng của lớp này và có lợi thế là được sử dụng nhiều hơn so với lớp cha của nó. Cách thức cấu hình dưới đây cho InternalResourceViewResolver tương tự như cấu hình của UrlBasedViewResolver  ở trên.

<bean id="viewResolver" class="""org.springframework.web.servlet.view.InternalResourceViewResolver">  
 <property name="viewClass" value="org.springframework.web.servlet.view.JstlView"/>  
 <property name="prefix" value="WEB-INF/jsp/"/>  
 <property name="suffix" value=".jsp"/>  
<bean>

Không như trước, nó có thể xác minh tồn tại duy nhất của trang khi được thêm vào

**ResourceBundleViewResolver**: Được định nghĩa từ ResourceBundle được định rõ trong file cấu hình. File cấu hình mặc định sử dụng views.properties. Cụ thể như dưới

<bean id="viewResolver" class="org.springframework.web.servlet.view.ResourceViewResolver">  
 <property name="base" value="web\_view"/>  
</bean>

 Cấu hình view.properties sẽ xác định chi tiết class view sẽ được sử dụng để ánh xạ URL tới view thực. Cụ thể như sau:

home.(class)= org.springframework.wev.servlet.view.JstlView

 Dòng lệnh sau xánh xạ view có tên là homepage:

homepage.url= /WEB-INF/page/welcome.jsp

**TilesViewResolver**: Framework Tiles sử dụng để định nghĩa các template để tạo ra layout cho các trang. Các template đã định nghĩa sẽ được sử dụng lại để đảm bảo tính đồng nhất của ứng dụng. Các trang được định nghĩa thành các file có định dạng .def như tiêu đề (tile), header, footer và menu. Các file này sẽ được ghép lại với nhau thành một trang khi chạy. Tên logic được trả về bởi controller khớp với các tile của template sẽ được trả về bởi view resolver.

Ngoài các view resolver trên, Spring cũng các view resolver khác như :FreeMarkerViewResolver, TileViewResolver, VelocityLayoutViewResolver, VelocityViewResolver và XsltViewResolver