

Spring MVC ile View Resolve İşlemi



View Resolve İşlemi

- DispatcherServlet “**viewResolver**” yardımı ile ModelAndView'da belirtilen “**logical view name**”i resolve eder
- Bu **genelde bir JSP sayfası** olur
- JSP sayfası olduğu vakit **RequestDispatcher** vasıtası ile web isteği logical view name ile belirtilen **JSP sayfasına yönlendirilir**
- JSP sayfası içerisinde de **model verisi kullanılarak** ekran içeriği görüntülenir

View Resolver

- **Logical view name** ile **View nesnesi** arasında eşleştirme yapar
- Değişik **ViewResolver** implemantasyonları vardır
- Her bir ViewResolver **belirli tipte bir View** nesnesi döner
- **View render** işlemi view nesnesi üzerinden gerçekleşir

ViewResolver Zinciri

- Spring Container içerisinde birden fazla **ViewResolver** bean'i tanımlanabilir
- View nesnesi tespit edilirken **order** attribute'larına göre sıra ile bu **ViewResolver** nesnelere danışılır
- Eğer ViewResolver, **View** nesnesi dönerse o View nesnesi ile **physical view** render edilir
- Aksi takdirde **null** dönülür ve zincirdeki bir sonraki bean'a danışılması sağlanır

InternalResource ViewResolver

- DispatcherServlet'in **built-in** viewResolver nesnesidir
- JSP gibi web uygulamasının **lokal path'indeki view'ları** render etmek için kullanılır
- View name ile belirtilen **jsp/html sayfasına RequestDispatcher ile forward** yapar
- View name'in başına “**redirect:**” **prefix'i** eklenerek redirect yapması da sağlanabilir

InternalResource ViewResolver

```
<bean id="viewResolver"  
class="org.springframework.web.servlet.view.Internal  
ResourceViewResolver">
```

```
  <property name="prefix" value="/WEB-INF/jsp/" />  
  <property name="suffix" value=".jsp" />
```

```
</bean>
```



İstenirse bean olarak override edilebilir
prefix ve suffix attribute'ları ile JSP sayfalarının lokasyonları özelleştirilebilir

InternalResource ViewResolver

- View nesnesine karşılık gelen bir sayfanın (HTML veya JSP) dosya sisteminde mevcut olup olmadığı işlem **RequestDispatcher'a delege** edilmeden anlaşılamaz
- Bu nedenle ViewResolver zincirinde **sıralamada en sonda** yer almalıdır
- Classpath'de **JSTL** kütüphanesi varsa logical view'dan **JstlView** oluşturur ve döner
- Aksi takdirde **InternalResourceView** nesnesi döner

İletişim

- **Harezmi** Bilişim Çözümleri
- Kurumsal Java Eğitimleri
- <http://www.java-egitimleri.com>
- info@java-egitimleri.com

