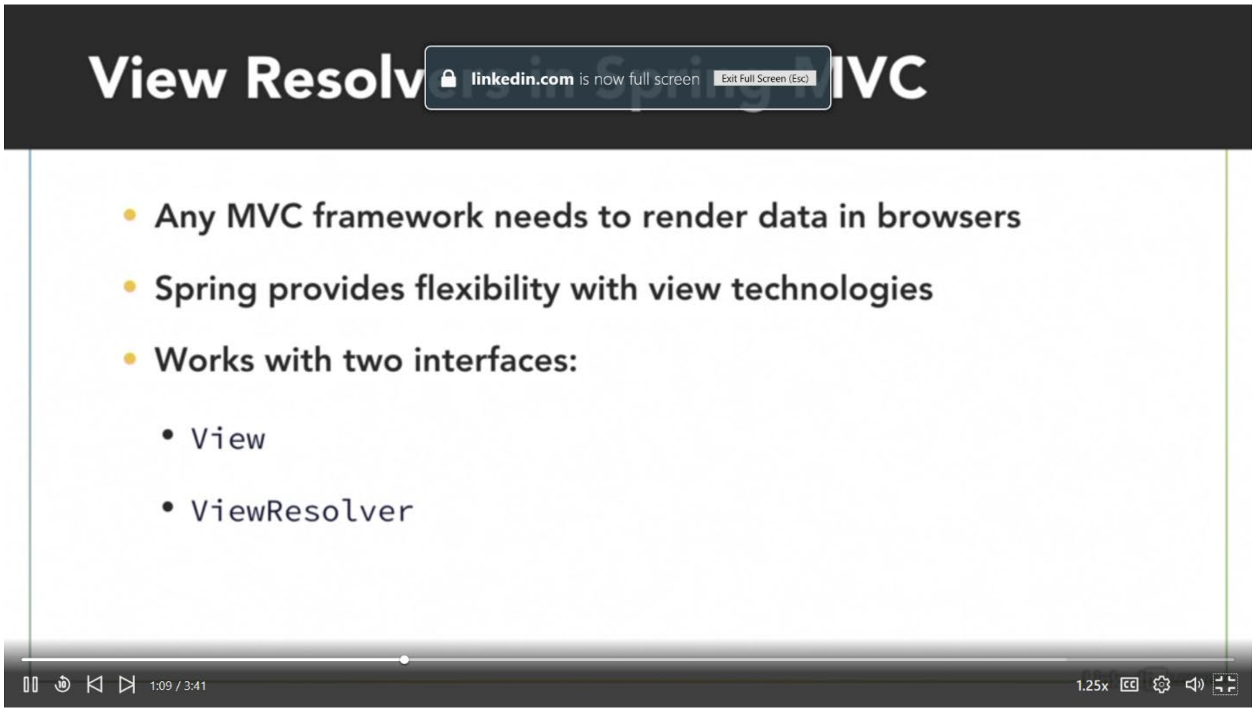
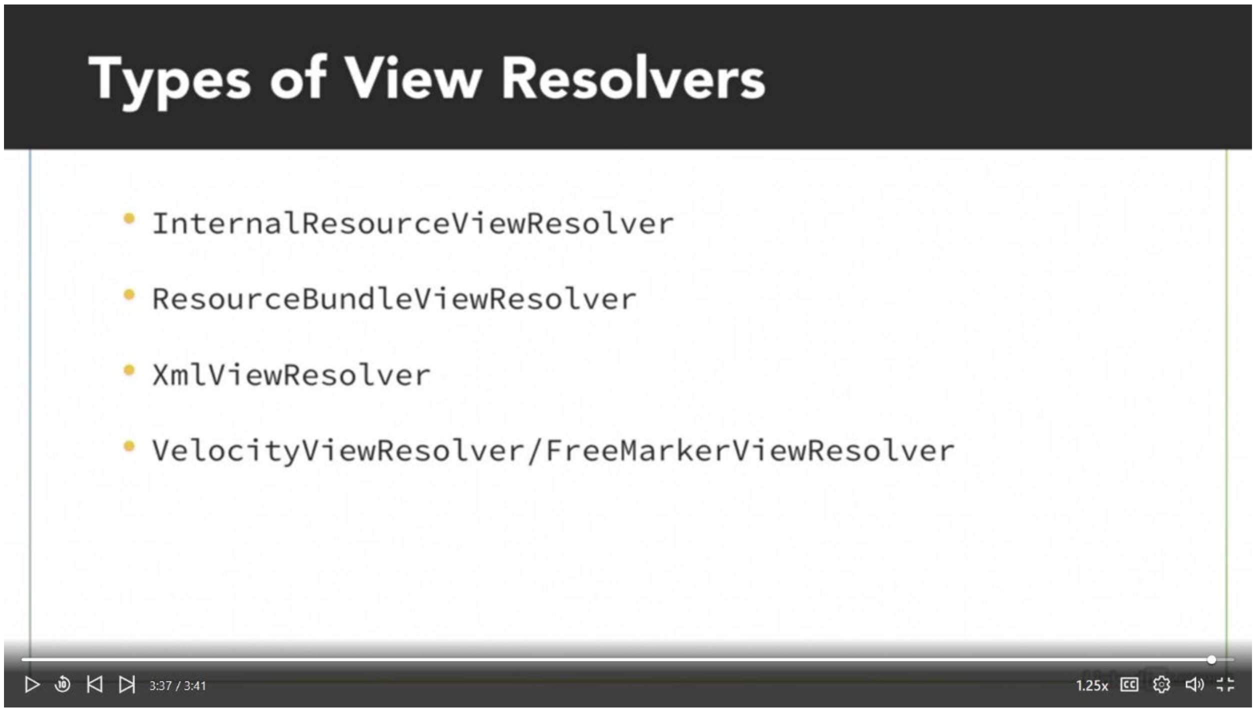
View Resolver in Spring MCV



View Resolvers in Spring MVC

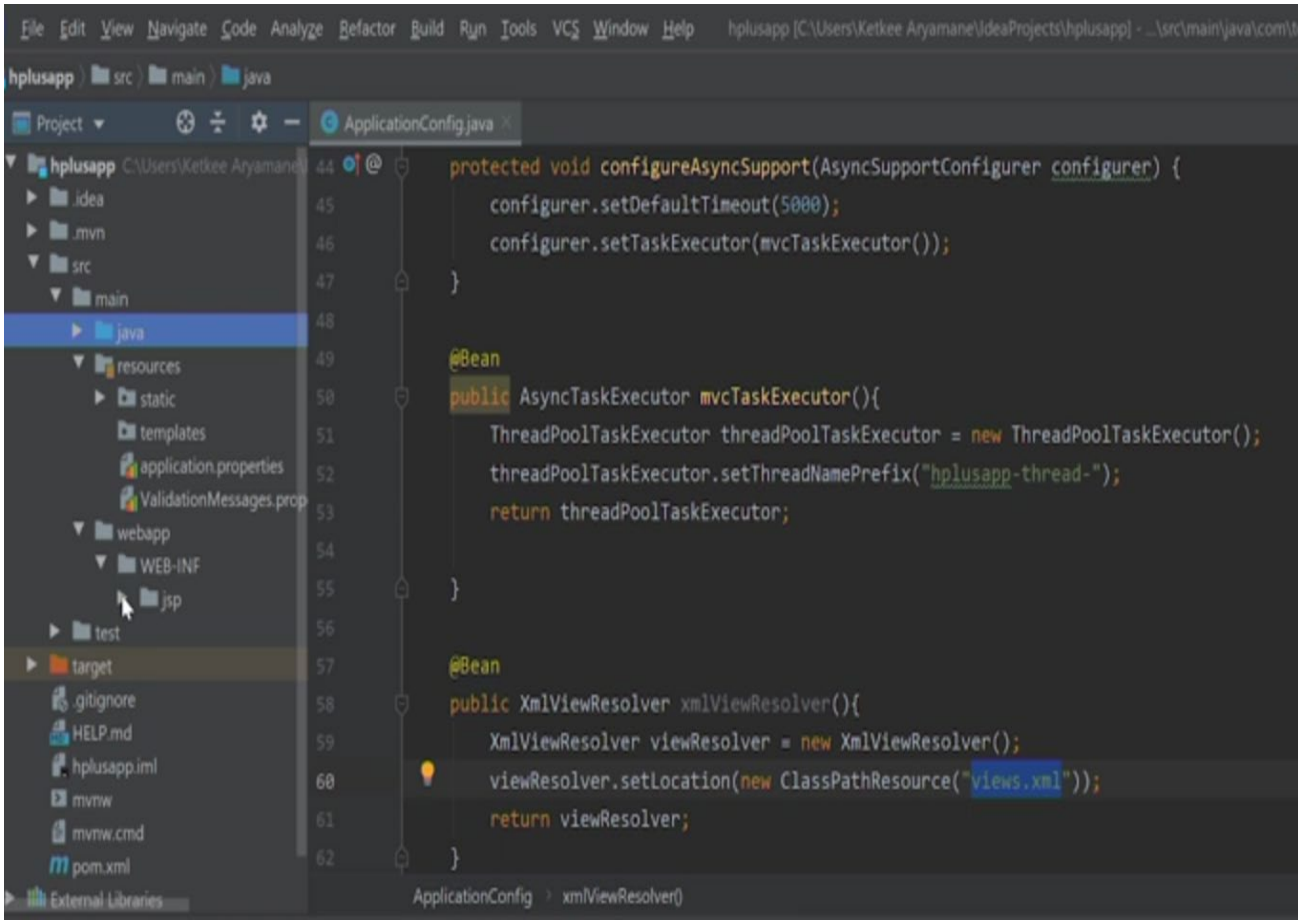
- Any MVC framework needs to render data in browsers
- Spring provides flexibility with view technologies
- Works with two interfaces:
  - `View`
  - `ViewResolver`



Types of View Resolvers

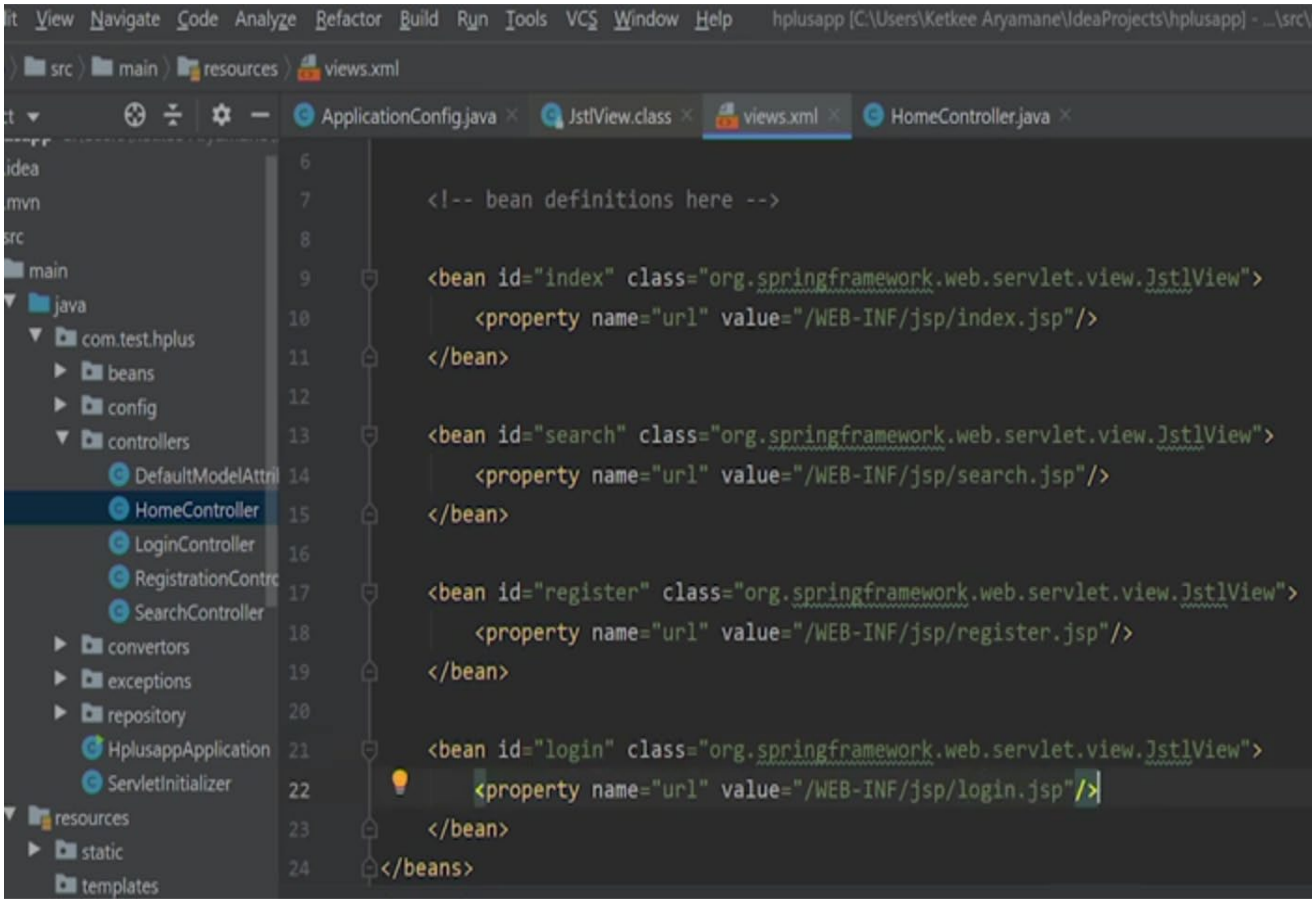
- `InternalResourceViewResolver`
- `ResourceBundleViewResolver`
- `XmlViewResolver`
- `VelocityViewResolver/FreeMarkerViewResolver`

XmlViewResolver Bean in ApplicationConfig.java

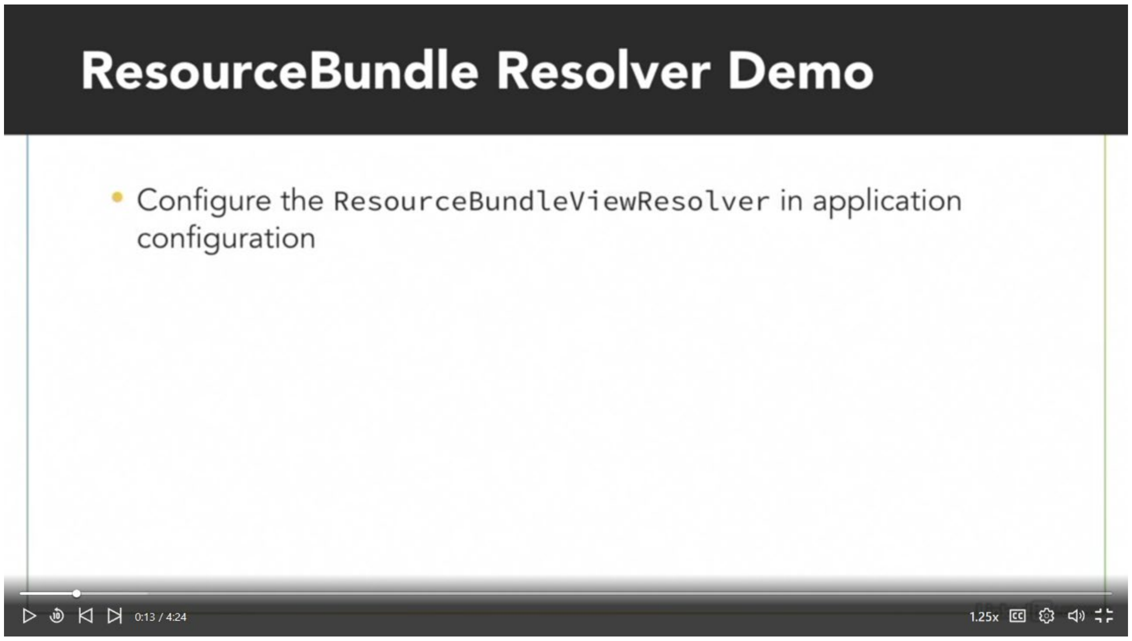


```java
protected void configureAsyncSupport(AsyncSupportConfigurer configurer) {
    configurer.setDefaultTimeout(5000);
    configurer.setTaskExecutor(mvcTaskExecutor());
}

@Bean
public AsyncTaskExecutor mvcTaskExecutor(){
    ThreadPoolTaskExecutor threadPoolTaskExecutor = new ThreadPoolTaskExecutor();
    threadPoolTaskExecutor.setThreadNamePrefix("hplusapp-thread-");
    return threadPoolTaskExecutor;

}

@Bean
public XmlViewResolver xmlViewResolver(){
    XmlViewResolver viewResolver = new XmlViewResolver();
    viewResolver.setLocation(new ClassPathResource("views.xml"));
    return viewResolver;
}
```

Views.xml in resource folder


it View Navigate Code Analyze Refactor Build Run Tools VCS Window Help hplusapp [C:\Users\Ketkee Aryamane\IdeaProjects\hplusapp] - ...\src\
src main resources views.xml
ApplicationConfig.java JstlView.class views.xml HomeController.java
idea
mvn
src
main
java
com.test.hplus
beans
config
controllers
DefaultModelAttri
HomeController
LoginController
RegistrationContr
SearchController
convertors
exceptions
repository
HplusappApplication
ServletInitializer
resources
static
templates

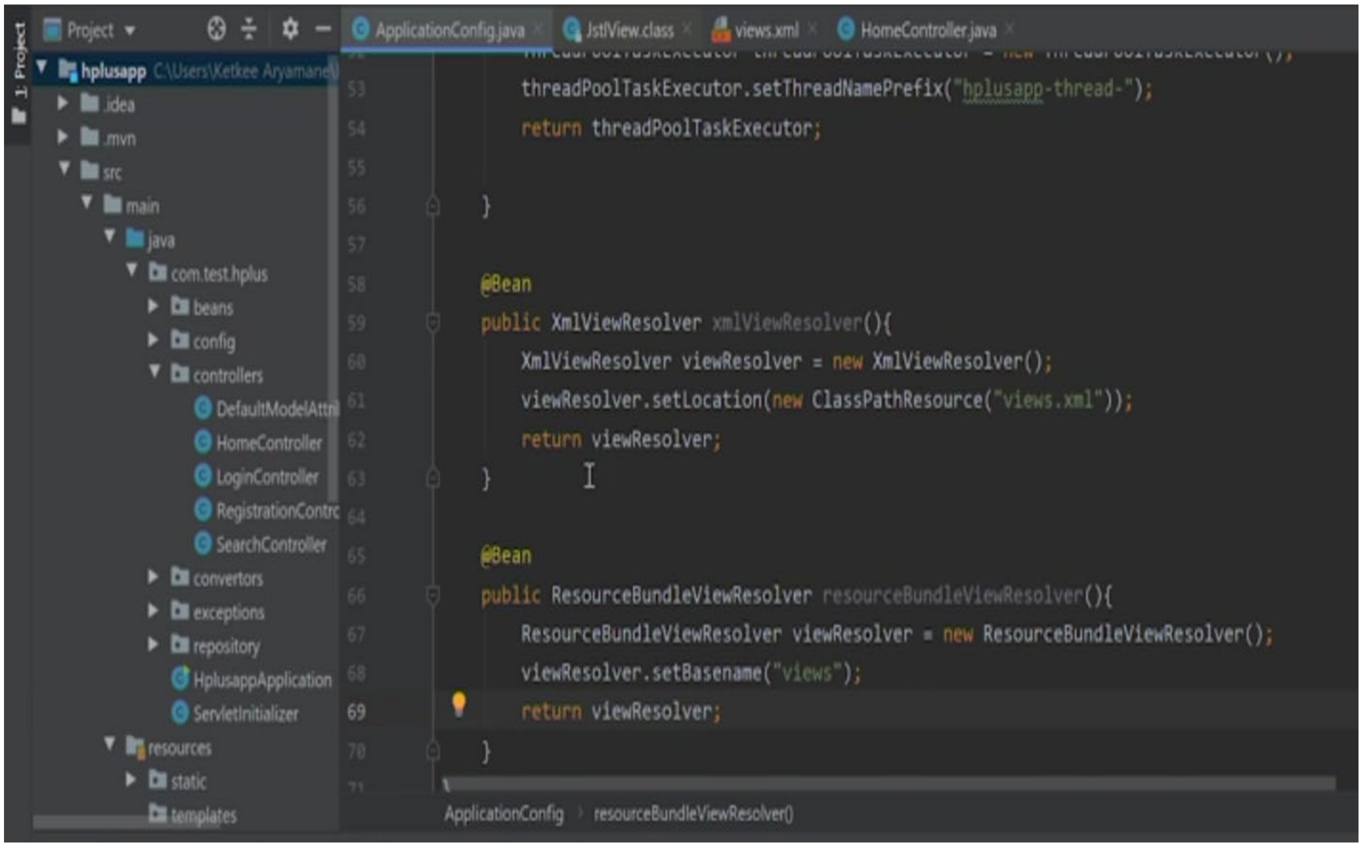
```xml

    <!-- bean definitions here -->

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

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

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

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



ResourceBundle Resolver Demo

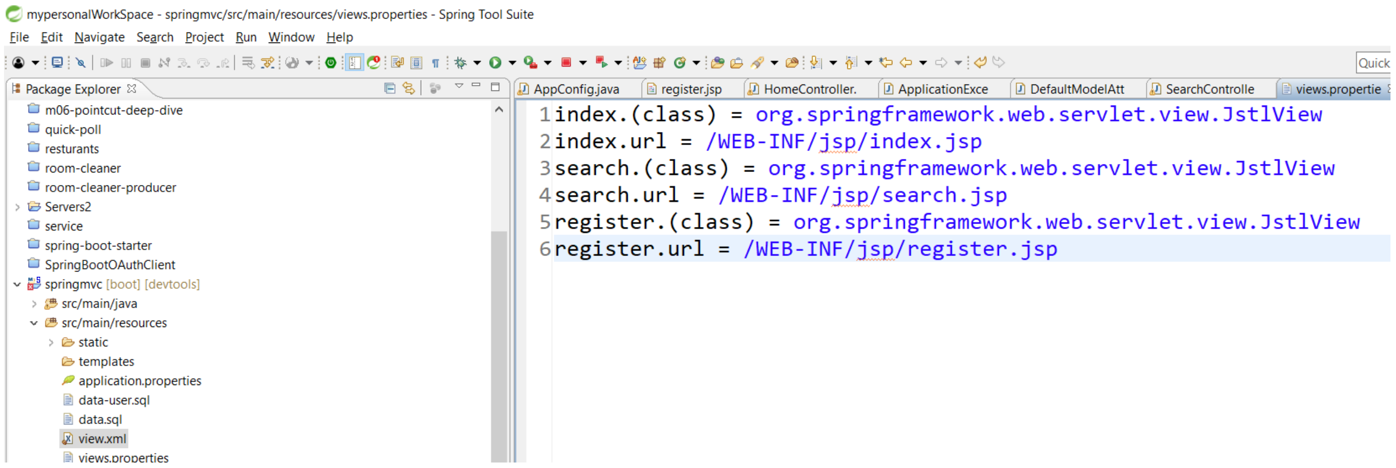
- Configure the `ResourceBundleViewResolver` in application configuration

ResoucreBundle View Resolver in ApplicationConfiguration

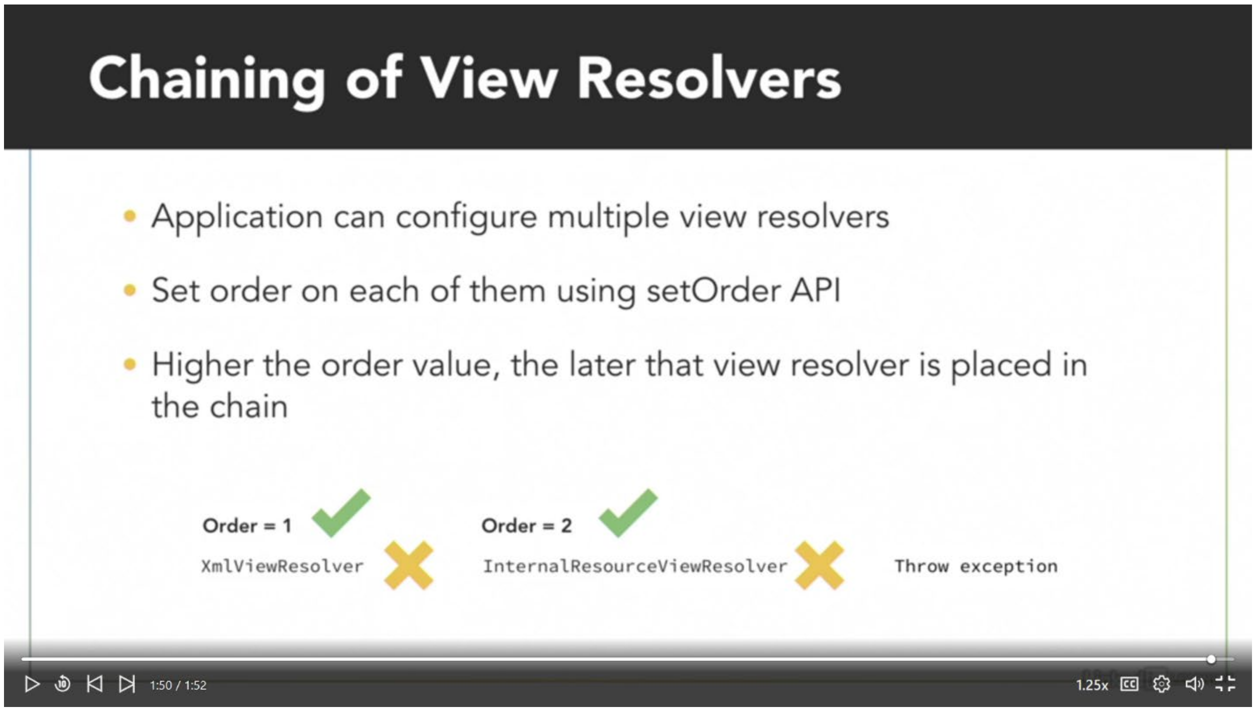


```java
        threadPoolTaskExecutor.setThreadNamePrefix("hplusapp-thread-");
        return threadPoolTaskExecutor;

    }

    @Bean
    public XmlViewResolver xmlViewResolver(){
        XmlViewResolver viewResolver = new XmlViewResolver();
        viewResolver.setLocation(new ClassPathResource("views.xml"));
        return viewResolver;
    }

    @Bean
    public ResourceBundleViewResolver resourceBundleViewResolver(){
        ResourceBundleViewResolver viewResolver = new ResourceBundleViewResolver();
        viewResolver.setBasename("views");
        return viewResolver;
    }
```

Views.properties in resources folder



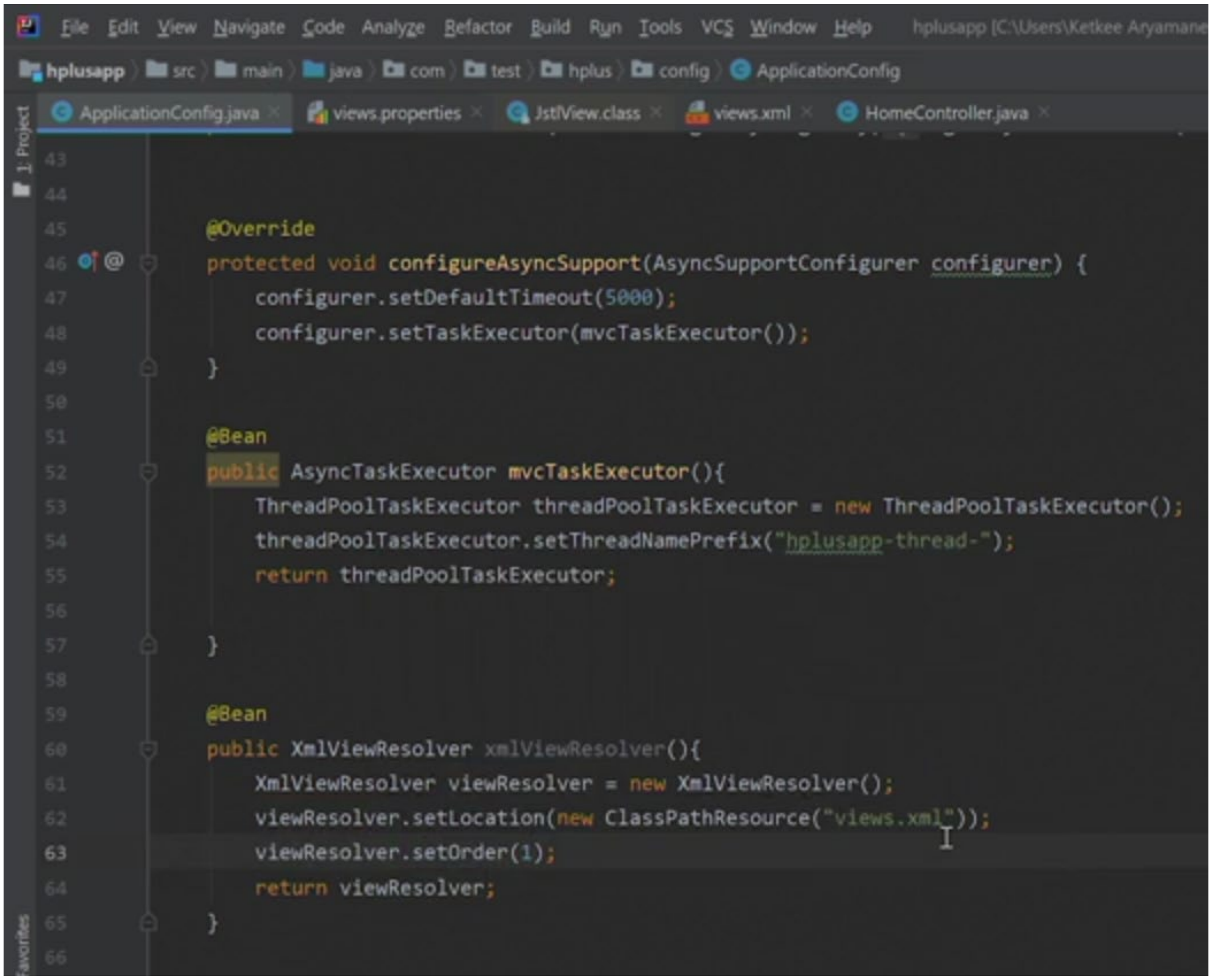
```
index.(class) = org.springframework.web.servlet.view.JstlView
index.url = /WEB-INF/jsp/index.jsp
search.(class) = org.springframework.web.servlet.view.JstlView
search.url = /WEB-INF/jsp/search.jsp
register.(class) = org.springframework.web.servlet.view.JstlView
register.url = /WEB-INF/jsp/register.jsp
```



Chaining of View Resolvers

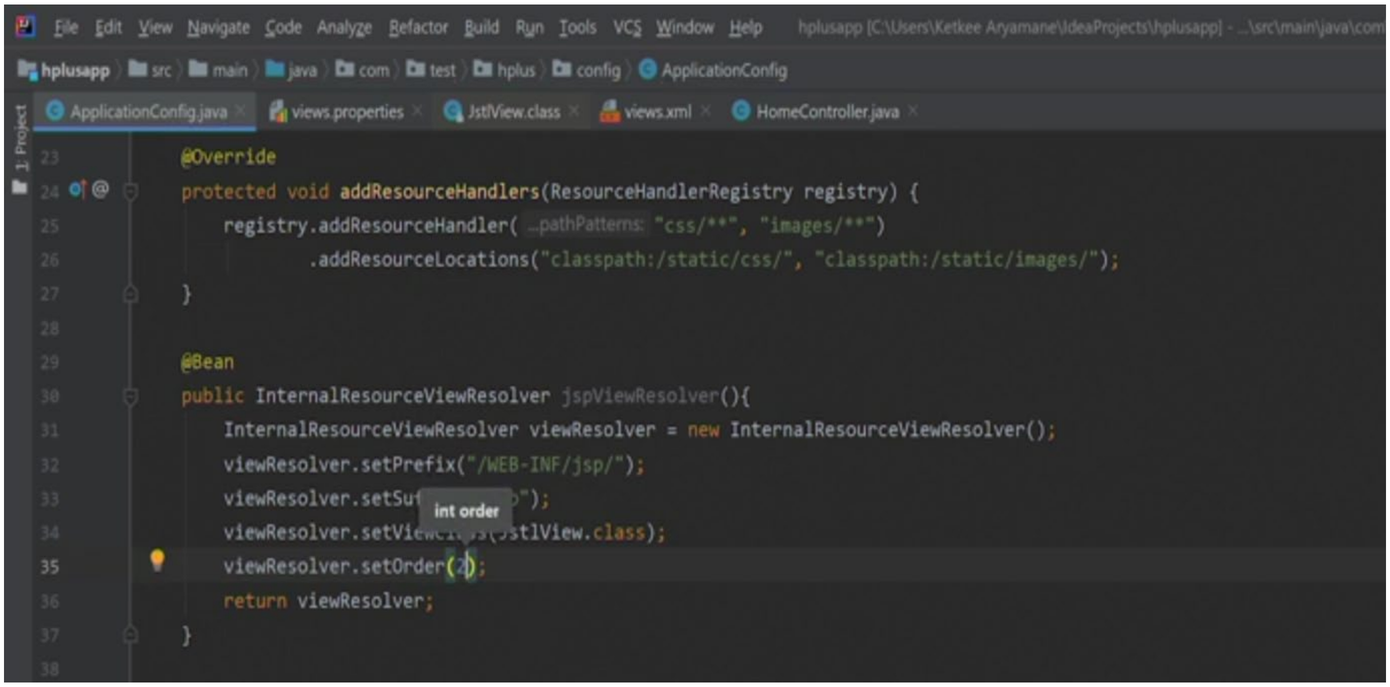
- Application can configure multiple view resolvers
- Set order on each of them using setOrder API
- Higher the order value, the later that view resolver is placed in the chain

Setting order of XmlViewResolver bean to 1 so that first all the views are get resolved by xml view resolver.


File Edit View Navigate Code Analyze Refactor Build Run Tools VCS Window Help hplusapp [C:\Users\Ketkee Aryamane
hplusapp src main java com test hplus config ApplicationConfig
ApplicationConfig.java views.properties JstlView.class views.xml HomeController.java

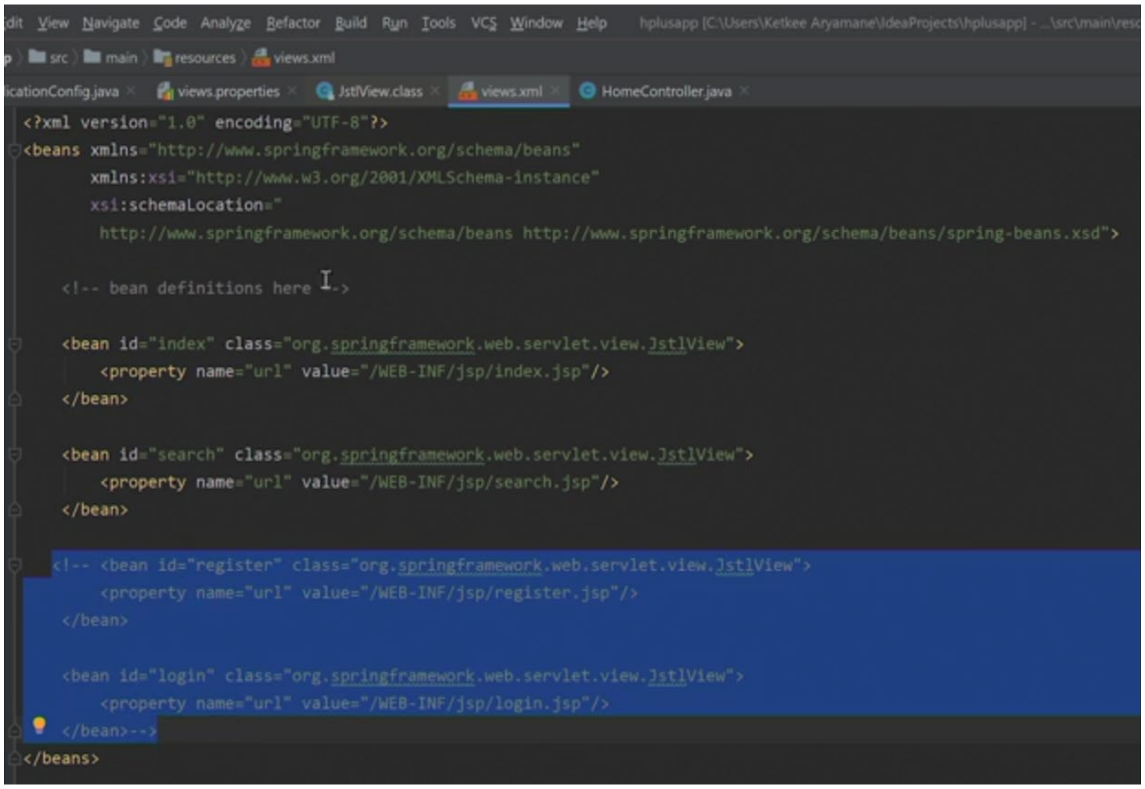
```java


    @Override
    protected void configureAsyncSupport(AsyncSupportConfigurer configurer) {
        configurer.setDefaultTimeout(5000);
        configurer.setTaskExecutor(mvcTaskExecutor());
    }

    @Bean
    public AsyncTaskExecutor mvcTaskExecutor(){
        ThreadPoolTaskExecutor threadPoolTaskExecutor = new ThreadPoolTaskExecutor();
        threadPoolTaskExecutor.setThreadNamePrefix("hplusapp-thread-");
        return threadPoolTaskExecutor;

    }

    @Bean
    public XmlViewResolver xmlViewResolver(){
        XmlViewResolver viewResolver = new XmlViewResolver();
        viewResolver.setLocation(new ClassPathResource("views.xml"));
        viewResolver.setOrder(1);
        return viewResolver;
    }

```

Setting Order of InternalViewResolver bean to 2 so that it will resolve the view once view resolver is done by XmlViewResolver and has failed to resolve some of the views that are going to resolve by the InternalViewResolver .


File Edit View Navigate Code Analyze Refactor Build Run Tools VCS Window Help
hplusapp [C:\Users\Ketkee Aryamane\IdeaProjects\hplusapp] - ...\src\main\java\com
hplusapp › src › main › java › com › test › hplus › config › ApplicationConfig
ApplicationConfig.java
views.properties
JstlView.class
views.xml
HomeController.java
1: Project
int order

```java
    @Override
    protected void addResourceHandlers(ResourceHandlerRegistry registry) {
        registry.addResourceHandler( ...pathPatterns: "css/**", "images/**")
                .addResourceLocations("classpath:/static/css/", "classpath:/static/images/");
    }

    @Bean
    public InternalResourceViewResolver jspViewResolver(){
        InternalResourceViewResolver viewResolver = new InternalResourceViewResolver();
        viewResolver.setPrefix("/WEB-INF/jsp/");
        viewResolver.setSu          ");
        viewResolver.setVie      stlView.class);
        viewResolver.setOrder(2);
        return viewResolver;
    }
```

Commenting some of the view from xml so it is not got resolved by XmlViewResolver and InternalViewResolver Comes to play.


dit View Navigate Code Analyze Refactor Build Run Tools VCS Window Help
hplusapp [C:\Users\Ketkee Aryamane\IdeaProjects\hplusapp] - ...\src\main\res
p › src › main › resources › views.xml
icationConfig.java
views.properties
JstlView.class
views.xml
HomeController.java

```xml
<?xml version="1.0" encoding="UTF-8"?>
<beans xmlns="http://www.springframework.org/schema/beans"
       xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
       xsi:schemaLocation="
        http://www.springframework.org/schema/beans http://www.springframework.org/schema/beans/spring-beans.xsd">

    <!-- bean definitions here -->

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

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

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

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