STRUTS - INTRO

- STRUTS USED ON COMPLEX FRONTEND APPS.

- IT USES MVC MODEL

- Controller is implemented via servlet JAVA.

- This configuration file (`struts-config.xml`) is essential for setting up the MVC (Model-View-Controller) architecture in Struts, where a single servlet acts as the controller, and JSPs are used for the view.
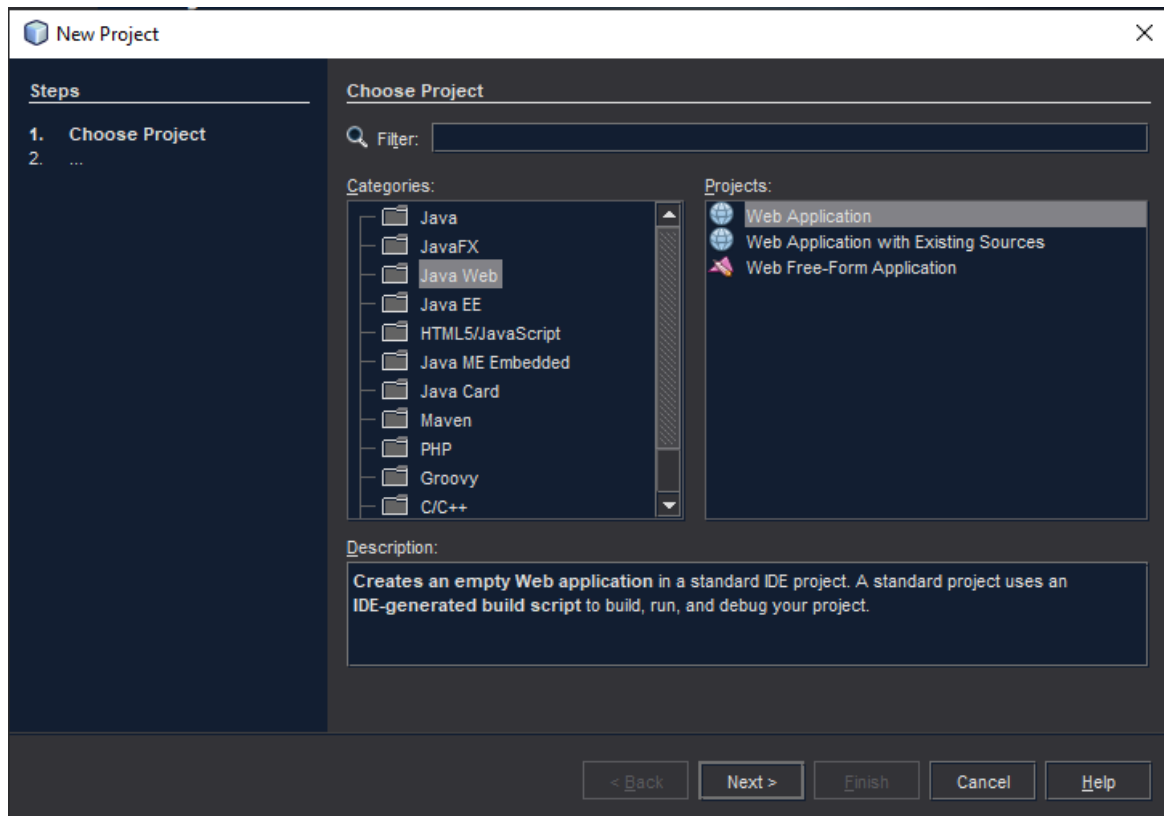
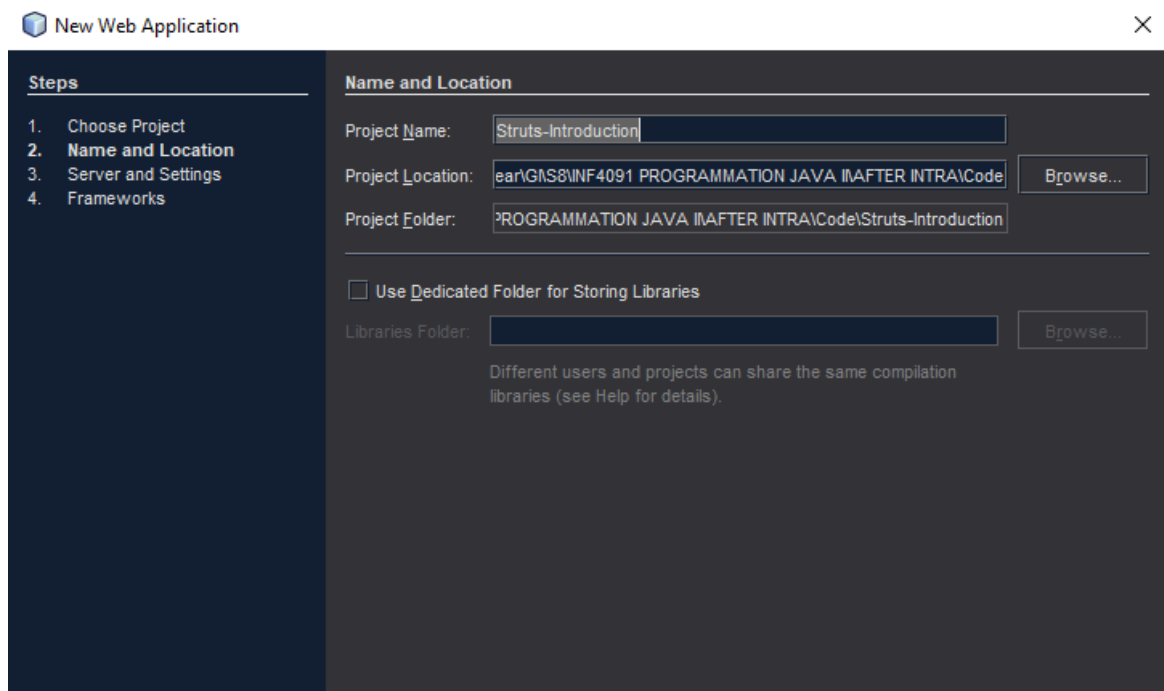- Each url we associate to it an action.

QUESTIONS

▼ STRUTS:

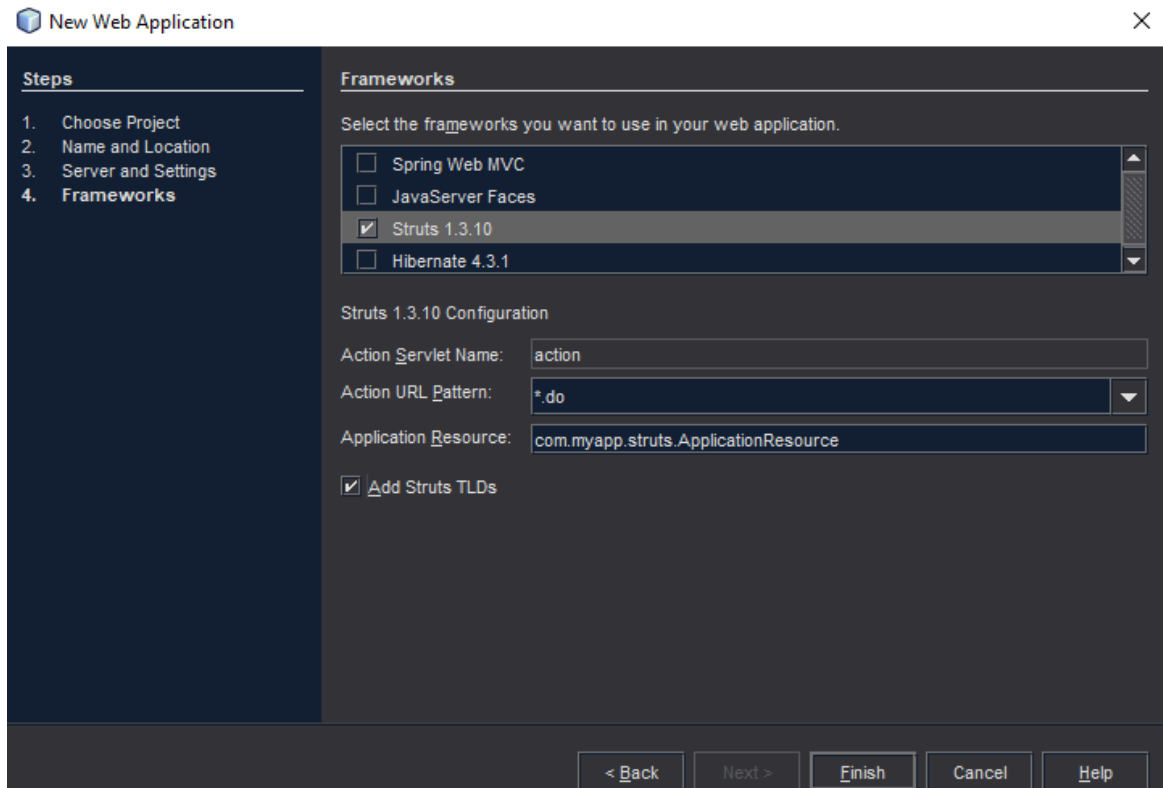▼ How to validate data? with validate attribute in action

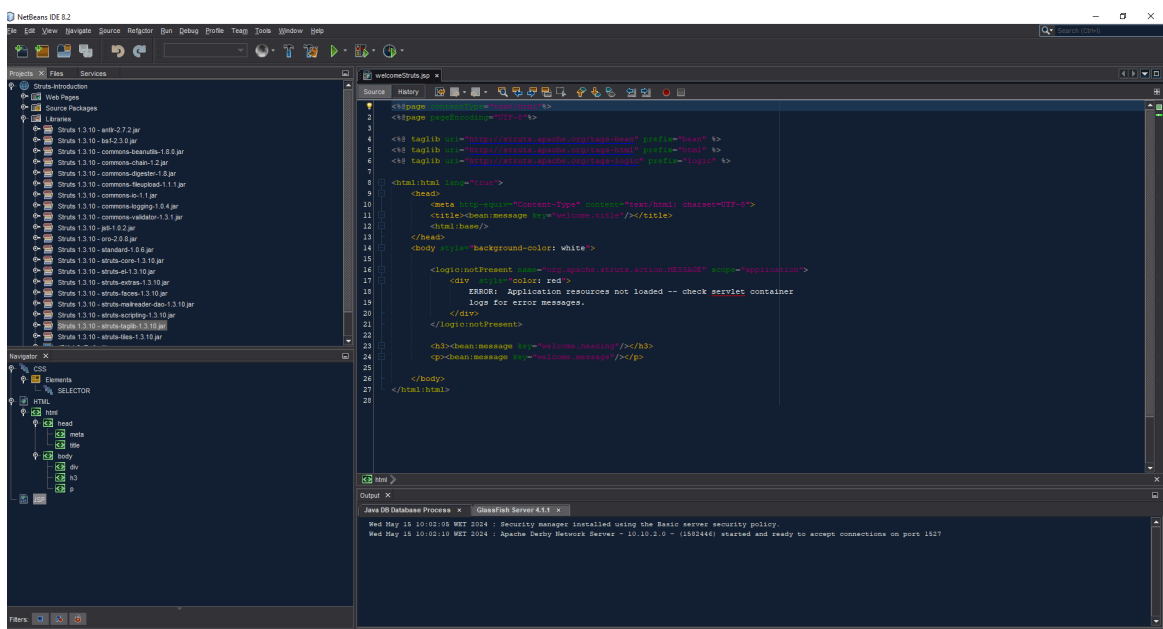**PROJECT WALKTHROUGH (Factorial)**

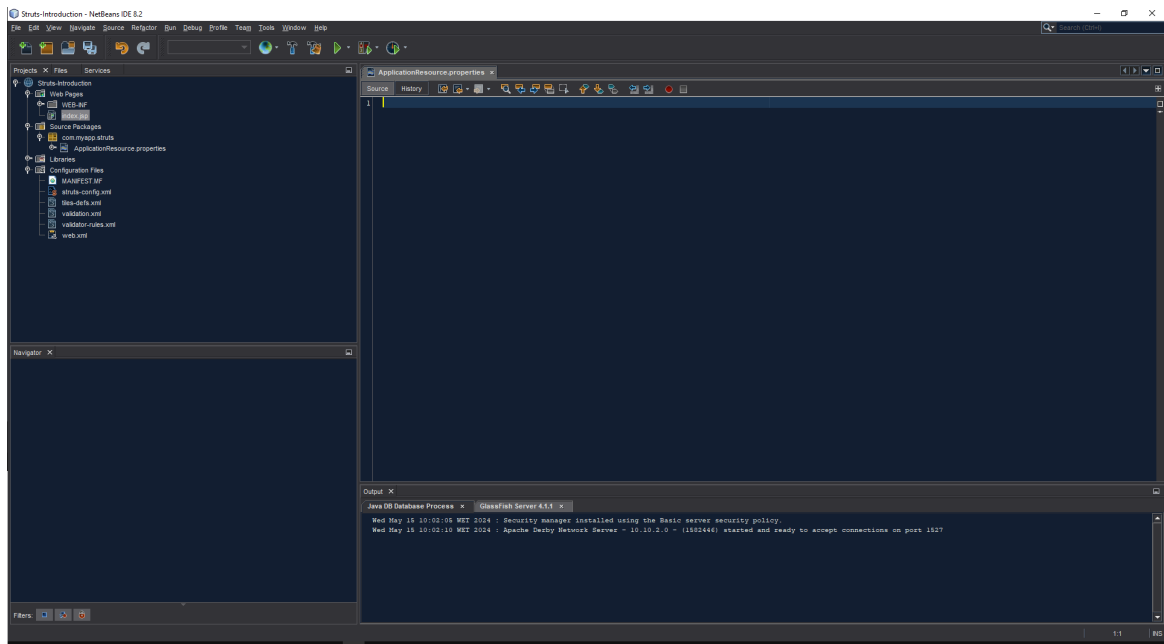# ▼ CREATE PROJECT STRUTS

JAVA WEB, WEB APP

Tag Library Descriptors (to use struts tags, you'll need to check that option)
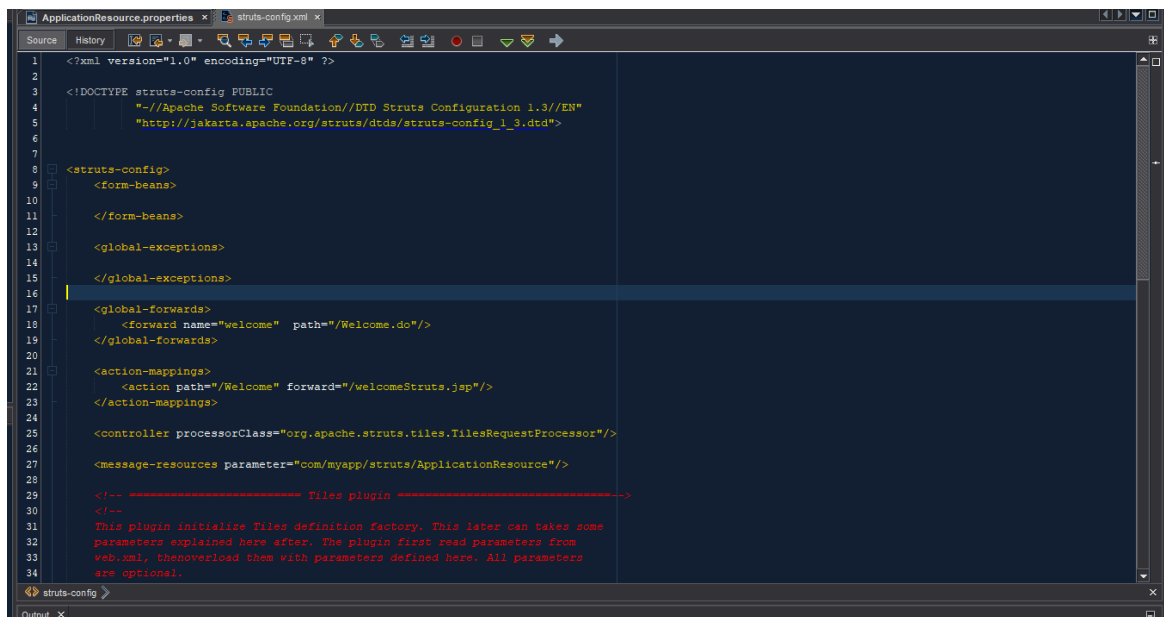

check libraries if you forgot something u won't see all these libraries, and by default the first page is `WelcomeStruts.jsp`
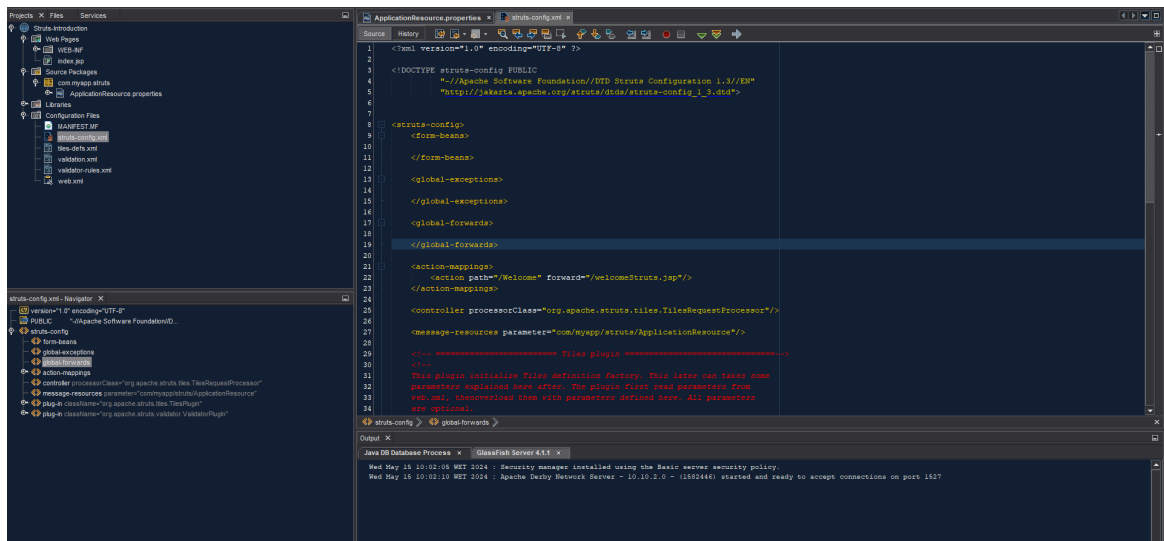
▼ MANIPULATION

Just delete the page `WelcomeStruts.jsp` and the content in `ApplicationResource.properties`
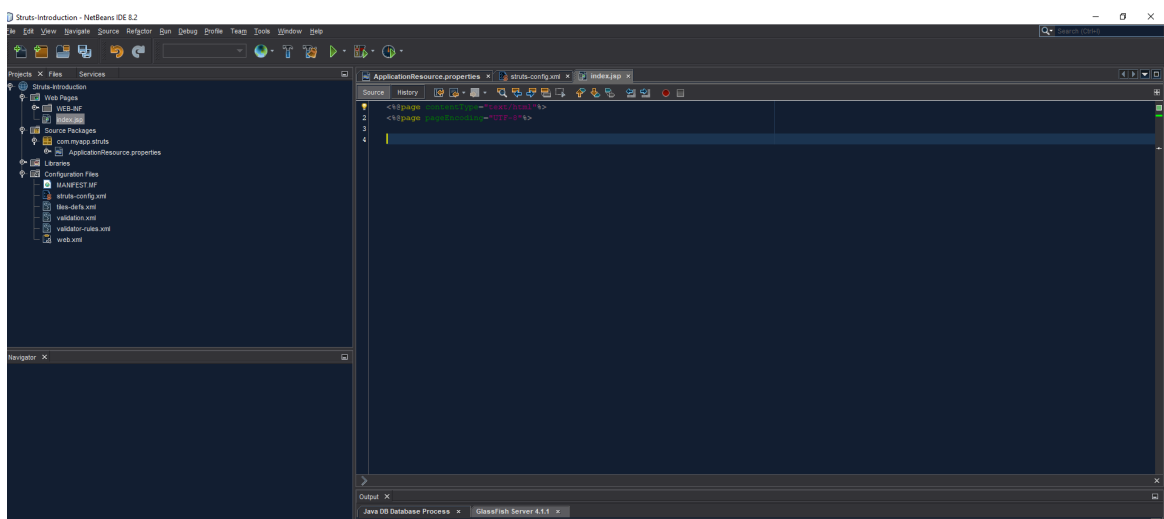
- In `struts-config.xml` Delete the action `welcome.do` as we didn't create it and it was added by default in our project

before deleting the action
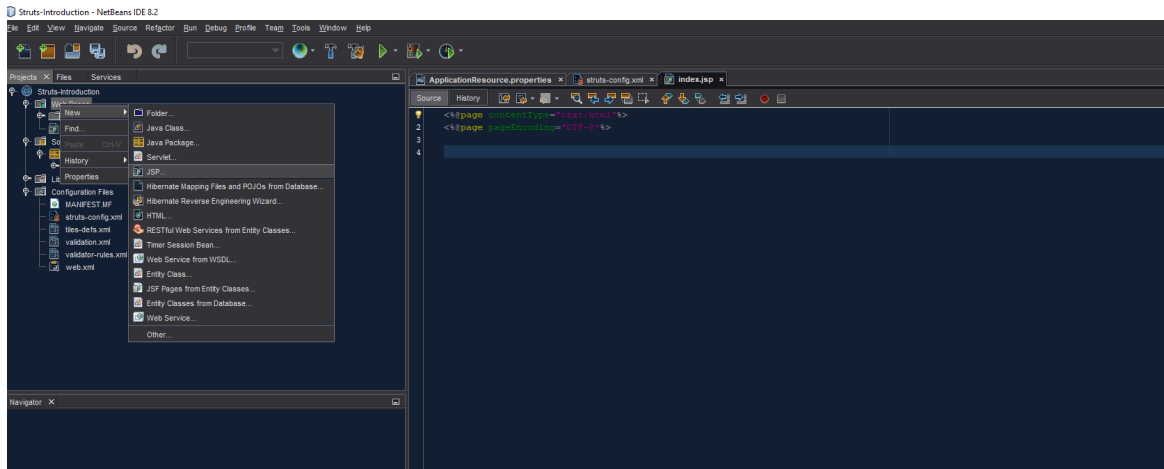
After deleting and saving!

Delete the jsp forward and start creating your own view with the controller...
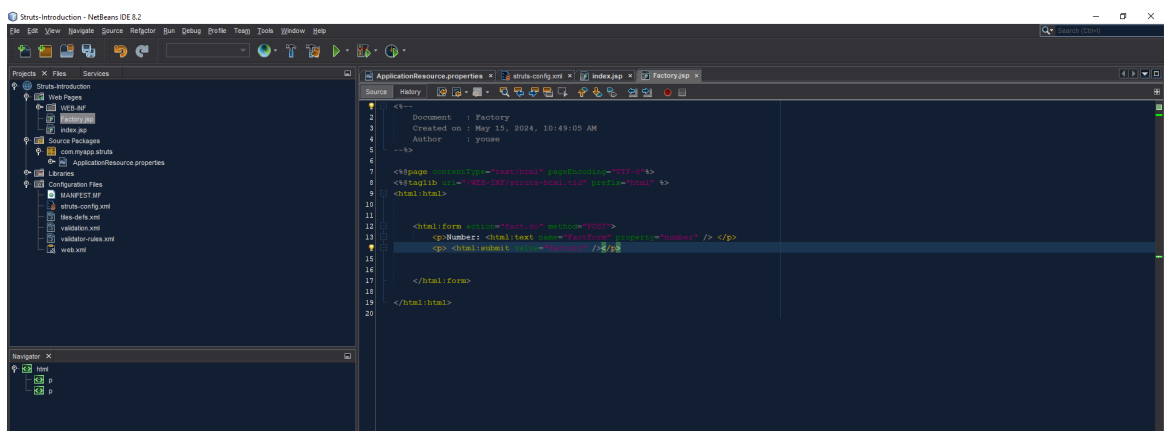
▼ CREATE JSP PAGE

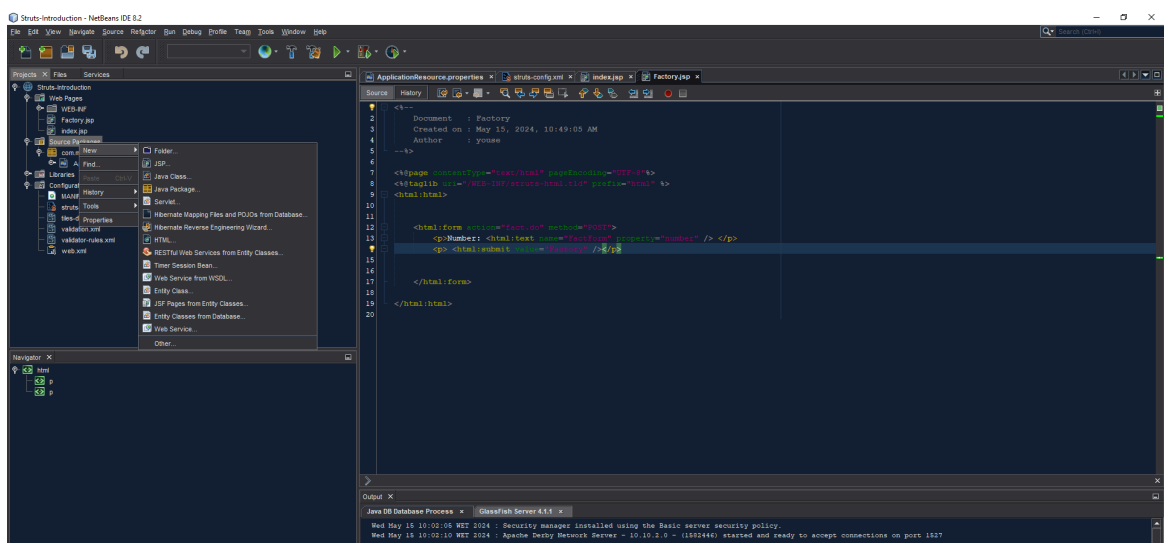So, in this first example we will be creating a page that calculate factory of a number...

- **Example:** 3! = 6

- Let's start by creating the jsp page

- Fill it with struts tags, not html

- Create our packages for organi purposes

I created 3 packages: DAO(Data Access Object), Controller and Model
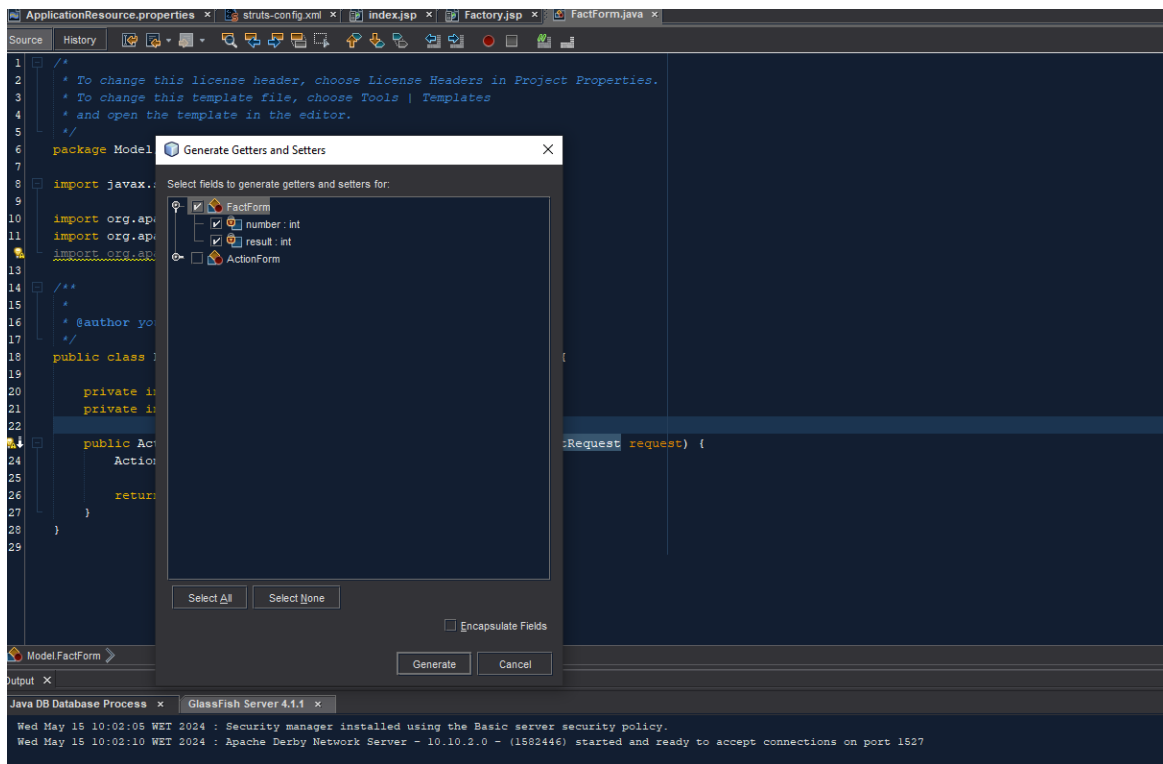
- Then in the `Factory.jsp` we will

the same name as the model in the view

- Then delete the getters and setters by default, and add my own attributes I'll be using `number` and `result`
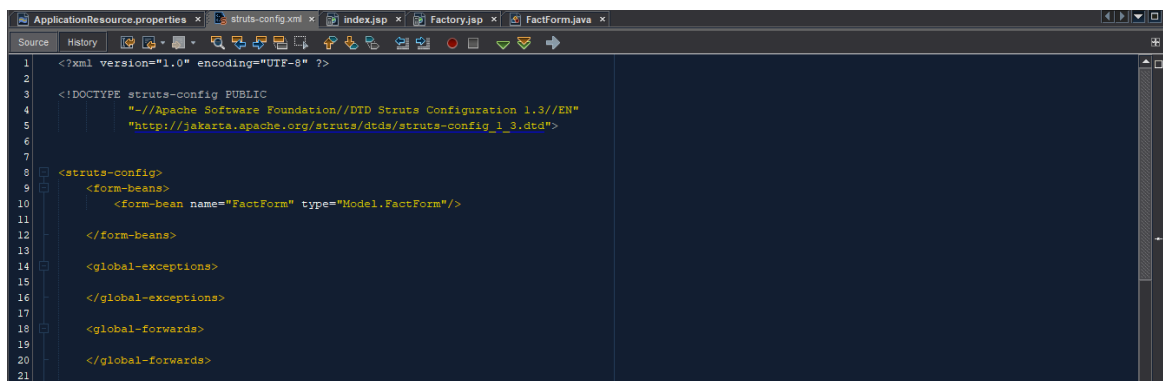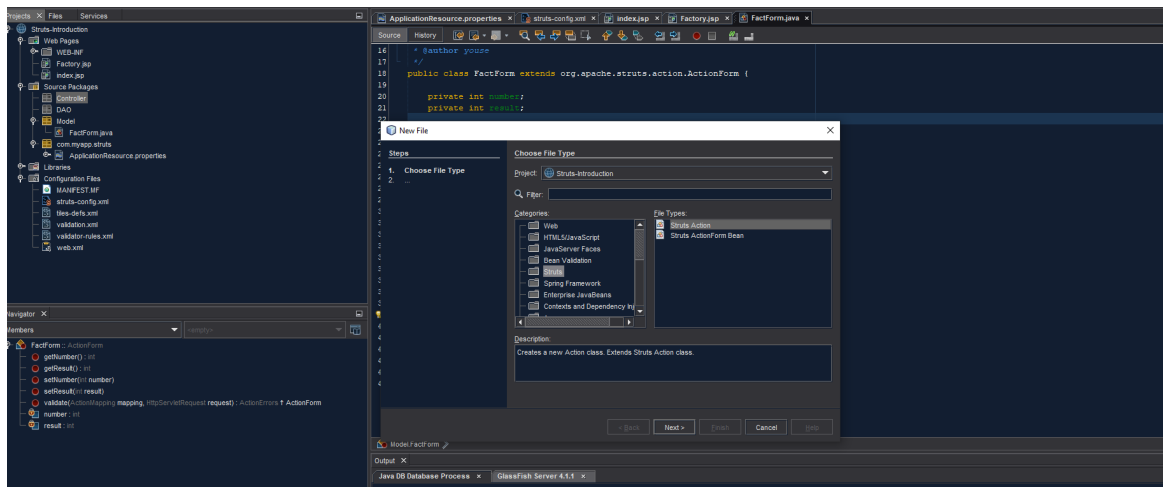

- Add getters and setters for the attributes we added

NB: If you go back to the struts-config.xml, you'll notice we have the model declared in it
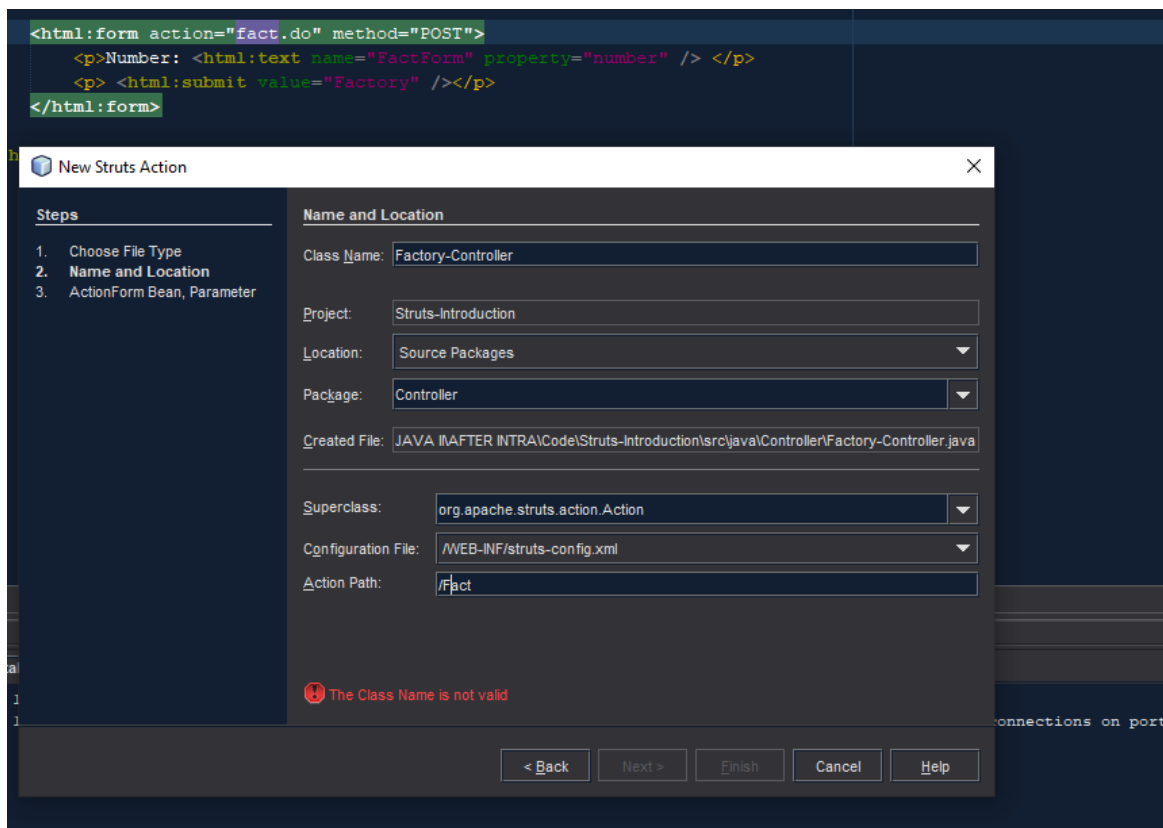

- Now we will need to create the action `fact.do` which will be created in the controller package!

Now we will be using Struts Action

- Don't forget add the path the same as the one you typed in the jsp page

- And then ActionForm will be used the one we created earlier FactForm and for the input the jsp page which is `Factory.jsp`
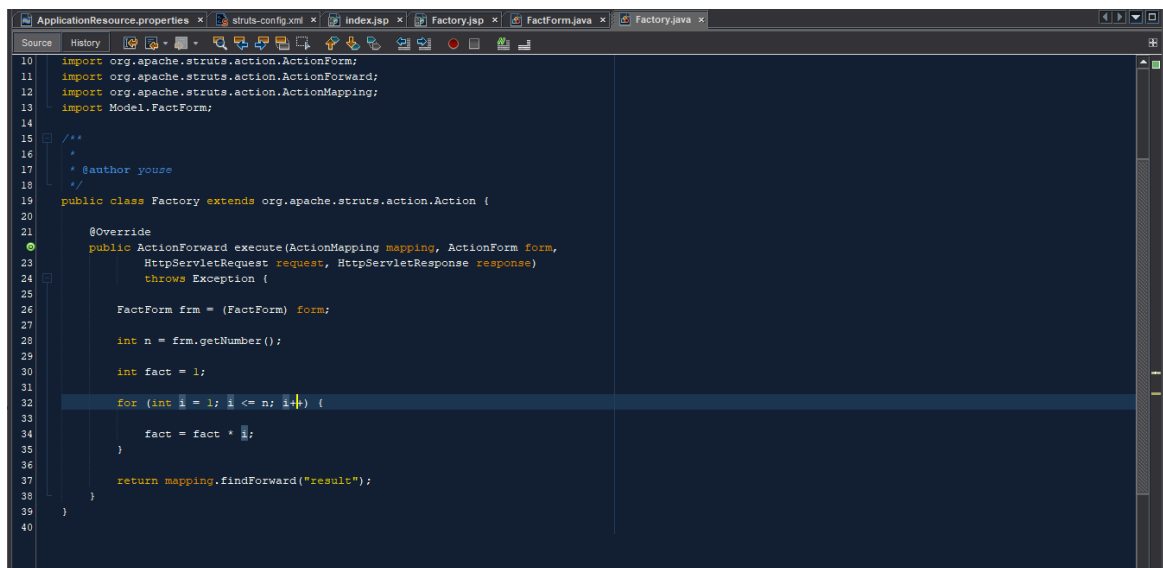
- Then import the FactForm in your controller

delete the uneccessary things

- Add the logic in your controller

- Go create the `result.jsp` page !
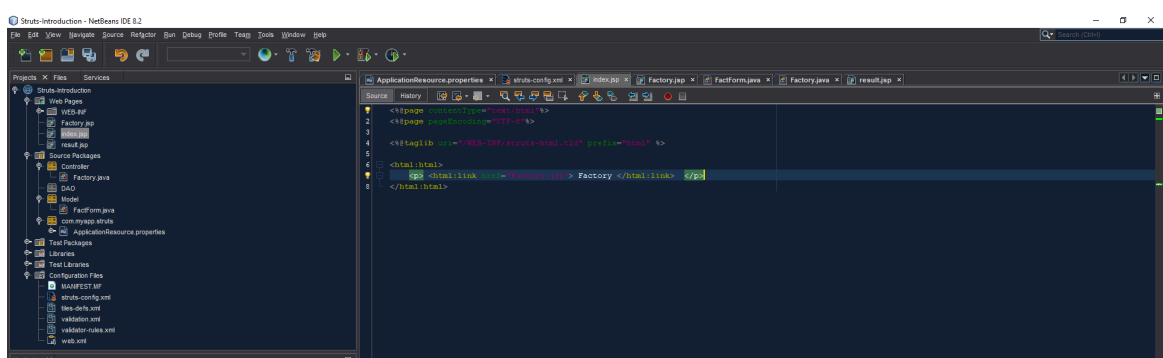
- Fill the index page to link the index with the logic of the

- Go back to `struts-config.xml` and add the global-forward for the result page

```
<struts-config>
    <form-beans>
        <form-bean name="FactForm" type="Model.FactForm"/>

    </form-beans>

    <global-exceptions>

    </global-exceptions>

    <global-forwards>
        <forward name="result"  path="/result.jsp"/>
    </global-forwards>

    <action-mappings>
        <action input="/Factory.jsp" name="FactForm" path="/fact" scope="session" type="Controller.Factory"/>
        <action path="/Welcome" forward="/welcomeStruts.jsp"/>
    </action-mappings>

    <controller processorClass="org.apache.struts.tiles.TilesRequestProcessor"/>

    <message-resources parameter="com/myapp/struts/ApplicationResource"/>
```
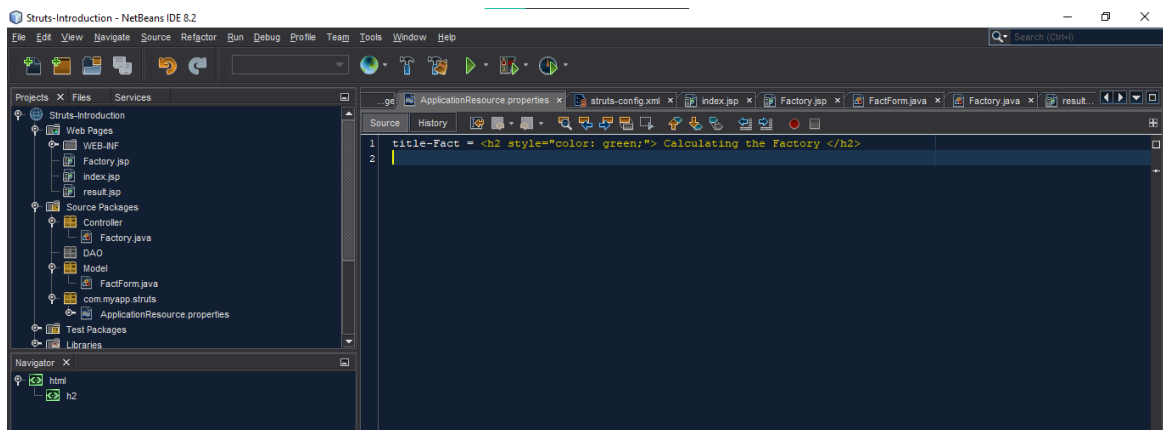
- Finally you can add a message with the style from
  `ApplicationResource.properties`

we added the title-fact with a custom style (css)

- Then the last thing just call title-fact in `result.jsp` page

use `bean:message` to retrieve localized messages from resource bundles