Evan Wilson / Nicholas Thomas

CST-235

September 2nd, 2018

Instructor: Dr. Toure

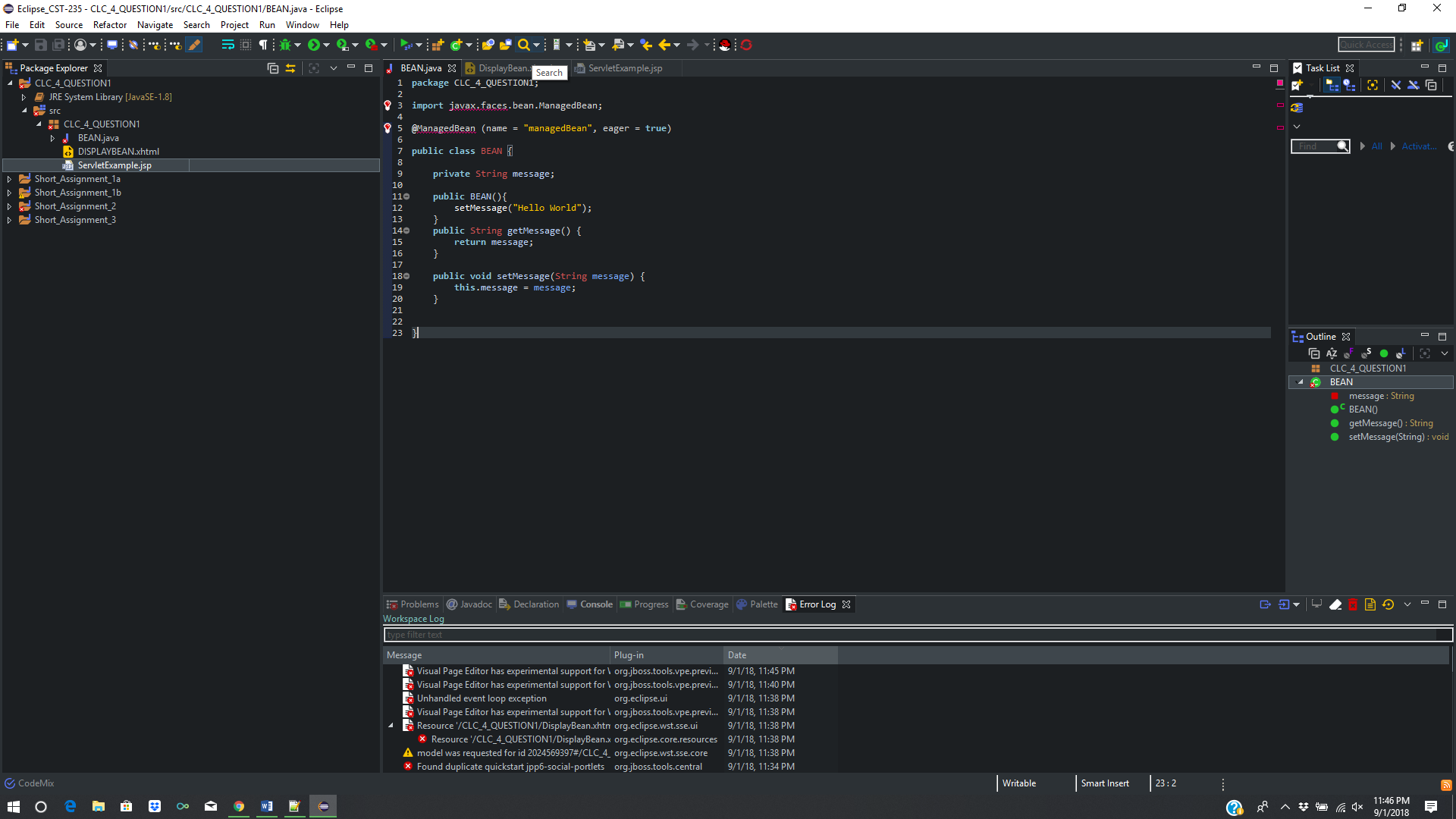
CLC Mini Project 4

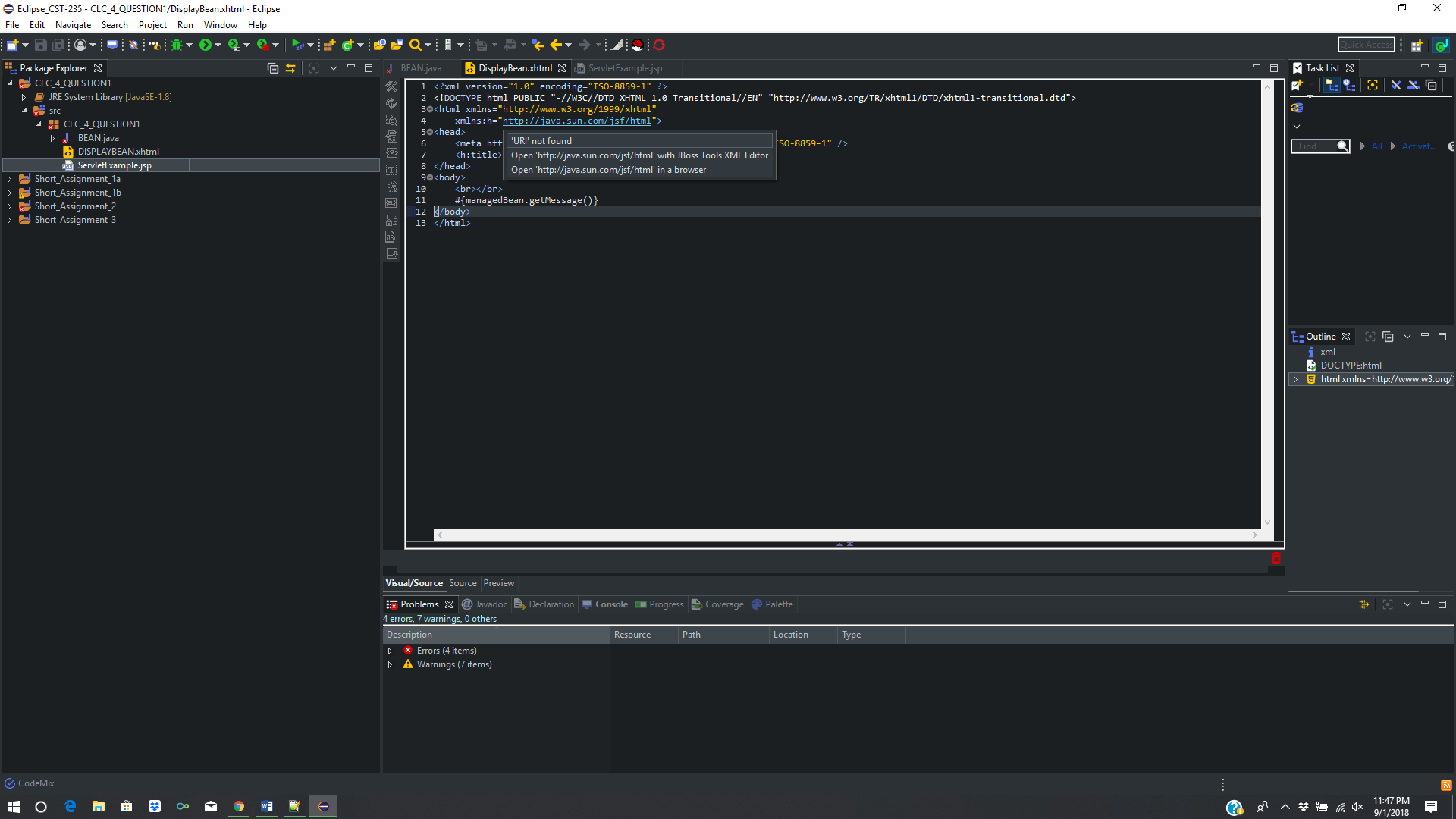
Green Group

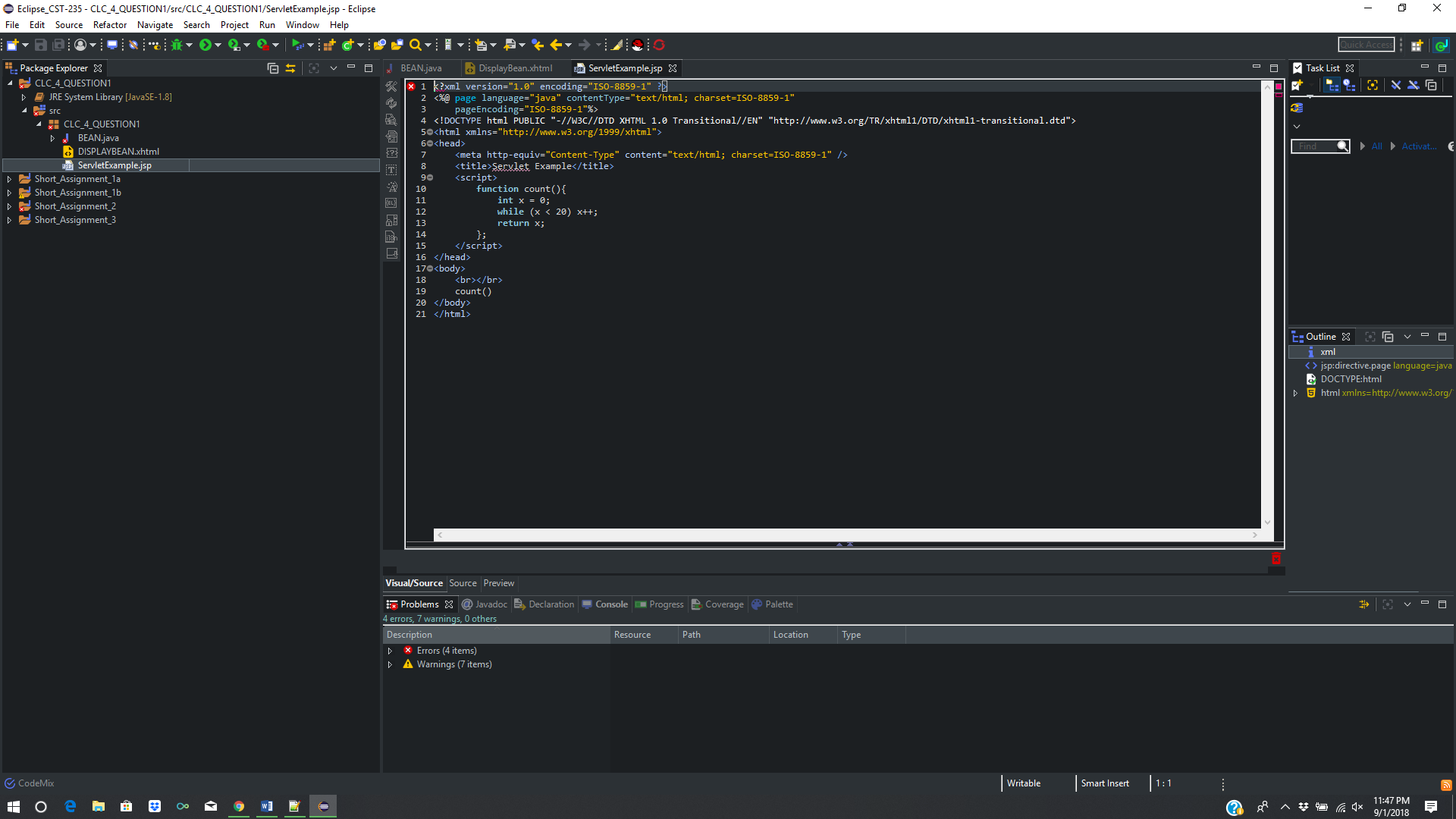
Question 1: What is the difference between JSF servlets and JSF applications?

JSF applications are request driven webpages that are generated when interacted upon. JSF servlets are predefined code and actions that talk between the client and server through web content and use XML and HTML files to connect and communicate with the java code portion of the applications.

When trying to run the program for question, I ran into many errors, when trying to correct them through what Eclipse had recommended, it still failed to run properly. Below is the code for my examples, I do feel I may have added my xhtml and jsp files wrong or I just don’t have the proper add-ons or features checked.







Question 2: When and why would you need to configure JSF?

JSF configuration settings are done in the web.xml and faces-config.xml files located in the WEB-INF directory of the application. Many of the settings can be created by the IDE during the new project or new file wizards in the respective IDE. However, sometimes settings within these XML files will need to be adjusted, and sometimes faces-config.xml file does not even get created the by wizard.

The web.xml file contains general web settings for the application, and faces-config.xml allows for application settings, control of managed beans, converters, validators and page navigation (Vogel, 2018). When a managed bean is declared in the faces-config.xml file, the JSF page gains access to the beans values through the use of expression language (EL) tags ${className.methodName} and #{className.methodName}.

Additionally, the faces-config.xml can impact application behavior, performance and security (Kiswani, 2017). A few ways this is done is the ability to force the use of a specific version of JSF with the following tag example for JSF 2.2:

<faces-config

xmlns="http://xmlns.jcp.org/xml/ns/javaee"

xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"

xsi:schemaLocation="http://xmlns.jcp.org/xml/ns/javaee http://xmlns.jcp.org/xml/ns/javaee/web-facesconfig\_2\_2.xsd"

version="2.2">

</faces-config

Another configuration that often needs to be done is with time zones. JSF defaults all time to UTC instead of the system clock. This can cause issues when dealing with dates and such and can be fixed like this:

<context-param>

<param-name>javax.faces.DATETIMECONVERTER\_DEFAULT\_TIMEZONE\_IS\_SYSTEM\_TIMEZONE</param-name>

<param-value>true</param-value>

</context-param>

Lastly, many other configurations can be done in the faces-config.xml and web.xml files to aid in the development, production, debugging and error-page displaying. One problem with JSF servlet behavior is sometimes not correctly showing exceptions or errors. The files can force logging of these issues for proper display of error screens and such like this:

<error-page>

<exception-type>com.sun.faces.context.FacesFileNotFoundException</exception-type>

<location>/pages/error/file-not-found.xhtml</location>

</error-page>

<error-page>

<exception-type>java.lang.RuntimeException</exception-type>

<location>/pages/error/bug.xhtml</location>

</error-page>

<error-page>

<exception-type>java.lang.Throwable</exception-type>

<location>/pages/error/error.xhtml</location>

</error-page>

<error-page>

<error-code>404</error-code>

<location>/pages/error/page-not-found.xhtml</location>

</error-page>

# References

Kiswani, J. (2017, July 26). *Java Server Faces (JSF) Web-Applications: Tuning and Configuration*. Retrieved from Java Zone: https://dzone.com/articles/tuning-and-configuring-java-server-facesjsf-web-ap

Vogel, L. (2018, February 13). *JSF (JavaServer Faces) - Tutorial*. Retrieved from Vogella: http://www.vogella.com/tutorials/JavaServerFaces/article.html