Create a JSF project (With CDI support)

Including and using Resources

Folder Structure required

Include a stylesheet in JSF page

Accessing resource in JSF page

Find dependency in War file

***Create a JSF project (With CDI support)***

The following steps create a simple JSF/CDI project (Archetype done by [Weld Community](http://weld.cdi-spec.org/community/))

1 – Enter Maven command (in a DOS prompt)

mvn archetype:generate -DgroupId=***com.mycompany.projectname***

-DartifactId=***projectname***

-DarchetypeGroupId=org.jboss.weld.archetypes

-DarchetypeArtifactId=weld-jsf-jee-minimal

-DarchetypeVersion=1.0.0-BETA1

***Including and using Resources (css, images & Java Script)***

***Folder Structure required***

JSF appears to expect a very structured folder if we want to use them in a JSF application

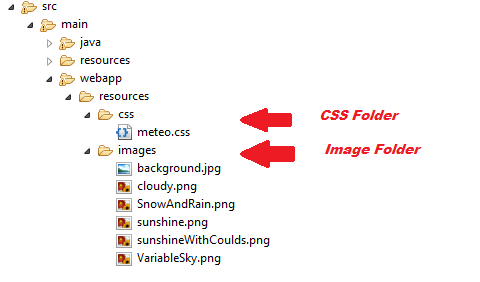
All artifacts grouped under a resources folder

A sub folder have to exist for the

* ***css*** (resources/css)
* ***images*** (resources /images)

***Note*** : a ‘s’ at the end of images fixed many problems

* ***javascript*** (resources/javascript)



***Include a stylesheet in JSF page***

<h:head>

<h:outputStylesheet library=*"css"* name=*"meteo.css"* />

</h:head>

***Accessing resource in JSF page***

The following syntax has worked very well so far

Note: You must have the correct folder structure in place (described above)

page-main-section **{**

background**:** url(*"#{resource['images/background.jpg']}"*)**;**

**}**

***Note***: the macro resource appears to have no ‘***s***’ at the end, despite the fact that the corresponding folder has an ‘s’

***Find dependency in War file***

Situations where a class not found exception happens during Web application startup or during the executions is frequently happening. A check at the ***class path*** and ***Deployment assembly*** of the Web Project (in Eclipse) will fix the problem in most situations.

But if doesn’t address the problem, a closer look at the dependency of the web application’s WAR file can greatly help the troubleshooting

***jar -tf MyWarFileName.war***