



# Advanced Java

*Trainer: Nilesh Ghule*



# Java EE webapp directory structure

Java EE web appn must be arranged in a fixed format while deploying in web server.

webapp/

- \*.html, \*.jsp
- \*.png
- \*.js, \*.css

} → may be added in sub directories

WEB-INF/

- web.xml

→ deployment descriptor

classes/

- \*.class

→ pojo, dao, | in  
servlets, ... | packages

lib/

- \*.jar

→ third-party libraries  
e.g. jdbc driver, jstl, ...

appn should deployed as a directory or as war (zip) file with  
← given structure.



To be deployed in web server → Hot deployment folder i.e. webapps in tomcat.

Note: Contents in WEB-INF are not directly accessible to client (by url), but contents above it are accessible (by url).



# JSP

- JSPs are outdated.

Servlet = Business Logic + Presentation Logic

JSP = Presentation Logic + Business Logic

\* JSP is converted into Servlet while execution.

\* JSP syntax: basically JSP is markup (tags).

① Directive `<%@ ... %>` → Defines JSP behavior, ...  
@Page, @include, @taglib

② Declaration `<%! ... %>` → fields & methods to be added in generated Servlet class.

③ Scriptlet `<% .... %>` → Java statements to be executed per request.

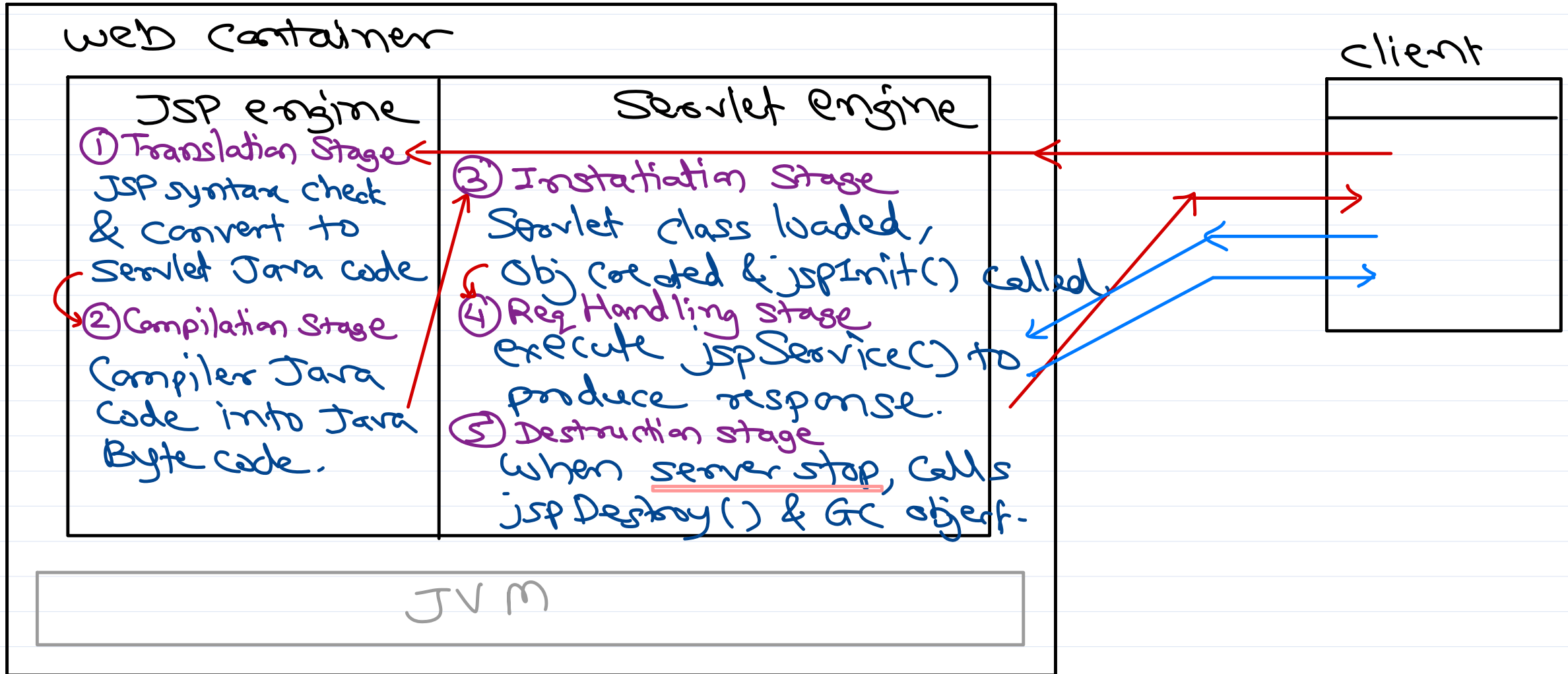
④ Expression `<%= ... %>` → Java expression which is evaluated per request & result is added in generated response.

⑤ Comment `<%-- ... --%>` → server side comment - not added to response.



# JSP Life Cycle

web server



# JSP Implicit Objects

\* local variables available in `jspService()` method.

\* They are directly available in req handling stage (no explicit declaration needed) i.e. scriptlet & expression.

- ① request : `HttpServletRequest`
- ② response : `HttpServletResponse`
- ③ session : `HttpSession`
- ④ config : `ServletConfig`
- ⑤ application : `ServletContext`
- ⑥ page : `Object` (current servlet obj)  
i.e. this
- ⑦ pageContext : `PageContext`
- ⑧ out : `JspWriter`
- ⑨ exception : `Throwable` \* only avail in JSP error pages.





*Thank you!*

Nilesh Ghule <nilesh@sunbeaminfo.com>

