Advanced Java

Trainer: Nilesh Ghule

Movie Review Assignment

Sunbeam Infotech

www.sunbeaminfo.com
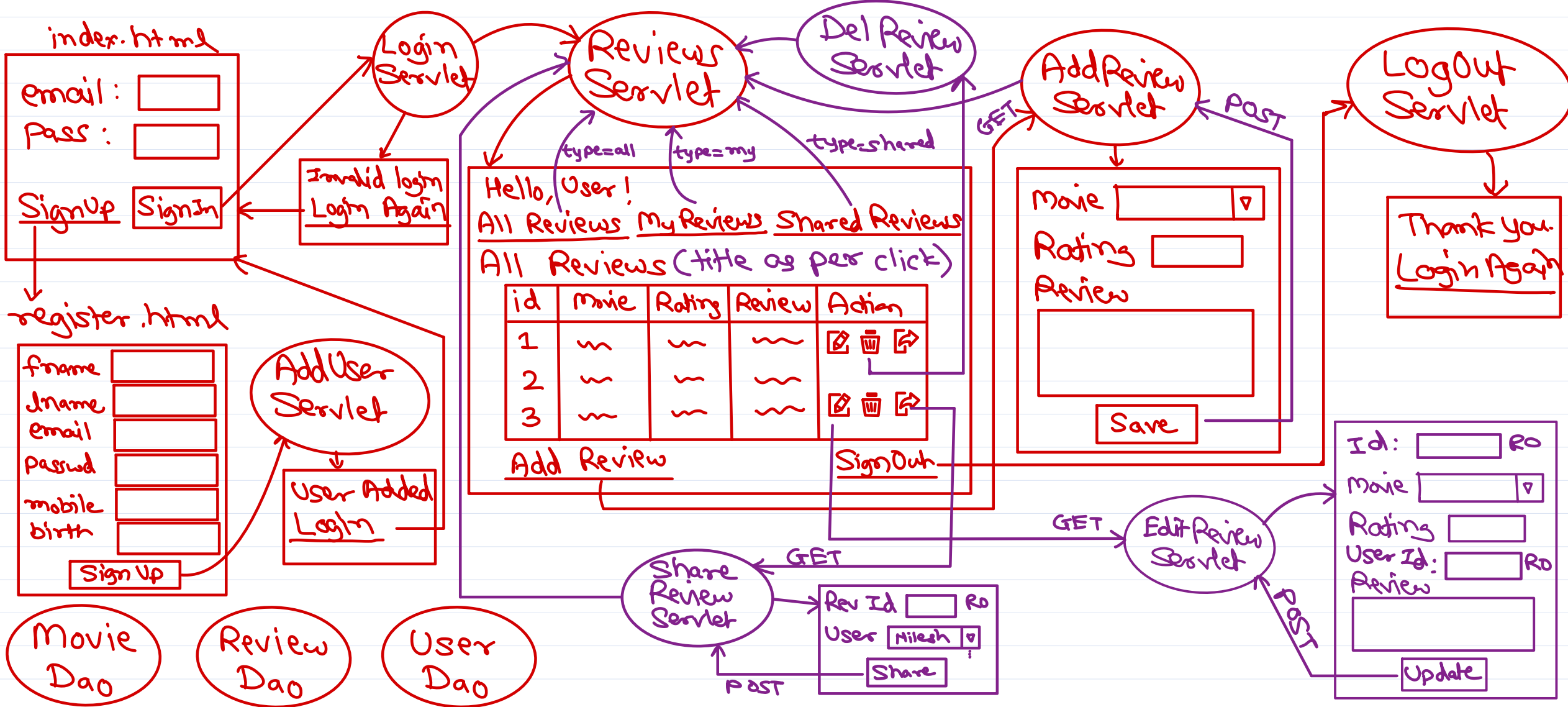
JSP = Java Server Pages

Servlet = **Business Logic** + Presentation Logic

JSP = **Presentation Logic** + Business Logic
        ↳ markup (tags)   ↳ Java

JSP is converted into Servlets while execution.

Ideal JSP doesn't have any Java code in it.
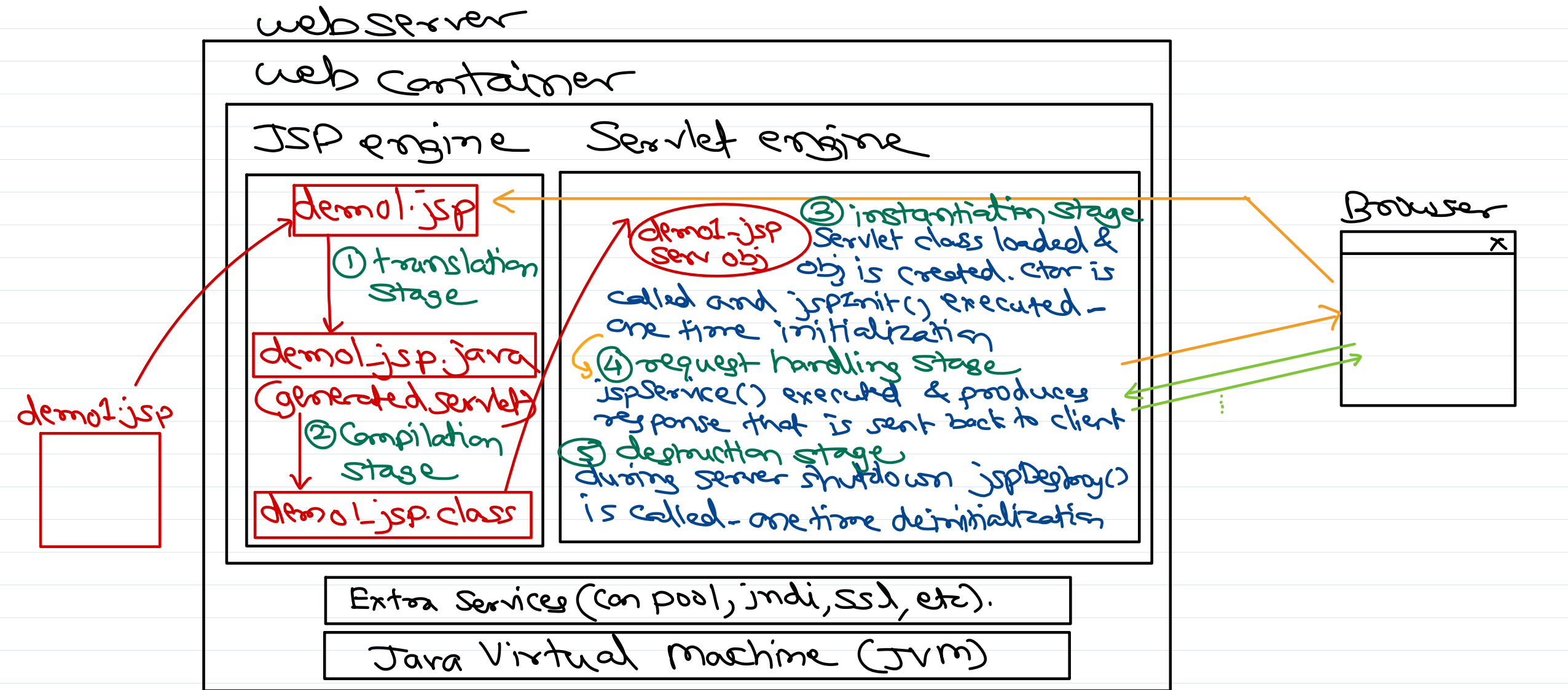

JSP Syntax → HTML and other tags ±...

① Directive : <%@ ... %> } instructions to convert JSP into the servlet class.

② Declaration : <%! ... %> } become part of generated servlet class

③ Scriptlet : <% ... %> } become part of jspService() & executed per req.

④ Expression : <%= ... %>

⑤ Comment : <%-- ... --%> } discarded during servlet conversion.

↗ JSPService() executed per request. It has few local variables a.k.a. JSP Implicit Objects.

① request : HttpServletRequest
② response : HttpServletResponse
③ session : HttpSession
④ config : ServletConfig
⑤ application : ServletContext
⑥ page : Object (this - JSP/serv obj)
⑦ out : JspWriter (like PrintWriter)
⑧ pageContext : PageContext (obj that holds all objs needed for executing JSP and storing page attributes).
⑨ exception : Throwable (obj to represent exception in error JSP pages).

JSP Life Cycle

web server

web Container

JSP engine   Servlet engine

demo1.jsp

① translation Stage

demo1_jsp.java
(generated servlet)

② Compilation Stage

demo1_jsp.class

demo1_jsp Serv obj

③ instantiation stage
Servlet class loaded &
obj is created. ctor is
called and jspInit() executed -
one time initialization

④ request handling stage
jspService() executed & produces
response that is sent back to client

⑤ destruction stage
during server shutdown jspDestroy()
is called - one time deinitialization

demo1.jsp

Browser

x

Extra Services (con pool, jndi, ssl, etc).

Java Virtual Machine (JVM)

JSP syntax

① directives `<%@ ... %>`
  - → page → jsp behavior/setting/config
  - → include → include another page
  - → taglib → using third party tags (spring/JSTL)

⑤ `<%-- Comment --%>`

② declaration
  `<%! ... %>` → JSP class fields & methods.

③ scriptlet
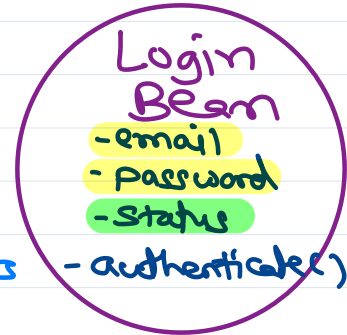 `<% ....java code ...%>` → executed per request

④ expression
  `<%= java expr %>` ---> Output is added to response.

JSP & Java Beans

Java Bean Scopes

① page : PageContext attrs
② request : ServletRequest attrs
③ session : HttpSession attrs
④ application : ServletContext attrs

Login Bean
- email
- password
- status
- authenticate()

demo 05:JSP

email : [____]
Passwd : [____]

Sign In

Login.jsp

if (status)
  welcome
else
  Failed

jsp:useBean internals

```
lb = pageContext.getAttribute("lb");
if (lb == null) {
    lb = new LoginBean();
    pageContext.setAttribute("lb", lb);
```

3

① To create JavaBean and **add in scope:**

`<jsp:useBean id="lb" class="pkg.LoginBean" scope="page" />`

② To set bean fields (by req param or fixed value).

`<jsp:setProperty name="lb" property="password" param="passwd" />`

internally calls setter method ← field/property name
internally calls req.getParameter("....") ← req param name

`<jsp:setProperty name="lb" property="*" />`

auto set all properties/fields of given bean whose name is matching with req parameter.

`<jsp:setProperty name="lb" property="fieldName" value="someValue" />`

i initialize given field with given value.

③ To get bean field value & display in output.

`<jsp:getProperty name="lb" property="email" />`

internally call getter method

④ To execute bean business logic method.

`<%. lb.authenticate(); %>`

Thank you!

Nilesh Ghule <nilesh@sunbeaminfo.com>