UNIT-VI: INTORDUCTION TO JSP

6.1   Problem with Servlet :-
- Thorough knowledge in java programming,
- Lumped Processing of code and the HTML elements.
- No support of markup language for mobile applications using wireless
- Recompilation of Servlet code even for small changes and updates
- Non-usability of advantageous web page development tools.

6.2 Anatomy of JSP page:
Jsp page consists of two parts:
    (i)   jsp elements
    (ii)   Template text.
- Generating dynamic contents using Jsp elements
- Template text can be html,wml,xml or any plain text.
- Merged template text with Jsp elements is send as response to browser.

The sample jsp page look like this
<%@ page language="java" contentType="text/html" %>
```
<html>
<body bgcolor="white">
```
<jsp:usebean id="userInfo" class="com.ora.jsp.bean.userinfo.UserInfoBean">
<jsp:setProperty name="userInfo" property=" *"/>
</jsp:useBean>
```
the following information  was saved :
<ul>
<li>User Name:
```
<jsp:getProperty name="userInfo" property="userName"/>
```
<li>Email Address:
```
<jsp:getProperty name="userInfo" property="emailAddr"/>
```
</ul>
</body>
</html>
```

The bold text represents the Jsp elements.
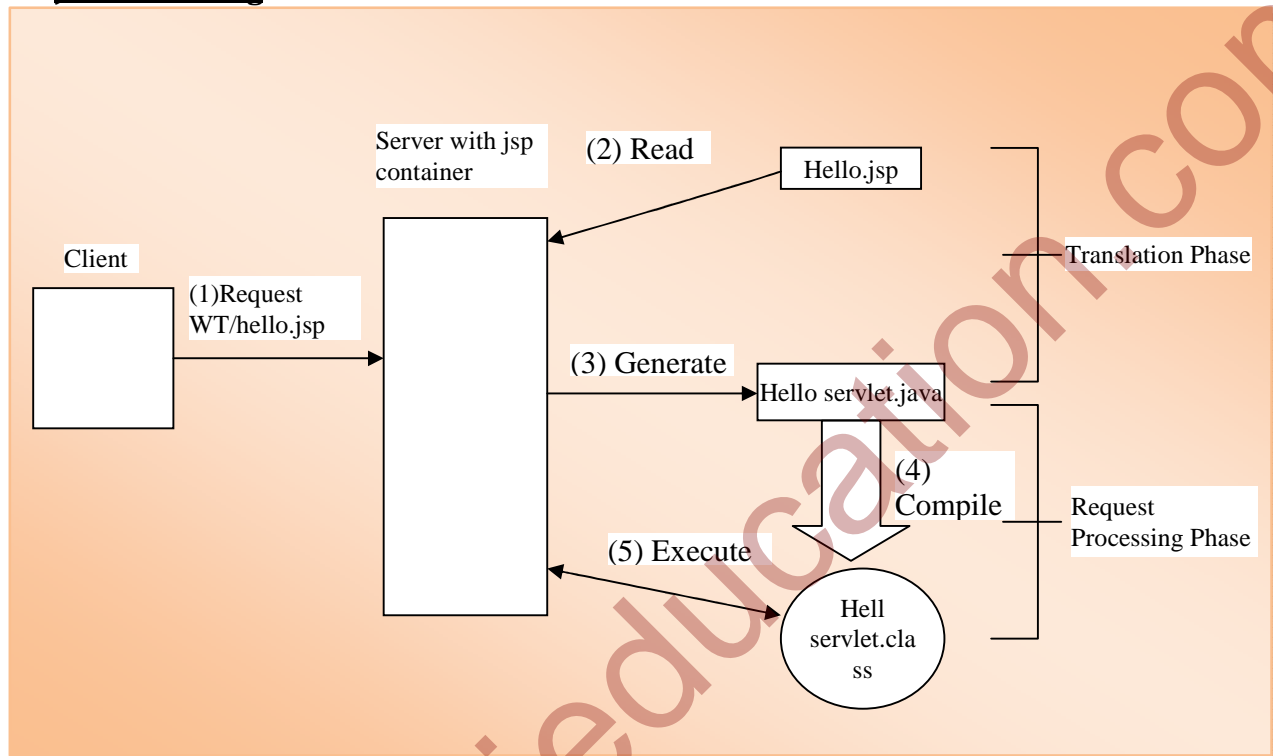 The normal text represents the html elements.

6.3 JSP Processing:
- Processing of jsp page requires, JSP container in web server.
- Jsp container intercepts the requests for jsp pages.

Processing of jsp page is carried out in two phases:
- Translation phase
- Request processing phase

JSP Processing

6.3.1 Translation phase :
- Jsp elements get converted to servlet program.
- Template text to println( ) statements of servlet program.
- Servlet program implements dynamic behavior.
- Compilation of servlet program.

6.3.2 Request processing phase :
- Jsp container invokes jsp page implementation class to process each request .
- Jsp container Generate response.

6.4 JSP ELEMENTS:
There are three types of JSP elements:
1. Directive
2. Action
3. Scripting

6.4.1 Directive elements:
- Specify information of jsp page.
- Checks if session tracking is required.

- Buffering requirements
- Error reporting

Example:-

``<% @ page.....%>	It defines page dependant attributes,such as session tracking,error page,and buffering requirements.
<% @ include......%>	It includes file during translation phase.
<% @ taglib......%>	It declares tag library,containing custom actions,that is used in page.

6.4.2Action elements:-
- Performs action based on the browser request jsp page.
- Accesses instance parameters for database look up.
- Dynamically generate HTML.

Example:-

<jsp:useBean>	It makes javabean component available in jsp page
<jsp:include>	It includes response from servlet, jsp page during request processing phase
<jsp:forward>.	forwards processing of request to servlet or jsp page

6.4.3 Custom action elements and jsp standard tag library(JSTL:
- Develop custom actions.
- JSTL specification bundled with jsp container.
- JSTL action elements performs conditional processing, database access, internationalization.

6.4.4 Scripting elements:-
- Add small piece of java code in jsp page.
- Executed on request of jsp page similar to action element

Example:-

<%..........%>	scriplet,used to embed scripting code.
<%=-------%>	Expression,used to embed scripting code expressions when result shall be added to response;also used as request time action attribute values.
<%!-------%>	Declaration,used to declare ,instance variables and methods in jsp page .

6.4.5 Expression language (EL):-
- JSTL specification contains EL.
- Simple language for accessing requested data.
- Data made available through application classes.
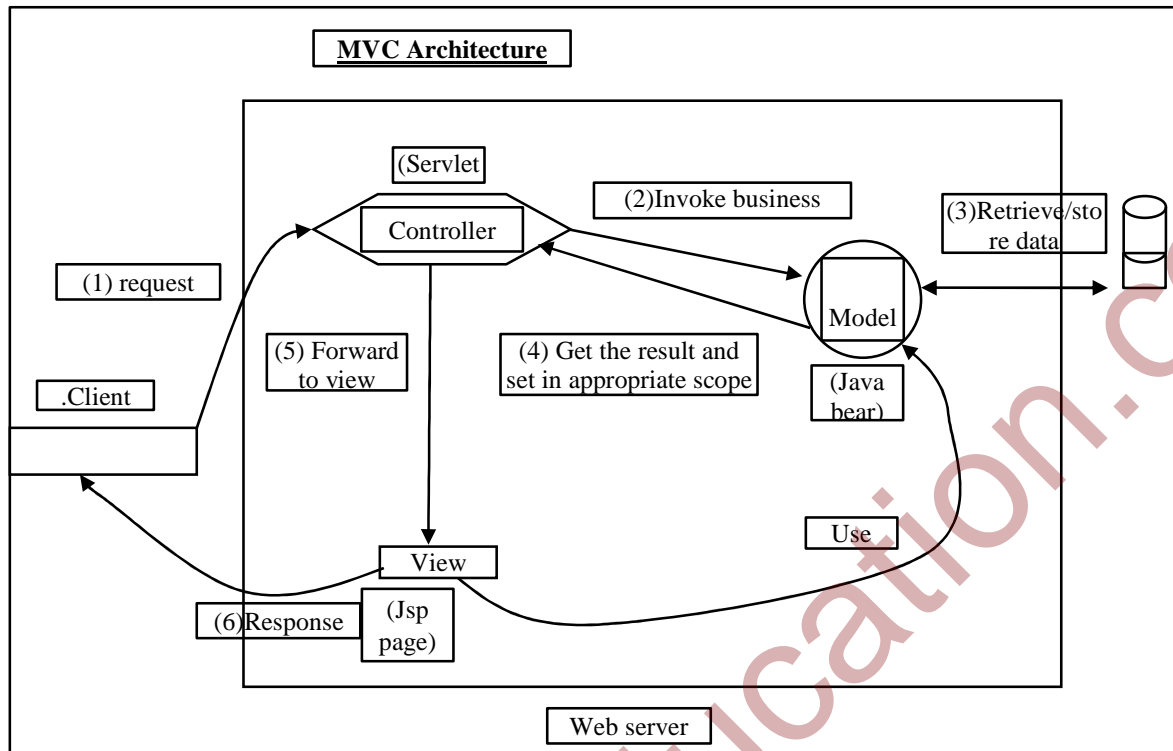
6.4.6 JavaBean components:-
- Work along with action, scripting elements.
- Acts as containers.
- Describe information as entities .

6.5 JSP APPLICATION DESIGN WITH MVC :-
The MVC separates logic in to three distinct units:
1. Model
2. View
3. Controller

1. In MVC, server application classify parts of application as business logic, presentation and request processing.
2. Business logic used for manipulation of application's data such as customer, product and order information.
3. Presentation refers to how an application data displayed to user, for example position, font and size.
4. Finally, request processing is what ties the business logic and presentation parts together.
5. In MVC terms, Model corresponds to business logic and data, view to presentation, and controller to request processing.

MVC Architecture

Diagram labels:
- (Servlet
- Controller
- (2)Invoke business
- (3)Retrieve/store data
- (1) request
- (5) Forward to view
- (4) Get the result and set in appropriate scope
- Model
- (Java bear)
- .Client
- Use
- View
- (6)Response
- (Jsp page)
- Web server

6.6 Installing the Java software development kit

Download java development kit form website http://java.sun.com/j2se.

6.6.1 Setting environment variables under windows NT/2000/XP.

The steps are
1. Open system properties dialog by right clicking on my computer icon and selecting properties option form menu.
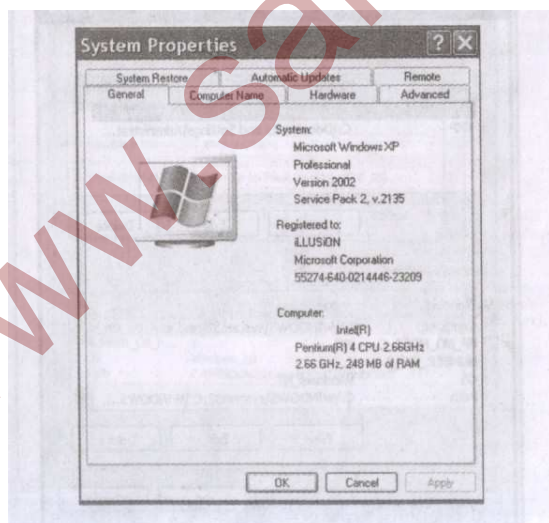
2. Click Advance tab which appears at top of system properties dialog.
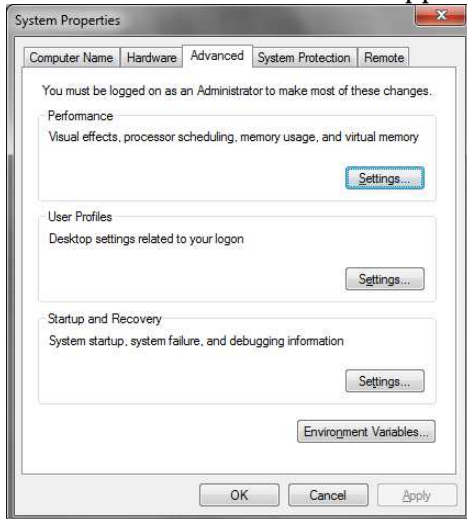
3. Click environment variables button at bottom of advanced tab to display environment variables dialog.
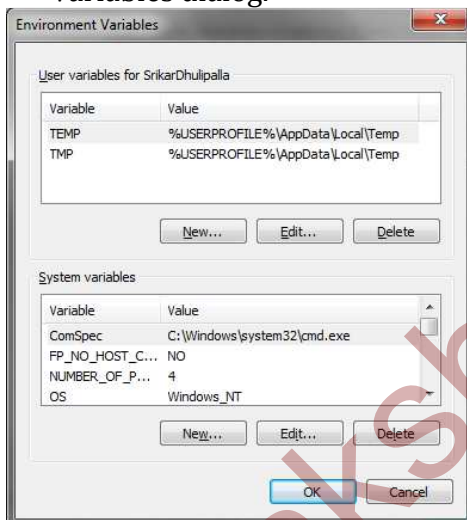
4. (i)click new button in user variables section to crate PATH variable.
   (ii)To set JAVA_HOME variable ,click new button again to display new user variable dialog.select variable name text box and type JAVA_HOME,then select variable value text box and type path of JDK installation directory.finally click ok.

5. Again click ok.

6.6.2 Installing Tomcat server:

Step 1: Download Apache Tomcat Web server from "http://tomcatapache.org"(or any latest version).

Step 2: Double click the "**apache-tomcat-5.5.33.exe"** to begin the installation. The
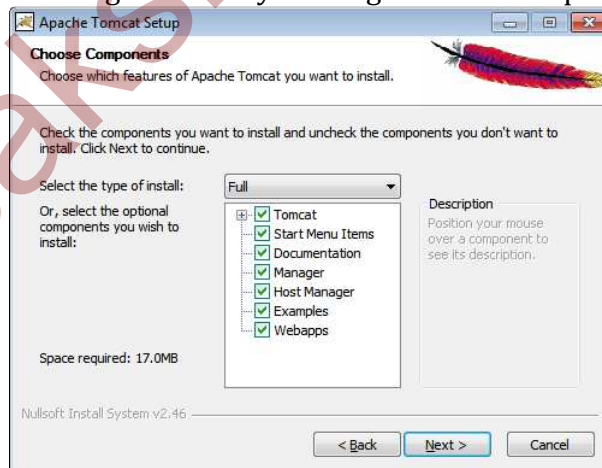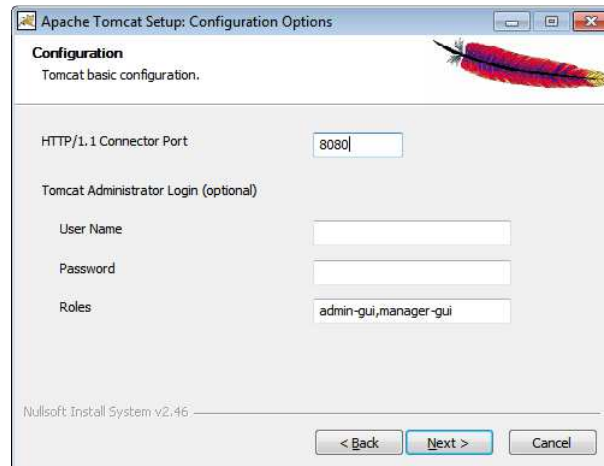
following window appears.
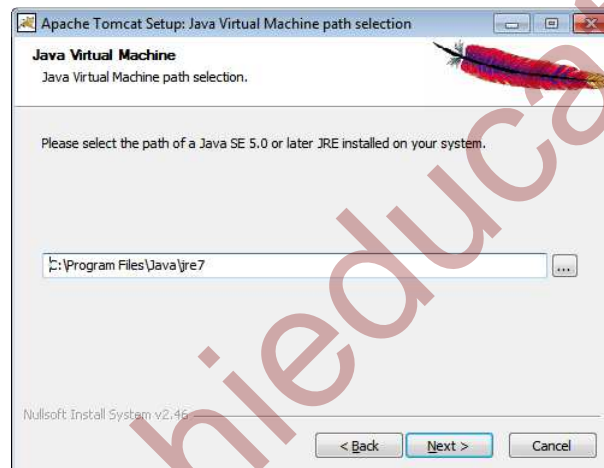

Step 3: Click next to continue
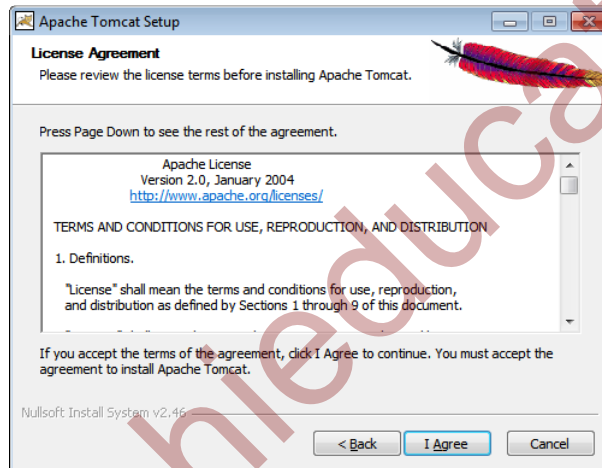
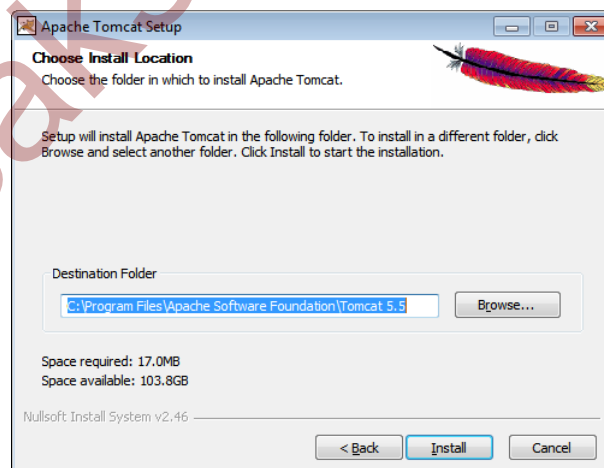**Step 4:** Accept the License Agreement by clicking "I AGREE" to proceed.

Step 5: Choose FULL installation type from the drop down and click next to continue.

Step 6: Assign the port number **8080** in the **HTTP/1.1 Connector Port** field and click next to continue.

Step 7: Provide the **Java Runtime Environment** Path. The installer by default assigns the path if any.

Step 8: Provide the path where the setup directories are to be stored. The default path is provided by the installer that can be modified accordingly. Click INSTALL to begin the installation process.
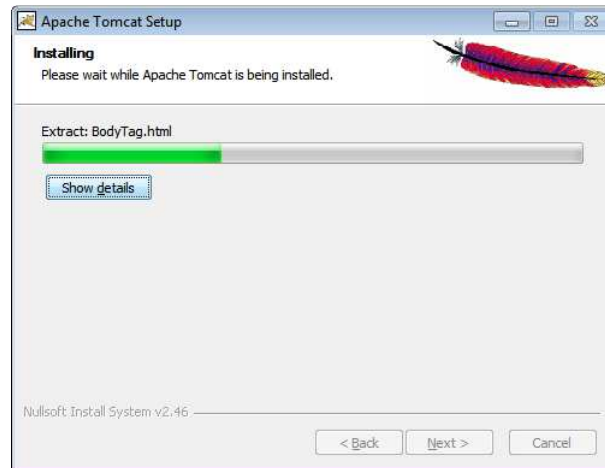
FIG: Commencing and processing of Apache Tomcat Setup

Step 10: Once the installation processing completes, a window is displayed notifying the successful installation of the web server and also provides two options (Run Apache Tomcat & Show Readme). Click FINISH to start the server and complete the installation.
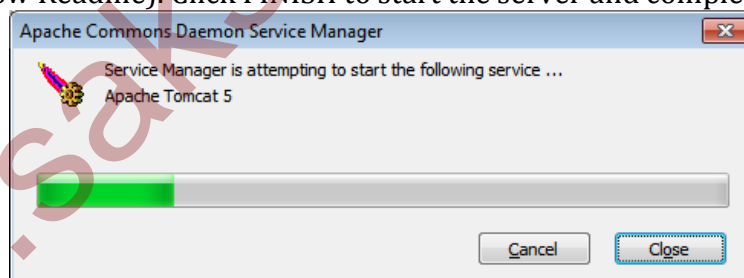

FIG: Commencing Of the Apache Tomcat 5 Web Server

6.6.3 Testing the tomcat:-

To test server run start up script as described in platform specific sections and open browser and enter this URL in location address field:http://loclahost:8080/ then welcome page
of tomcat shown on browser

Tomcat installation directory contains number of subdirectories described inREADME.txt file, but most important ones are:

bin
Scripts for starting and stopping of tomcat server.

conf
Tomcat configuration files.

webapps
Default location for web applications served by tomcat. Two more subdirectories under tomcat home directory are created on start of server.

logs
Server log files. If something doesn't work as expected, look into files in this directory for clues as to know what's wrong.

Work
A directory for temporary files is created by jsp container and other files. This directory stores servlets generated from jsp page.