

Assignment No. : 1

TITLE

Design and develop any suitable static web application using HTML and CSS and Bootstrap. Use and configuration of Web Server. (Apache Tomcat)

OBJECTIVES

1. Understand how to install tomcat server and set up environment variables.

PROBLEM STATEMENT

- (i) Installation of Apache TOMCAT Server and Configuration of it.
- (ii) Develop a static website/application using HTML, CSS and Bootstrap.(Student must mention the static website/application name)

OUTCOME

Students will be able to,

1. Install TOMCAT Server and Configuration of it.
2. Design UI interface for web applications

SOFTWARE & HARDWARE REQUIREMENTS

1. Java 7 or Later
2. Apache Tomcat Server

THEORY-CONCEPT

Web Application:

A web application runs over the Internet. Ex. eBay, Amazon, Google, facebook etc

A webapp contains five components:

1. HTTP Server: Examples are- Google Web Server, Apache HTTP Server, Apache Tomcat Server, Microsoft Internet Information Server (IIS) etc
2. HTTP Client (Web Browser): Examples are- Internet Explorer, Firefox, Google Chrome, Safari etc.
3. Database: Examples are- MySQL, Apache Derby, MS SQL Server, SQLite, PostgreSQL, Commercial Oracle, IBM DB2, SAP SyBase, MS Access etc
4. Client-Side Programs: It can be written in HTML Form, VBScript, JavaScript, Flash etc.
5. Server-Side Programs: could be written in Java Servlet/JSP, ASP, PHP, Perl, Python, CGI, and others.

A web app is 3-tier (or multi-tier) client-server database application which run over the Internet as shown in the following diagram,

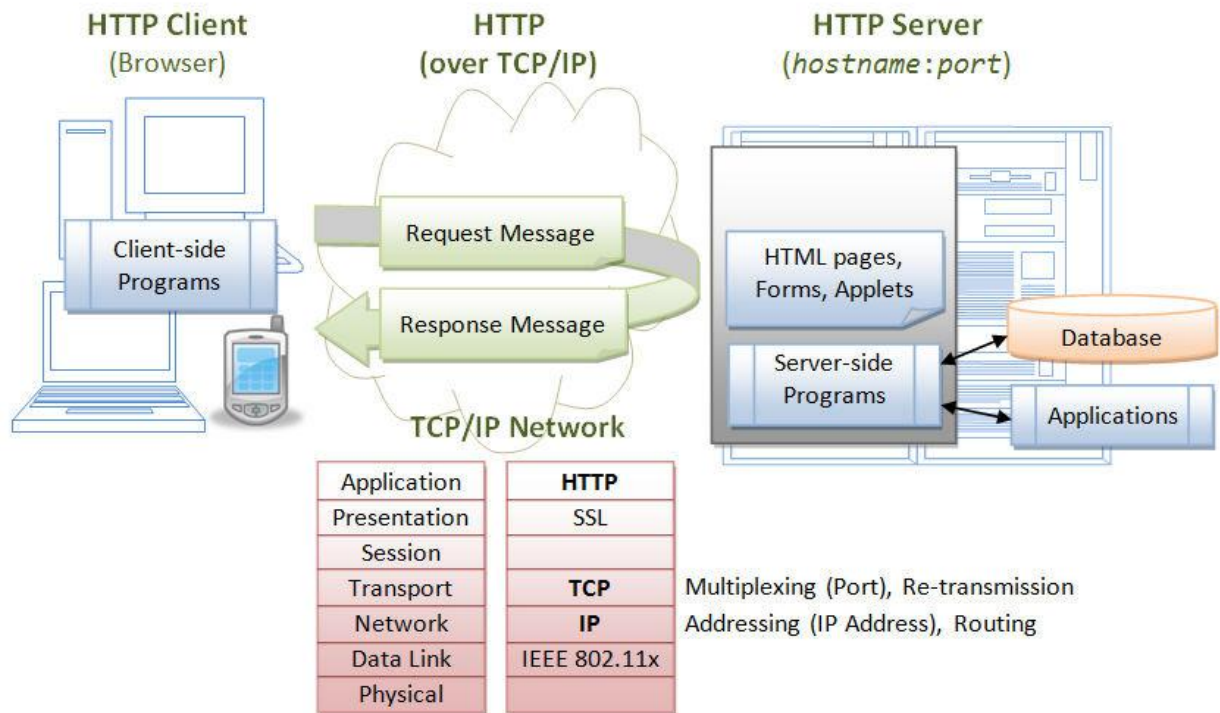


Figure.1: Three-Tier Client-Server Architecture

1. To start a webapp, A user, issues a URL request via a web browser (HTTP client), to HTTP server.
2. The HTTP server returns an HTML form (client-side program), which is loaded into the client's browser.
3. The user fills up the query data inside the form and submits that form.
4. The client-side program sends the query parameters to a server-side program.
5. The server-side program receives the query parameters, queries the database based on these parameters, and returns the query result to the client-side program.
6. The client-side program displays the query result on the browser.
7. The process repeats for the next request.

Apache tomcat

Tomcat is an open-source project, under the “Apache Software Foundation” (which also provides the most use, open-source, industrial-strength Apache HTTP Server). The mother site for Tomcat is <http://tomcat.apache.org>. Alternatively, you can find tomcat via the Apache mother site: <http://www.apache.org>.

EXECUTION STEPS

Step 1: Installing Java 8

1. Before installing Tomcat make sure you have the latest version of **Java Development Kit (JDK)** installed and configured on the system. It is preferred to use oracle Java.

To install latest Oracle Java JDK (jdk-8u131) on Linux, you may like to refer our recent posts on Oracle jdk/jre/jar installations here:

- [Install Java 8 JDK on Linux](#)
- [Install Java 8 JDK/JRE on RHEL/CentOS](#)

Step 2: Download and Install Apache Tomcat 8

2. Once latest Java installed and configured correctly on the system, we will move forward to download and install latest stable version of Tomcat 8 (i.e. 8.5.14). If you want to cross check, if any newer version available, go to following Apache download page and cross check.

- <http://tomcat.apache.org/download-80.cgi>

3. Next create a **/opt/tomcat/** directory and download the latest version of Apache Tomcat 8 under this directory, also for cross checking the download file, we will be downloading hash file. The download will take some time depending upon your connection speed.

```
# mkdir /opt/tomcat/ && cd /opt/tomcat
# wget http://mirror.fibergrid.in/apache/tomcat/tomcat-8/v8.5.14/bin/apache-tomcat-8.5.14.zip
# wget https://www.apache.org/dist/tomcat/tomcat-8/v8.5.14/bin/apache-tomcat-8.5.14.zip.md5
```

Note: Make sure to replace the version number in the above download link with the latest version available if it was different.

4. Now verify the MD5 Checksum against the key.

```
# cat apache-tomcat-8.5.14.zip.md5
```

```
# md5sum apache-tomcat-8.5.14.zip
```

Make sure that the output (Hash Value) matches, as shown below.

```
[root@tecmint opt]# cat apache-tomcat-8.5.14.zip.md5
e84f5516f2de7dd0a27faabff26b59be *apache-tomcat-8.5.14.zip[root@tecmint opt]#
[root@tecmint opt]#
[root@tecmint opt]# md5sum apache-tomcat-8.5.14.zip
e84f5516f2de7dd0a27faabff26b59be  apache-tomcat-8.5.14.zip
[root@tecmint opt]# _
```

5. Extract the Tomcat zip and cd to ‘apache-tomcat-8.5.14/bin/’ directory.

```
# unzip apache-tomcat-8.5.14.zip
```

```
# cd apache-tomcat-8.5.14/bin/
```

6. Now make Linux scripts executable that is under ‘apache-tomcat-8.5.14/bin/’ and then create symbolic link of **startup** and **shutdown** script for tomcat as:

Change all scripts *.sh executable only for root as,

```
# chmod 700 /opt/tomcat/apache-tomcat-8.5.14/bin/*.sh
```

Create Symbolic link for startup script as,

```
# ln -s /opt/tomcat/apache-tomcat-8.5.14/bin/startup.sh /usr/bin/tomcatup
```

Create Symbolic link for shutdown script as,

```
# ln -s /opt/tomcat/apache-tomcat-8.5.14/bin/shutdown.sh /usr/bin/tomcatdown
```

7. Now to start tomcat, you just need to fire the below command as root from anywhere in the shell.

```
# tomcatup
```

Sample Output

```
Using CATALINA_BASE: /opt/tomcat/apache-tomcat-8.5.14
```

```
Using CATALINA_HOME: /opt/tomcat/apache-tomcat-8.5.14
```

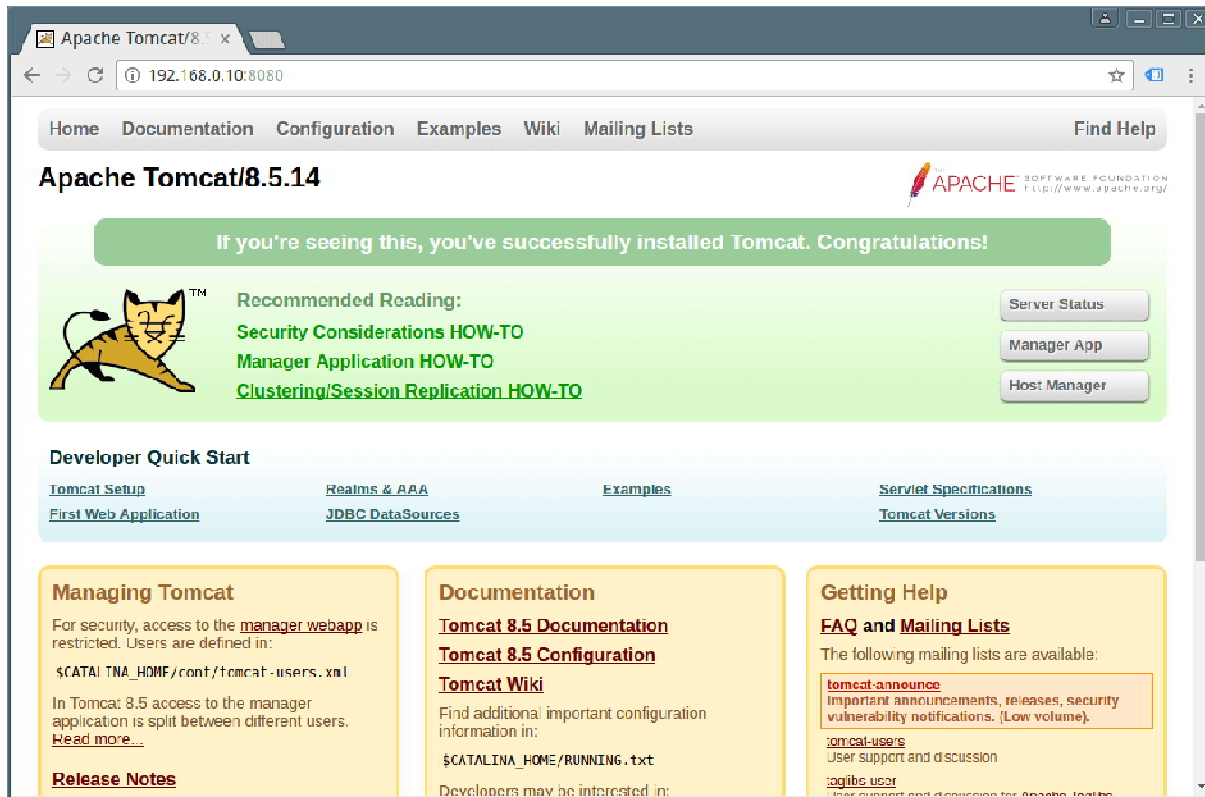
```
Using CATALINA_TMPDIR: /opt/tomcat/apache-tomcat-8.5.14/temp
```

```
Using JRE_HOME: /opt/java/jdk1.8.0_131/jre/
```

```
Using CLASSPATH: /opt/tomcat/apache-tomcat-8.5.14/bin/bootstrap.jar:/opt/apache-
tomcat-8.5.14/bin/tomcat-juli.jar
```

Tomcat started.

Once ‘Tomcat Started’, you can point your browser to **http://127.0.0.1:8080** and you should see something as:



Step 3: Changing Apache Tomcat Port

8. Do remember that 8080 is the default port tomcat usages. To change tomcat port from 8080 to any other unused port (say 137), you may do as:

First shutdown tomcat server simply by typing the below command from anywhere in the shell.

```
# tomcatdown
```

Next, open ‘/opt/tomcat/apache-tomcat-8.5.14/conf/server.xml’ file in your favorite editor (mine is nano) to edit. Change port 8080 to 137, save and exit.

```
# nano /opt/tomcat/apache-tomcat-8.5.14/conf/server.xml
```

Sample Output

```
Connector port="137" protocol="HTTP/1.1"
connectionTimeout="20000"
redirectPort="8443" />;
```

After changing port to 137, restart tomcat service again and point your browser to **http://127.0.0.1:137**.

tomcatup

Similarly, you can use any port of your choice. Make sure the port you are using don't conflict with any other application/resource.

Step 4: Configuring Apache Tomcat 8

9. By default the Tomcat page is accessed by you only due its security implementation so that unauthorized users don't have access to it.

To access admin and other sections like Server Status, Manager App and Host Manager. You need to add user accounts for admins and managers.

To add a Tomcat user edit file `/opt/tomcat/apache-tomcat-8.5.14/conf/tomcat-users.xml`, in your favorite editor.

nano /opt/tomcat/apache-tomcat-8.5.14/conf/tomcat-users.xml

10. Add the following lines just before `</tomcat-users>`.

```
<role rolename="admin-gui"/>
<user username="admin"; password="tecmint" roles="admin-gui"/>
<role rolename="manager-gui"/>
<user username="avi" password="tecmint123" roles="manager-gui"/>
```

```
File Edit View Search Terminal Help
GNU nano 2.4.1 File: ...apache-tomcat-8.0.23/conf/tomcat-users.xml Modified

-->
<!--
<role rolename="tomcat"/>
<role rolename="role1"/>
<user username="tomcat" password="tomcat" roles="tomcat"/>
<user username="both" password="tomcat" roles="tomcat,role1"/>
<user username="role1" password="tomcat" roles="role1"/>
-->
<role rolename="admin-gui"/>
<user username="admin" password="tecminf" roles="admin-gui"/>

<role rolename="manager-gui"/>
<user username="avi" password="tecminf123" roles="manager-gui"/>

</tomcat-users>

^G Get Help      ^O Write Out     ^W Where Is     ^K Cut Text     ^J Justify
^X Exit          ^R Read File     ^\ Replace      ^U Uncut Text   ^T To Spell
```

Add Tomcat Users

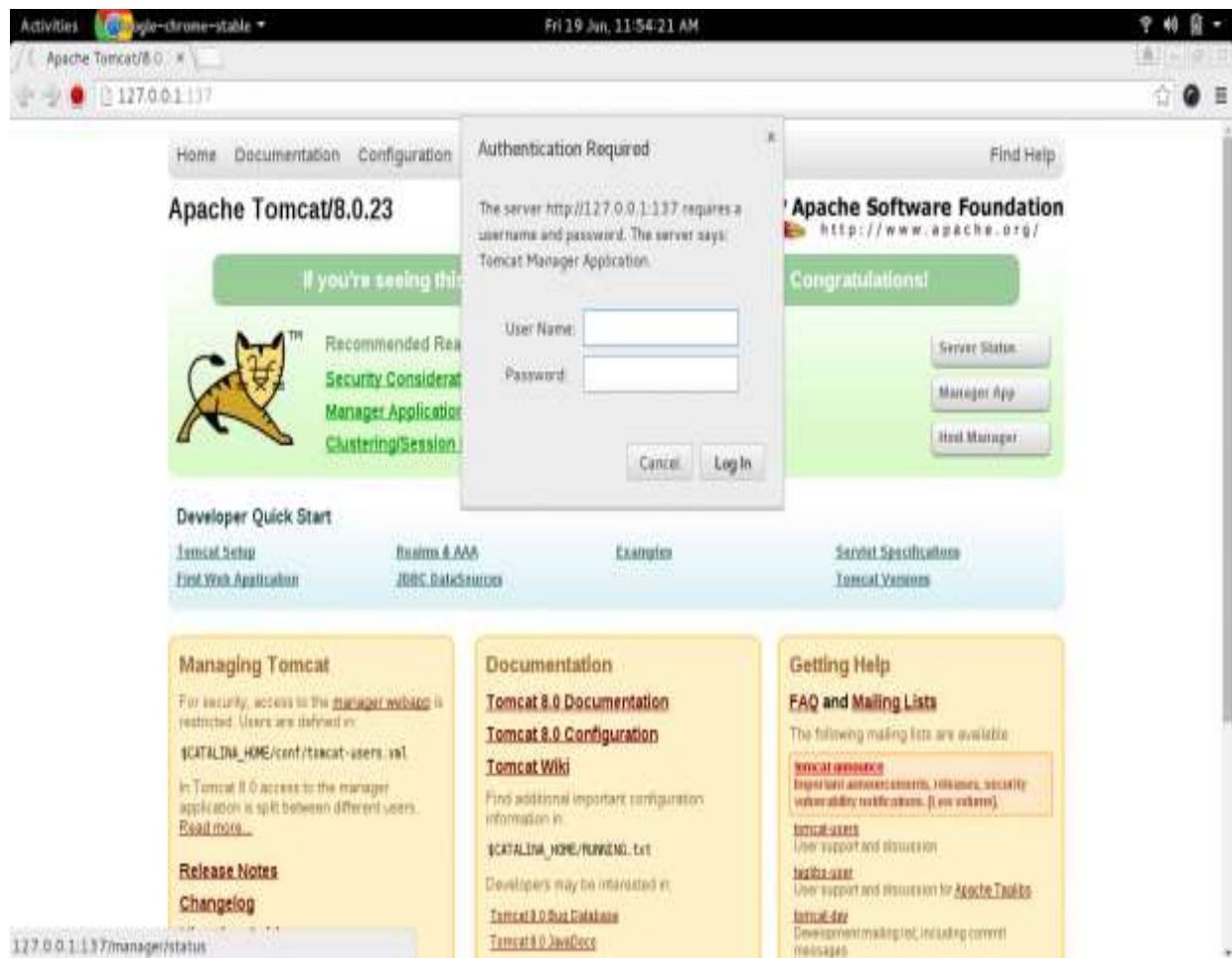
Again, you need to restart Tomcat to take new changes into effect.

tomcatdown

tomcatup

11. After restarting Tomcat, make sure to access the admin other sections like Server Status, etc at <http://127.0.0.1:137>.

It you will be asked to Enter User_name and Password, you just created above, after login you will see something like the below interface.



Apache Tomcat User Login

The screenshot shows the Apache Tomcat Manager web interface in a Google Chrome browser. The browser's address bar shows the URL `http://127.0.0.1:8080/manager/status`. The page features the Apache Software Foundation logo and a cartoon cat. The main heading is "Server Status". Below this, there is a "Manager" section with links: "List Applications", "HTML Manager Help", "Manager Help", and "Complete Server Status". The "Server information" section contains a table with the following data:

Tomcat Version	JVM Version	JVM Vendor	OS Name	OS Version	OS Architecture	Hostname	IP Address
Apache Tomcat/8.0.23	1.8.0_45-b14	Oracle Corporation	Linux	3.18.3-4-amd64	amd64	deb	127.0.1.1

Below the server information, the "JVM" section displays memory statistics: "Free memory: 82.98 MB Total memory: 135.50 MB Max memory: 1758.50 MB". This is followed by a table of memory pools:

Memory Pool	Type	Initial	Total	Maximum	Used
PS Eden Space	Heap memory	31.00 MB	62.00 MB	649.00 MB	20.26 MB (3%)
PS Old Gen	Heap memory	83.00 MB	68.50 MB	1319.00 MB	22.34 MB (1%)
PS Survivor Space	Heap memory	5.00 MB	5.00 MB	5.00 MB	0.00 MB (0%)
Code Cache	Non-heap memory	2.43 MB	7.68 MB	240.00 MB	6.86 MB (2%)
Compressed Class Space	Non-heap memory	0.00 MB	2.62 MB	1024.00 MB	2.46 MB (0%)
Metaspace	Non-heap memory	0.00 MB	23.87 MB	-0.00 MB	23.30 MB

The "ajp-nio-8009" section shows connection statistics: "Max threads: 250 Current thread count: 0 Current thread busy: 0 Keeped alive sockets count: 0 Max processing time: 0 ms Processing time: 0.0 s Request count: 0 Error count: 0 Bytes received: 0.00 MB Bytes sent: 0.00 MB".

Apache Tomcat Server Status

Your all setting and basic configuration of Tomcat has finished now. Now we will see a simple example (beyond the scope of this article), a very basic program (JSP Program) to run in Tomcat. It would be dumb if we do not show it.

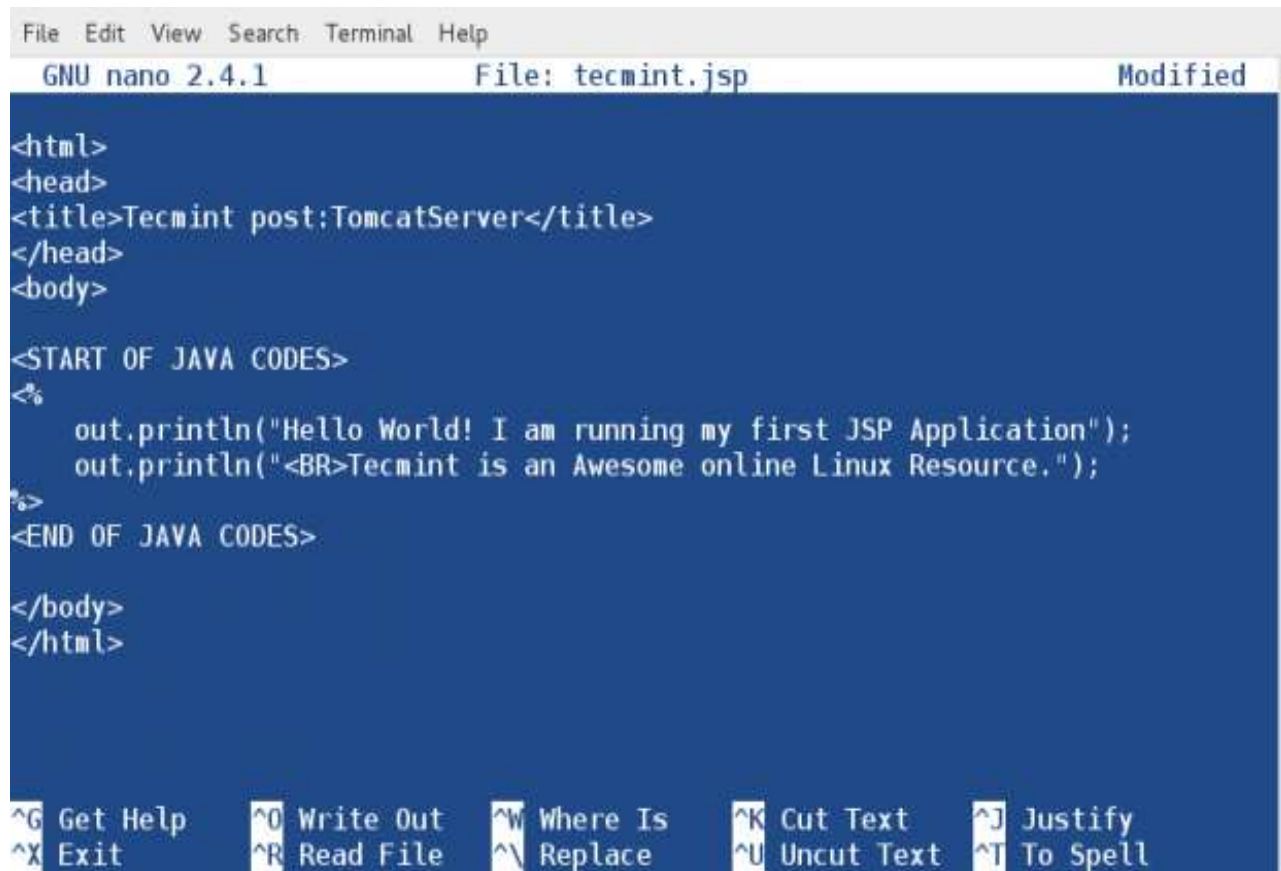
12. Create a file called `tecmint.jsp` under `/opt/tomcat/apache-tomcat-8.5.14/webapps/ROOT` directory.

```
# touch /opt/tomcat/apache-tomcat-8.5.14/webapps/ROOT/tecmint.jsp
```

13. Now put the below contents in the new file (`tecmint.jsp`), save and exit. You may edit it, if

you know what you are doing.

```
<html>
<head>
<title>Tecmint post:TomcatServer</title>
</head>
<body>
<START OF JAVA CODES>
<%
out.println("Hello World! I am running my first JSP Application");
out.println("<BR>Tecmint is an Awesome online Linux Resource.");
%>
<END OF JAVA CODES>
</body>
</html>
```



```
File Edit View Search Terminal Help
GNU nano 2.4.1 File: tecmint.jsp Modified

<html>
<head>
<title>Tecmint post:TomcatServer</title>
</head>
<body>

<START OF JAVA CODES>
<%
    out.println("Hello World! I am running my first JSP Application");
    out.println("<BR>Tecmint is an Awesome online Linux Resource.");
%>
<END OF JAVA CODES>

</body>
</html>

^G Get Help  ^O Write Out  ^W Where Is  ^K Cut Text  ^J Justify
^X Exit      ^R Read File  ^\ Replace   ^U Uncut Text ^T To Spell
```

Create JSP Program

Most of the things in the above codes are self explaining. We have put simple Java code to print two lines of output and embed it in between HTML codes, which can be accessed at <http://127.0.0.1:137/tecmint.jsp>.



JSP Page

(Students must include all the below points in your **write up**)

- 1) Introduction to Web technology
- 2) Structure of HTML Document
- 3) HTML Elements: Head, Paragraph, Line Break, Colors, Fonts, Links, Frames, Lists, Tables, Images, Forms, Inputs with Example
- 4) CSS Introduction
- 5) Types of CSS with syntax and Example.
- 6) CSS Selectors
- 7) Bootstrap
- 8) Flowchart of the application which you have selected to develop.

CONCLUSION / ANALYSIS

Hence, we have learned how to install and configure Apache Tomcat server.

(Also mention the conclusion for static web application development)

Output: Code and Screen Shots of designed Website (attach print out of input code and output)

Date of Performance:

Date of Completion:

Date of Submission: Faculty Sign

Grade/Marks