Tomcat Scenario Task:

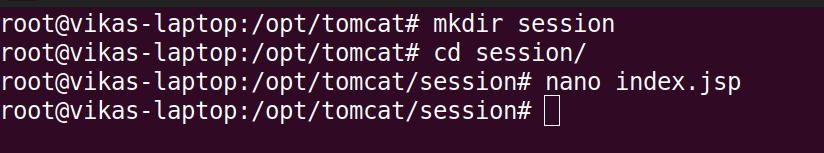
Create a simple web application that uses session management to track the number of visits by a user.

# SESSION MANAGEMENT:

Session management in Tomcat is a way for the server (Tomcat) to keep track of individual users' interactions with a web application. It allows the server to maintain specific information about each user, such as login status, preferences, or data that needs to be remembered during their visit.

STEP1:

Create a folder and index.jsp file inside tomcat folder



STEP 2:

Add this code inside the index.jsp

<%@ page language="java" contentType="text/html; charset=UTF-8"

pageEncoding="UTF-8" %>

<%@ page session="true" %>

<!DOCTYPE html>

<html>

<head>

<meta charset="UTF-8">

<title>Session Management Demo</title>

</head>

<body>

<%

Integer visitCount = (Integer) session.getAttribute("visitCount");

if (visitCount == null) {

visitCount = 1;

} else {

visitCount++;

}

session.setAttribute("visitCount", visitCount);

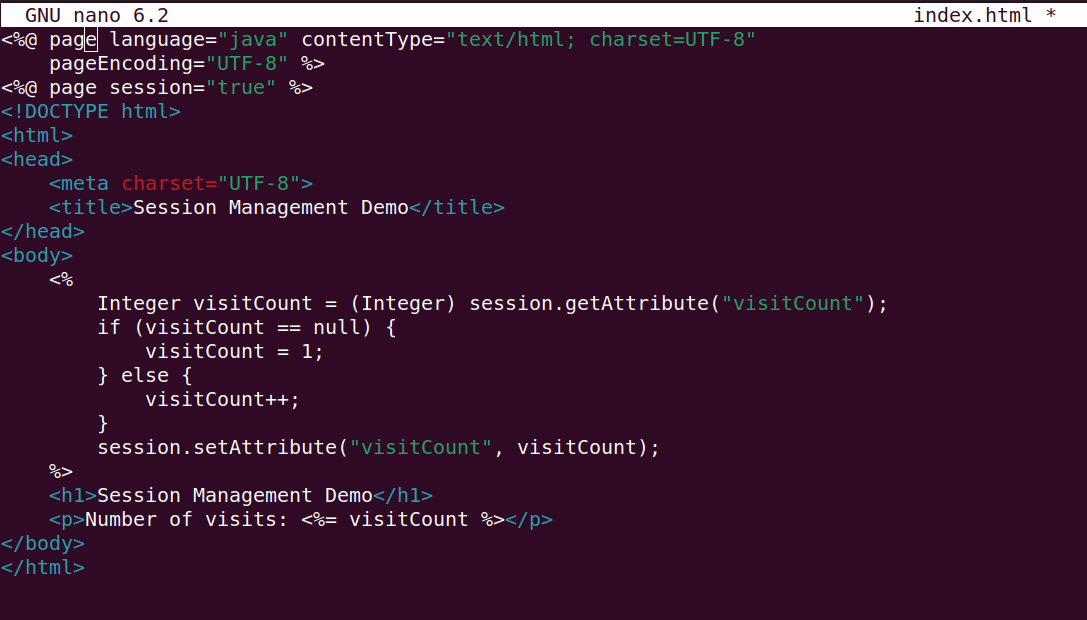
%>

<h1>Session Management Demo</h1>

<p>Number of visits: <%= visitCount %></p>

</body>

</html>



STEP 3:

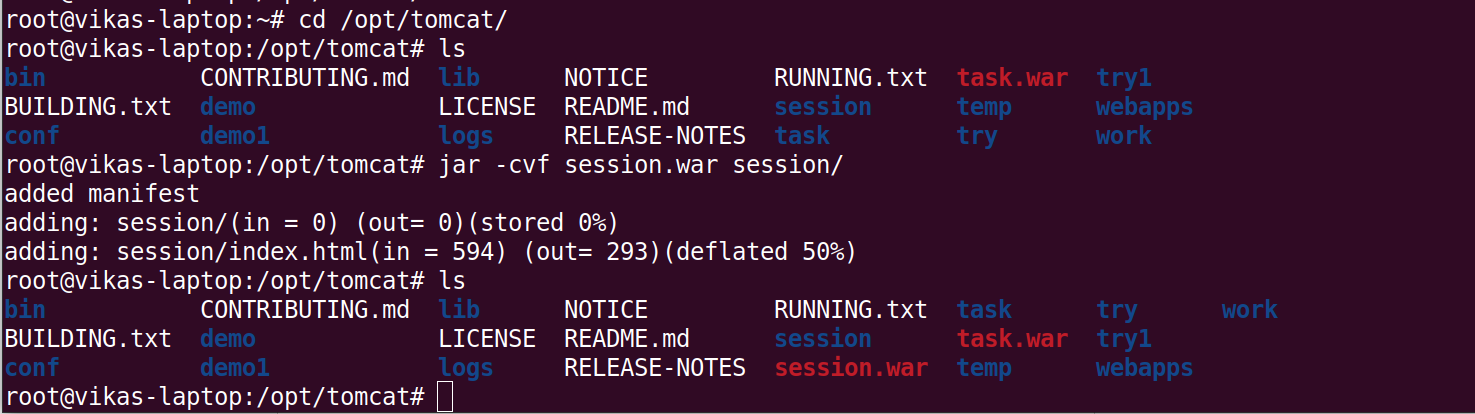
Create a war file of this session directory and move it inside webapps ,which will automatically extract the directory.

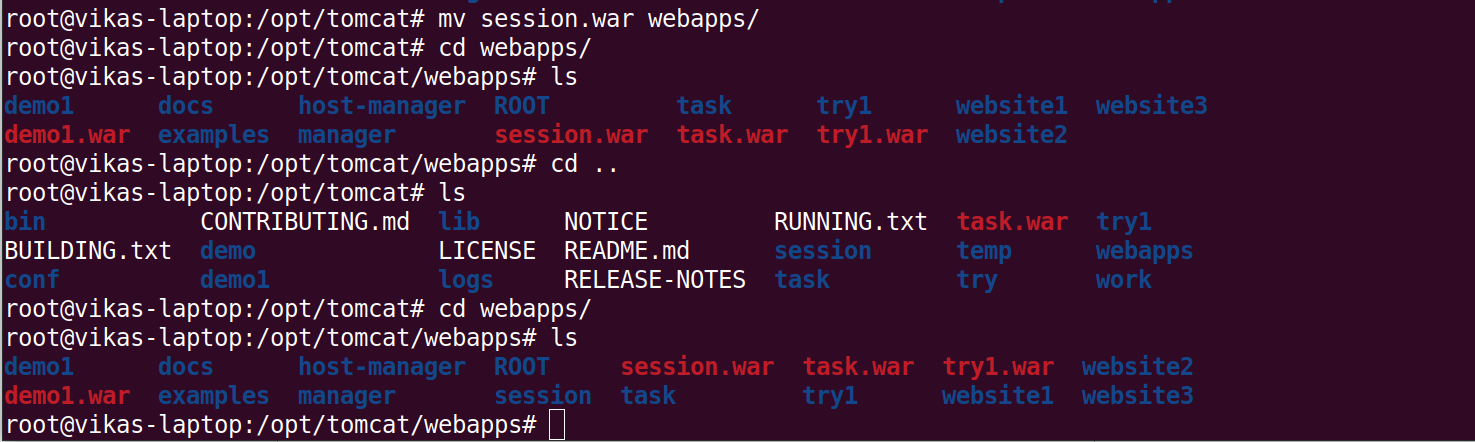
→cd /opt/tomcat

→jar -cvf session.war session

→mv session.war webapps

- Also move the index.jsp file which is from the extracted war file, directly inside the session folder which is created automatically.





STEP 4:

- Open the server.xml file and configure the path for the new website.

→nano /opt/tomcat/conf/server.xml

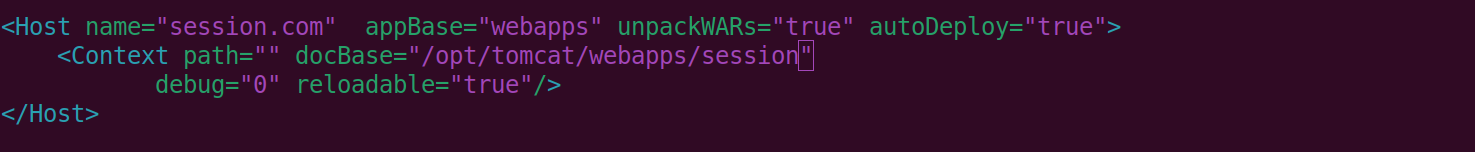
- Add the following lines to set the appropriate path, inside the server.xml

<Host name="session.com" appBase="webapps" unpackWARs="true" autoDeploy="true">

<Context path="" docBase="/opt/tomcat/webapps/session"

debug="0" reloadable="true"/>

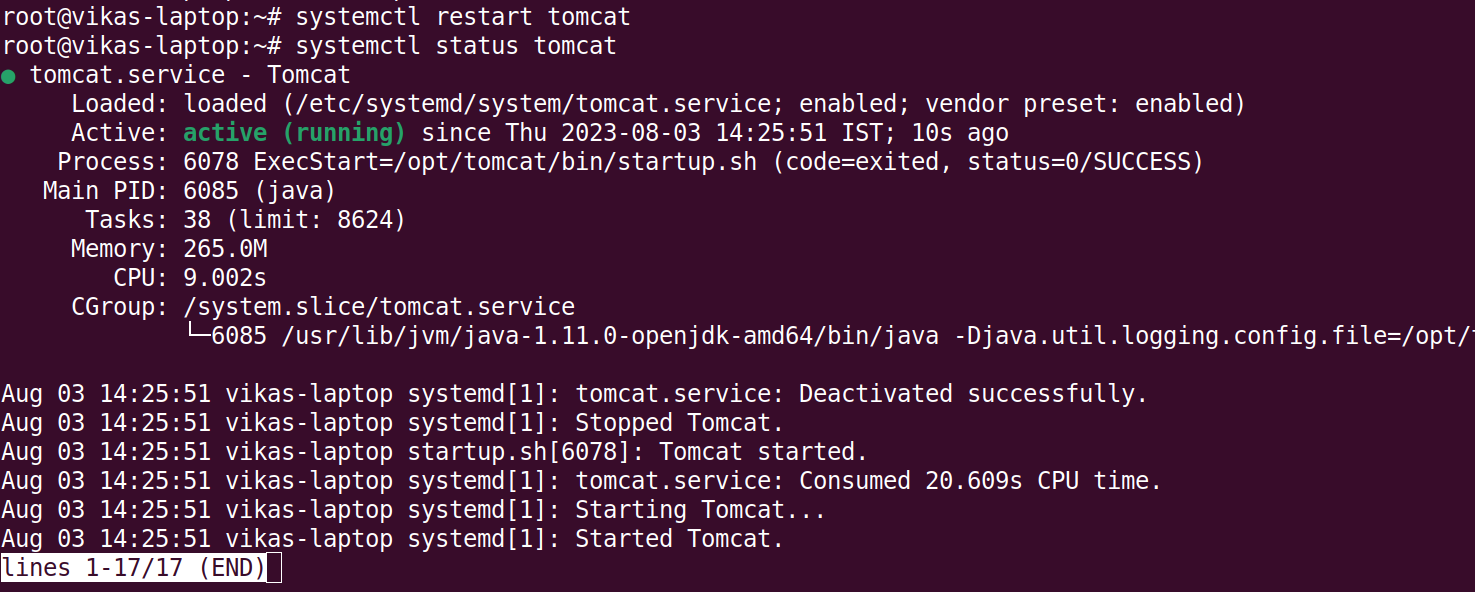
</Host>



- Save and exit

STEP 5:

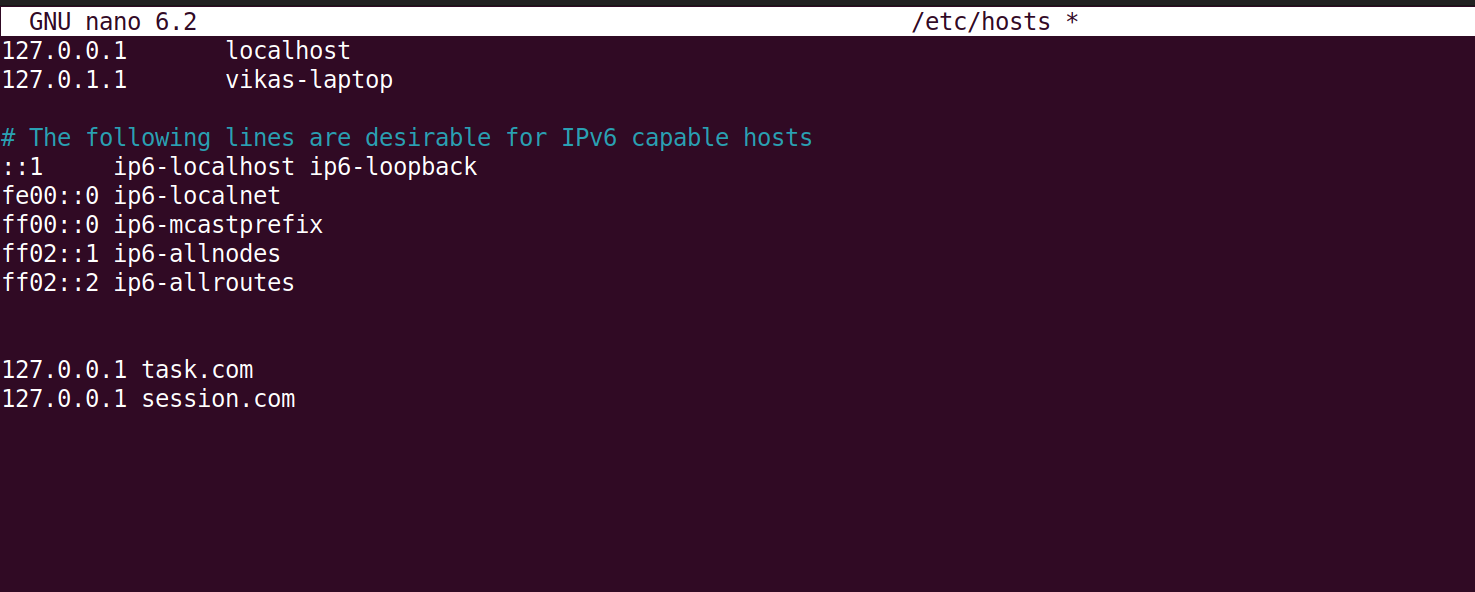
Restart tomcat after updating the server.xml file



STEP 6:

Open hosts file and add your machine ip and assign session.com to it as DNS name.

→ nano /etc/hosts



STEP 7:

Type session.com:8080 in the browser and then we will get the required page,refresh many times ,and the number of visits gets added up,this shows the session object is working.

