**PRACTICAL-1**

**Target : Installation of APACHE TOMCAT Web Server**

Step 1: Download Tomcat for Windows

Browse to the [official Apache Tomcat website](https://tomcat.apache.org/). Locate the Download section and click the **latest Tomcat version** available. At the time of writing this article, the latest Tomcat version was version 10.

On the Download page, scroll down and locate the Binary Distributions area.

In the Core list, depending on the installation type you prefer, click the download link for the **Windows Service Installer** or the **32bit**/**64bit Windows zip file**.

The next step configures the Tomcat server. For instance, enter the **Administrator login credentials** or choose a different **connection port**. When finished, click **Next** to proceed to the next step.

Choose the Tomcat server install location or keep the default one and click **Install**.

A popup window appears that starts the Tomcat service. After the process completes, the window closes automatically. The Apache Tomcat web server is now successfully installed .

Navigate to the conf sub-directory within the extracted directory and locate the **server.xml** file.

The default connection port is **8080**. To choose a different port, edit the **server.xml** file with a text editor, such as Notepad++, and locate the following lines:

<Connector port="8080" protocol="HTTP/1.1"

connectionTimeout="20000"

redirectPort="8443" />

Change the **connector port** number to any number between **1024** and **65535**.

To [enable directory browsing](https://phoenixnap.com/kb/403-forbidden#ftoc-heading-9), locate the **web.xml** file in the conf directory and edit the file with a text editor. Directory browsing helps when testing the system, and sometimes it may be the [solution for a 403 forbidden error](https://phoenixnap.com/kb/403-forbidden).

Locate the following lines and change the **listings** value from **false** to **true**:

<servlet>

<servlet-name>default</servlet-name>

<servlet-class>org.apache.catalina.servlets.DefaultServlet</servlet-class>

<init-param>

<param-name>debug</param-name>

<param-value>0</param-value>

</init-param>

<init-param>

<param-name>listings</param-name>

<param-value>false</param-value>

</init-param>

<load-on-startup>1</load-on-startup>

</servlet>

Implement an auto-reload feature by editing the **context.xml** file. Above all, auto-reload is useful in development to prevent restarting the server manually each time a change is made.

Using a text editor, open the **context.xml** file. Locate the following line and change the value from **false** to **true** in each instance:

<Context reloadable="false" crossContext="false" parallelAnnotationScanning="false">

......

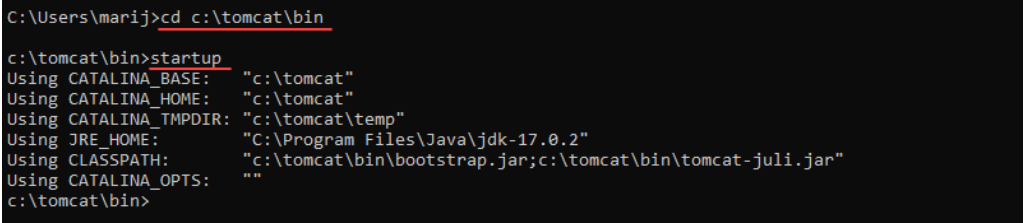
......

</Context>

After making the changes, start the server. Press the **Windows key** and type cmd. Press **Enter** to open a Command Prompt window.

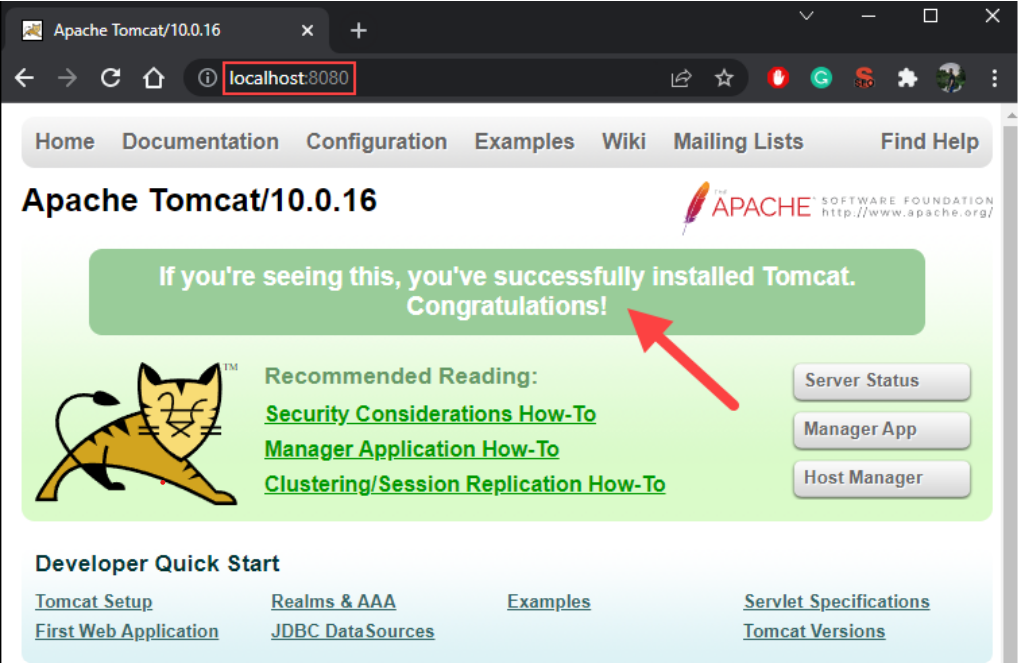
Move to the *bin* directory of your Tomcat server and run:

Startup



Access the server using a browser as an HTTP client. Browse to http://localhost:8080 and access the Tomcat welcome page to ensure the server works.

In addition, use the Developer Quick Start links to see more information about the server and start using and configuring the server.

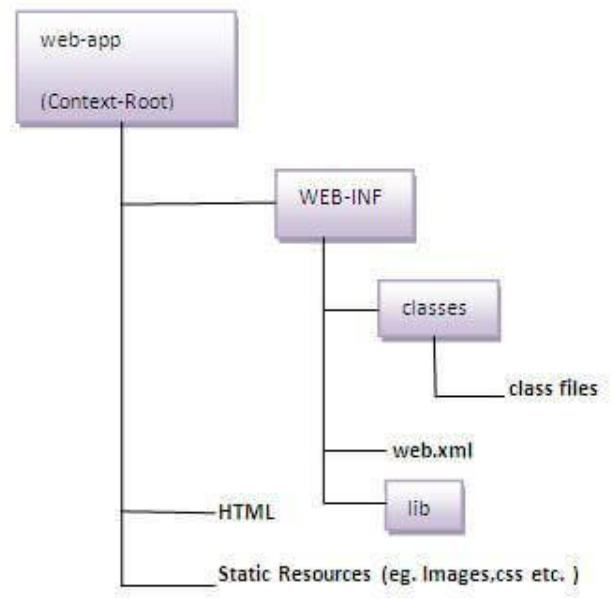


**PRACTICAL-2**

**Target : Folder Structure of Servlet**

The **directory structure** defines that where to put the different types of files so that web container may get the information and respond to the client.

The Sun Microsystem defines a unique standard to be followed by all the server vendors. Let's see the directory structure that must be followed to create the servlet.



## The Root Directory

The root directory of you web application can have any name. In the above example the root directory name is mywebapp,

Under the root directory, you can put all files that should be accessible in your web application.

## he WEB-INF Directory :

The WEB-INF directory is located just below the web app root directory. This directory is a meta information directory. Files stored here are not supposed to be accessible from a browser (although your web app can access them internally, in your code).

## web.xml :

The web.xml file contains information about the web application, which is used by the Java web server / servlet container in order to properly deploy and execute the web application. For instance, the web.xml contains information about which servlets a web application should deploy, and what URL's they should be mapped to.

## classes Directory :

The classes directory contains all compiled Java classes that are part of your web application. The classes should be located in a directory structure matching their package structure.

## lib folder :

The lib directory contains all JAR files used by your web application. This directory most often contains any third party libraries that your application is using. You could, however, also put your own classes into a JAR file, and locate it here, rather than putting those classes in the classes directory.We will discuss more about this in next tutorial.

**PRACTICAL-3**

**Target : Implement Servlet to display a Message using GenericServlet Class**

Generic.java

import java.io.\*;

import jakarta.servlet.\*;

public class Generic extends GenericServlet {

public void service(ServletRequest req, ServletResponse res) throws

IOException, ServletException {

res.setContentType("text/HTML");

PrintWriter w = res.getWriter();

w.print(" <html> <head> Generic Servlet</head>");

w.print(" <body><h1> GENERIC SERVLET</h1>");

w.print(" <h2><a href ='index.html'>Go Back</a></h2></body></html>");

}

}

web.xml

<web-app>

<servlet>

<servlet-name>Generic</servlet-name>

<servlet-class>Generic</servlet-class>

</servlet>

<servlet-mapping>

<servlet-name>Generic</servlet-name>

<url-pattern>/Generic</url-pattern>

</servlet-mapping>

</web-app>

Index.html

<html>

<head>

<title> GenericServlet </title>

</head>

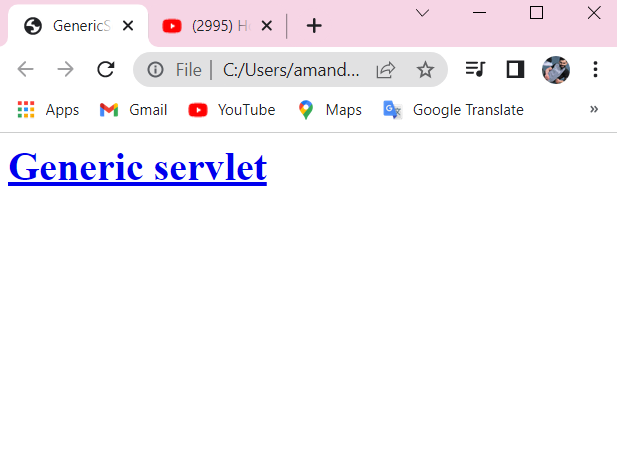
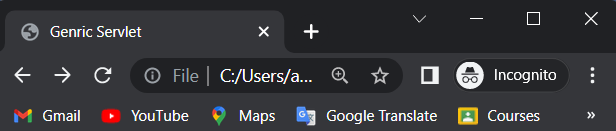
<body>

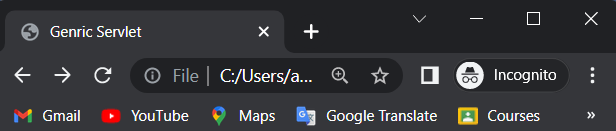
<p><h1> <a href ="Generic">Generic servlet</a></h1></p>

</body>

</html>

**-OUTPUT-**







**PRACTICAL-4**

Target : Implement Servlet to display a Message using Servlet Interface

Generic.java

import java.io.\*;

import jakarta.servlet.\*;

public class Interface implements Servlet {

ServletConfig sc = null;

public void init(ServletConfig sc) {

this.sc = sc;

System.out.println("Servlet initialized")

public void service(ServletRequest req, ServletResponse res) throws IOException, ServletException {

res.setContentType("text/HTML");

PrintWriter w = res.getWriter()

w.print(" <html> <head> Servlet Interface</head>");

w.print("<body><h1> SERVLET INTERFACE</h1>");

w.print(" <h2><a href ='index.html'>Go Back</a></h2></body></html> ");

}

public void destroy() {

System.out.println("Servlet destroys");

}

public ServletConfig getServletConfig() {

return sc;

}

public String getServletInfo() {

return "INFORMATION";

}

}

web.xml

<web-app>

<servlet>

<servlet-name>Interface</servlet-name>

<servlet-class>Interface</servlet-class>

</servlet>

<servlet-mapping>

<servlet-name>Interface</servlet-name>

<url-pattern>/Interface</url-pattern>

</servlet-mapping>

</web-app>

Index.html

<html>

<head>

<title> Servlet Interface </title>

</head>

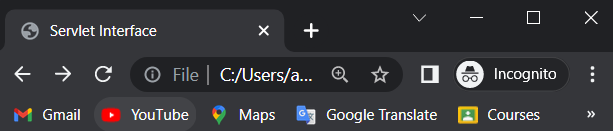
<body>

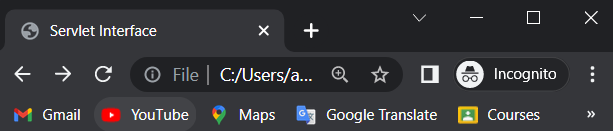
<p><h1> <a href ="Interface"> Servlet Interface </a></h1></p>

</body>

</html>

**-OUTPUT-**







**PRACTICAL-5**

**Target : Implement Servlet to display a Message using HttpServlet Class**

Generic.java

import java.io.\*;

import jakarta.servlet.\*;

import jakarta.servlet.http.\*;

public class Http extends HttpServlet {

public void doGet(HttpServletRequest req, HttpServletResponse res) throws IOException,

ServletException {

res.setContentType("text/HTML");

PrintWriter w = res.getWriter();

w.print(" <html> <head> Http Servlet</head>");

w.print(" <body ><h1> This is HTTP SERVLET</h1></font>");

w.print(" <h2><a href ='demo.html'>Go Back</a></h2></body></html> ");

}

}

web.xml

<web-app>

<servlet>

<servlet-name>Http</servlet-name>

<servlet-class>Http</servlet-class>

</servlet>

<servlet-mapping>

<servlet-name>Http</servlet-name>

<url-pattern>/Http</url-pattern>

</servlet-mapping>

</web-app>

Index.html

<html>

<head>

<title> Servlet Interface </title>

</head>

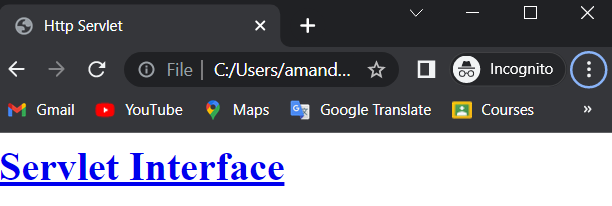
<body>

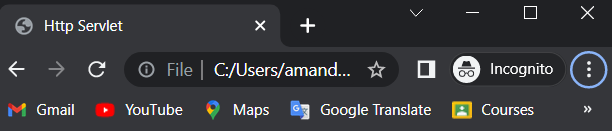
<p><h1> <a href ="Http"> Http Servlet</a></h1></p>

</body>

</html>

**-OUTPUT-**







**PRACTICAL-6**

**Target : Design an HTML Page using Lists, Headings, Image and Anchor Tag**

Index.html

<html>

<title>

HTML

</title>

<body>

<h1>This is heading 1</h1>

<h2>This is heading 2</h2>

<h3>This is heading 3</h3>

<h4>This is heading 4</h4>

<h5>This is heading 5</h5>

<h6>This is heading 6</h6>

<h1><b><u>lists</u></b></h1>

<ul>

<li>apple</li>

<li>mango</li>

<li>papaya</li>

</ul>

<ol type="a">

<li>apple</li>

<li>banana</li>

<li>watermelon</li>

</ol>

<center>

<img src ="youtube.png" ></image>

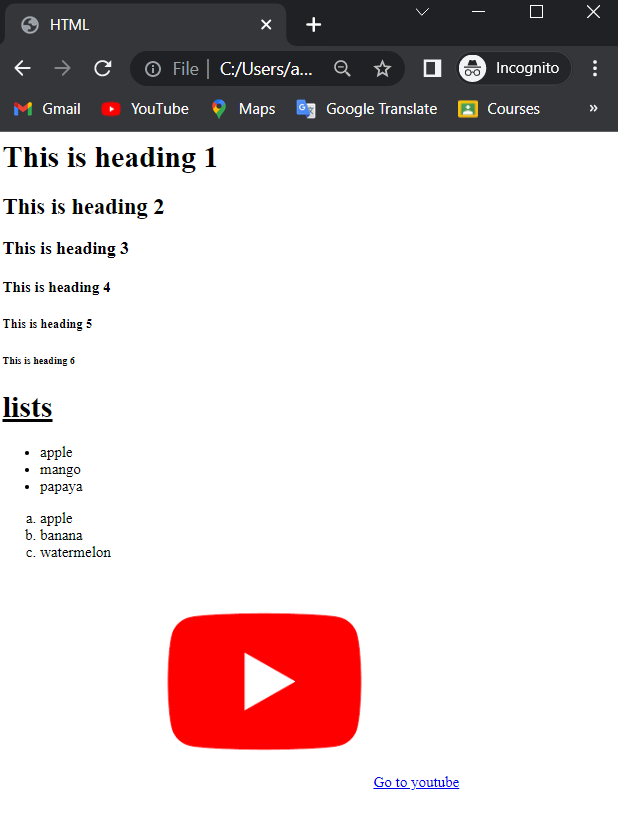
<a href="www.youtube.com">Go to youtube</a>

</center>

</body>

</html>

**-OUTPUT-**



**PRACTICAL-7**

**Target : Design an HTML Page using Tables**

Index.html

<html>

<head><tittle>Table by HTML</tittle><head>

<body><center>

<table border="2" bordercolor="black" bgcolor="RED">

<tr>

<th border="2" >Sno.</th>

<th border="2" >Name</th>

<th border="2" >Marks</th>

<th border="2" >Age</th>

<th border="2" >Branch</th></tr>

<tr>

<td border="2" align=center>1</td>

<td border="2" align=center>DAKSH</td>

<td border="2" align=center>100</td>

<td border="2" align=center>20</td>

<td border="2" >cse</td></tr>

<tr>

<td border="2" align=center>2</td>

<td border="2" align=center>DEEPAK</td>

<td border="2" align=center>100</td>

<td border="2" align=center>20</td>

<td border="2" >cse</td></tr>

<tr>

<td border="2" align=center>3</td>

<td border="2" align=center>AMAN</td>

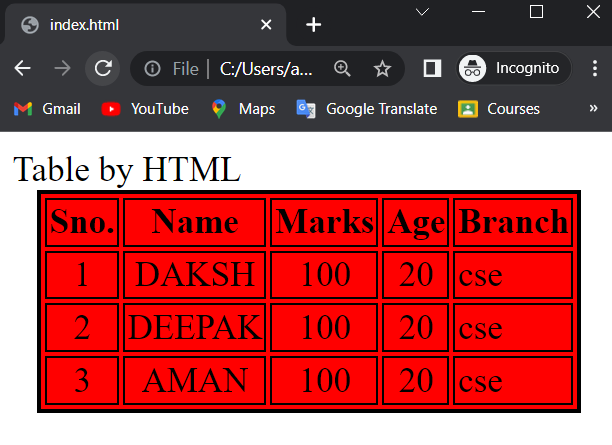
<td border="2" align=center>100</td>

<td border="2" align=center>20</td>

<td border="2" >cse</td></tr>

</tr></table></center</body></html>

**-OUTPUT-**

****

**PRACTICAL-8**

**Target : Design an HTML Form using different Form Elements**

Index.html

<html>

<head><title> Form </title></head>

<body>

<form name="Registeration" method="get">

<table align=center bgcolor="RED" border="2">

<tr>

<td>EnollmentNumber :</td>

<td><input type="text" name="EnollmentNumber" /></td>

</tr>

<tr>

<td>Name :</td>

<td><input type="name" name="Name" /></td>

</tr>

<tr>

<td>Branch :</td>

<td>

<select>

<option value="CSE">CSE </option>

<option value="CIVIL">CIVIL</option>

<option value="ME">ME</option>

<option value="EC">EC</option>

</select>

</td>

</tr>

<tr>

<td>Semester :</td>

<td>

<select>

<option value="I">I </option>

<option value="II">II</option>

<option value="III">III</option>

<option value="IV">IV</option>

</select>

</td>

</tr>

<tr>

<td>Gender</td>

<td>

<input type="radio" checked=true name="Gender" />

Male

<input type="radio" name="Gender" />

Female

</td>

</tr>

<tr>

<td>Hobbies :</td>

<td>

<input type="checkbox" name="Reading" value="Reading" />

Reading

<input type="checkbox" name="Swimming" value="Swimming" />

Swimming

<input type="checkbox" name="Music" value="Music" />

Music

</td>

</tr>

<tr>

<td>Password :</td>

<td><input type="password" name="Password" /></td>

</tr>

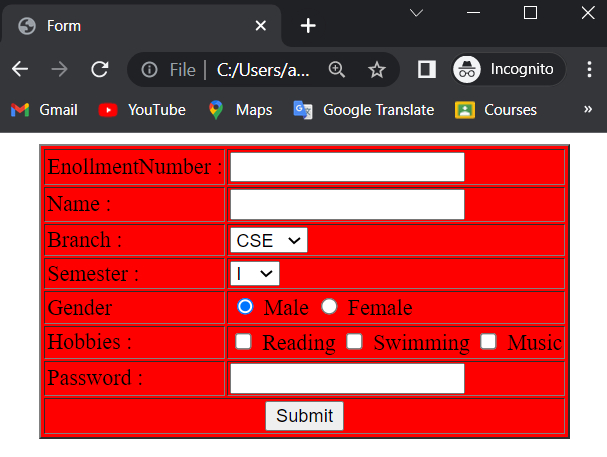
<tr>

<td colspan="2" align="center"><input type="submit" name="Submit" value="Submit"></td>

</tr>

</table></form></body></html>

**-OUTPUT-**



**PRACTICAL-9**

**Target : Implement Servlet to transfer Data from one Page (HTML / Servlet) to another (Servlet) using GET Method**

HTMLSERVLET.java

import java.io.\*;

import jakarta.servlet.\*;

import jakarta.servlet.http.\*;

public class HTMLSERVLET extends HttpServlet

{

public void doGet(HttpServletRequest req,HttpServletResponse res) throws

IOException,ServletException

{

res.setContentType("text/HTML");

PrintWriter w = res.getWriter();

String name = req.getParameter("fname");

int age = Integer.parseInt(req.getParameter("Age"));

w.print(" <html>");

w.print(" <body bgcolor ='#71COA7'>");

w.print(" <font color = '#A80000' text ='TIMES NEW ROMAN'>");

w.print(" <h1> You are "+name +"</h1>");

w.print(" <h1> Your age is "+age +"</h1></font>");

w.print("<h2><a href='doGet.html'>GoBack</a></h2></body></html> ");

}}

web.xml

<web-app>

<servlet>

<servlet-name>HTMLSERVLET</servlet-name>

<servlet-class>HTMLSERVLET</servlet-class>

</servlet>

<servlet-mapping>

<servlet-name>HTMLSERVLET</servlet-name>

<url-pattern>/HTMLSERVLET</url-pattern>

</servlet-mapping>

</web-app>

doGet.html

<html>

<head><title> deGet </title></head>

<body>

<form action ="Servlet4" name = "Registeration" method = "get">

<table align = center bgcolor = "#00FFFF" border ="2">

<tr>

<td>Name :</td>

<td><input type="text" name="fname"/></td></tr>

<tr>

<td>Age :</td>

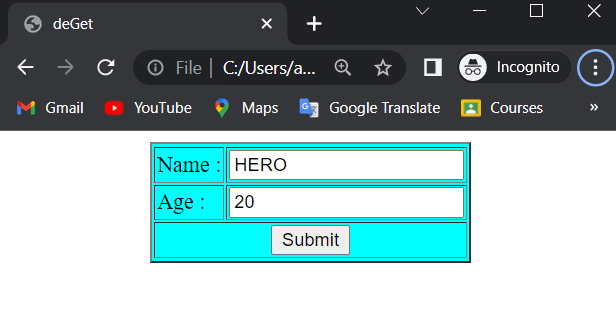
<td><input type="int" name="Age"/></td></tr>

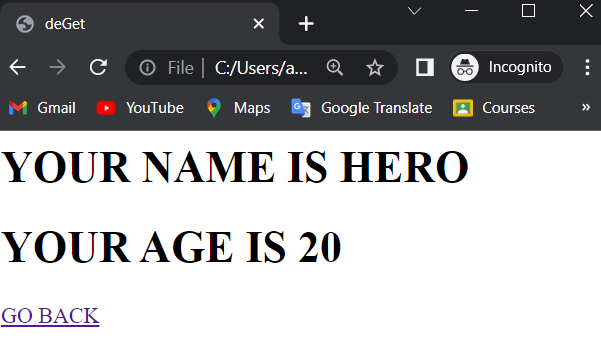
<tr>

<td colspan ="2" align ="center">

<input type="submit" name="Submit" value ="Submit"/></td></tr></table></form></body></html>

**-OUTPUT-**





**PRACTICAL-10**

**Target : Implement Servlet to transfer Data from one Page (HTML / Servlet) to another (Servlet) using POST Method**

HTMLSERVLET2.java

import java.io.\*;

import jakarta.servlet.\*;

import jakarta.servlet.http.\*;

public class HTMLSERVLET2 extends HttpServlet

{

public void doPost(HttpServletRequest req,HttpServletResponse res) throws

IOException,ServletException

{

res.setContentType("text/HTML");

PrintWriter w = res.getWriter();

String name = req.getParameter("fname");

int age = Integer.parseInt(req.getParameter("Age"));

w.print(" <html>");

w.print(" <body bgcolor ='#71COA7'>");

w.print(" <font color = '#A80000' text ='TIMES NEW ROMAN'>");

w.print(" <h1> You are "+name +"</h1>");

w.print(" <h1> Your age is "+age +"</h1></font>");

w.print("<h2><a href='doPost.html'>GoBack</a></h2></body></html> ");

}

}

web.xml

<web-app>

<servlet>

<servlet-name>HTMLSERVLET2</servlet-name>

<servlet-class>HTMLSERVLET2</servlet-class>

</servlet>

<servlet-mapping>

<servlet-name>HTMLSERVLET2</servlet-name>

<url-pattern>/HTMLSERVLET2</url-pattern>

</servlet-mapping>

</web-app>

doPost.html

<html>

<head>

<title>doPost</title></head>

<body>

<form action ="HTMLSERVLET2" name = "Registeration" method = "Post">

<table align = center bgcolor = "#00FFFF" border ="2">

<tr>

<td>Name :</td>

<td><input type="text" name="fname"/></td></tr>

<tr>

<td>Age :</td>

<td><input type="int" name="Age"/></td></tr>

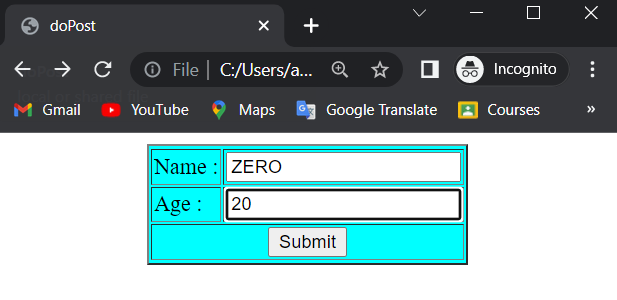
<tr>

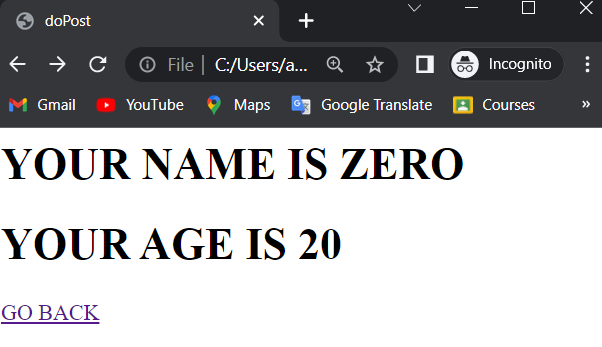
<td colspan ="2" align ="center">

<input type="submit" name="Submit" value ="Submit"/></td></tr>

</table></form></body></html>

**-OUTPUT-**





**PRACTICAL-11**

Target : Implement Servlet to compute Average of 3 Input Number

AVG.java

import java.io.\*;

import jakarta.servlet.\*;

import jakarta.servlet.http.\*;

public class AVG extends HttpServlet {

public void doPost(HttpServletRequest req, HttpServletResponse res) throws IOException, ServletException {

res.setContentType("text/HTML");

PrintWriter w = res.getWriter();

double n1 = Double.parseDouble(req.getParameter("number1"));

double n2 = Double.parseDouble(req.getParameter("number2"));

double n3 = Double.parseDouble(req.getParameter("number3"));

double sum = n1 + n2 + n3;

double average = sum / 3;

w.print(" <html>");

w.print(" <body bgcolor ='#71COA7'>");

w.print(" <font color = '#A80000' text ='TIMES NEW ROMAN'>");

w.print(" <h1> SUM : " + sum + "</h1>");

w.print(" <h1> AVERAGE: " + average + "</h1></font>");

w.print(" <h2><a href ='http://localhost:9999/myservlet'>Go Back</a></h2></body></html> ");

}

}

web.xml

<web-app>

<servlet>

<servlet-name>AVG</servlet-name>

<servlet-class>AVG</servlet-class>

</servlet>

<servlet-mapping>

<servlet-name>AVG</servlet-name>

<url-pattern>/AVG</url-pattern>

</servlet-mapping>

</web-app>

Average.html

<html>

<head>

<title>Average </title>

</head>

<body>

<form action ="Servlet5" name = "calculate" method = "post">

<table align = center bgcolor = "#00FFFF" border ="2">

<tr>

<td>Enter 1st number :</td>

<td><input type="double" name="number1"/></td>

</tr>

<tr>

<td>Enter 2nd number :</td>

<td><input type="double" name="number2"/></td>

</tr>

<tr>

<td>Enter 3rd number :</td>

<td><input type="double" name="number3"/></td>

</tr>

<tr>

<td colspan ="2" align ="center"><input type="submit" name="Calculate" value

="Calculate"/></td>

</tr>

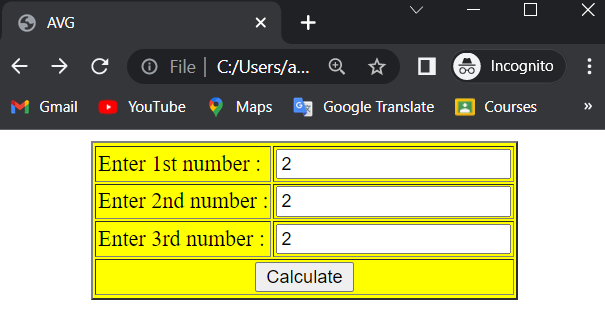
</table>

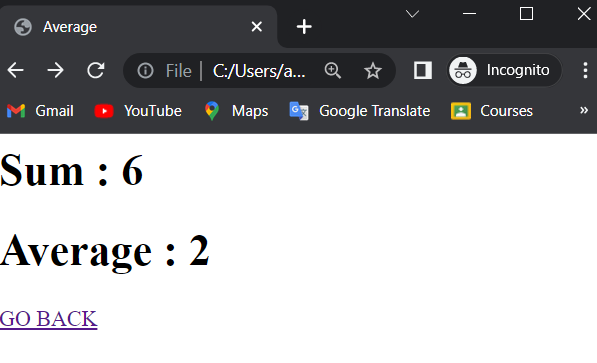
</form>

</body>

</html>

**-OUTPUT-**

****

****

**PRACTICAL-11**

**Target : Implement Servlet to display Table of an Input Number**

Table.java

import java.io.IOException;

import java.io.PrintWriter;

import jakarta.servlet.ServletException;

import jakarta.servlet.http.HttpServlet;

import jakarta.servlet.http.HttpServletRequest;

import jakarta.servlet.http.HttpServletResponse;

public class Table extends HttpServlet {

@Override

protected void doPost(HttpServletRequest request, HttpServletResponse response)

throws ServletException, IOException {

response.setContentType("text/html;charset=UTF-8");

try ( PrintWriter out = response.getWriter()) {

int number = Integer.parseInt(request.getParameter("number"));

out.print("<html> <title> Table of number </title><body> <div align ='center'>");

out.print("<table>");

for (int i = 1; i <= 10; i++) {

out.print("<tr><td>" + number + " \* " + i + " = " + i \* number + "</td></tr>");

}

out.print("</table>");

out.print("<a href ='print.html'>Go back </a>");

out.print("</div></body> </html>");

}

}

}

web.xml

<web-app>

<servlet>

<servlet-name>AVG</servlet-name>

<servlet-class>AVG</servlet-class>

</servlet>

<servlet-mapping>

<servlet-name>AVG</servlet-name>

<url-pattern>/AVG</url-pattern>

</servlet-mapping>

</web-app>

print.html

<html lang="en">

<head>

<meta charset="UTF-8">

<meta http-equiv="X-UA-Compatible" content="IE=edge">

<meta name="viewport" content="width=device-width, initial-scale=1.0">

<title>Table</title>

</head>

<body>

<div align="center">

<form action="intro" method="post">

<table>

<tr>

<td>

Enter the number :

</td>

<td><input type="text" name="number" />

</td>

</tr>

<tr>

<td><input type="submit" value="submit" />

</td>

<td><input type="reset" value="clear" />

</td>

</tr>

</table>

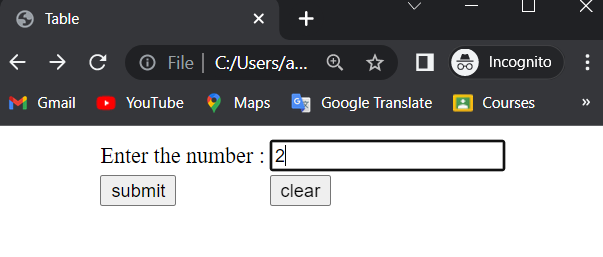
</form>

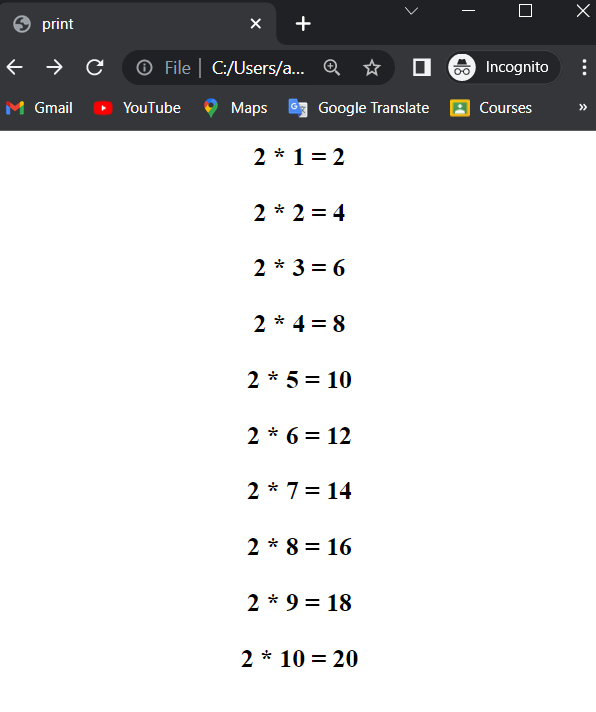
</div>

</body>

</html>

**-OUTPUT-**





Bottom of Form