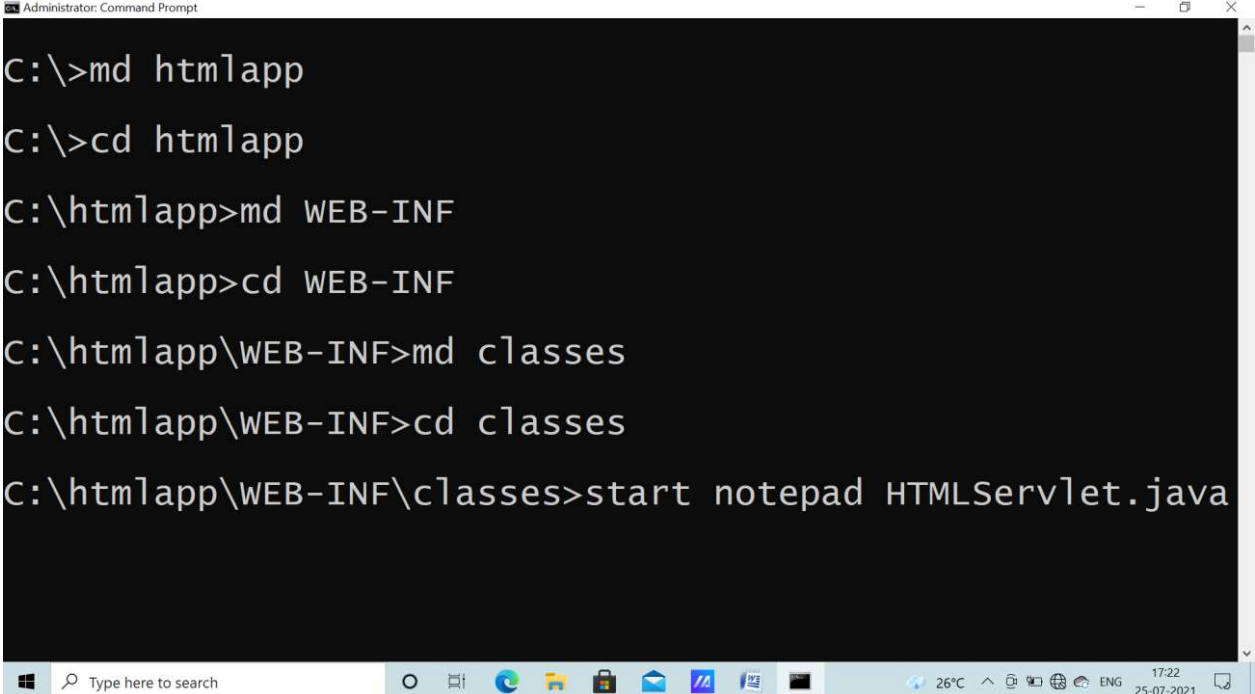


SERVLETS PART-2

Steps to develop web application:

- 1) Create a web application folder structure.
- 2) Create and compile web application source code(Servlet program).
- 3) Write deployment descriptor(web.xml).
- 4) Create a WAR(Web ARchive) file.
- 5) Start the server.
- 6) Deploy a war file on server.

1) Create a web application folder structure:



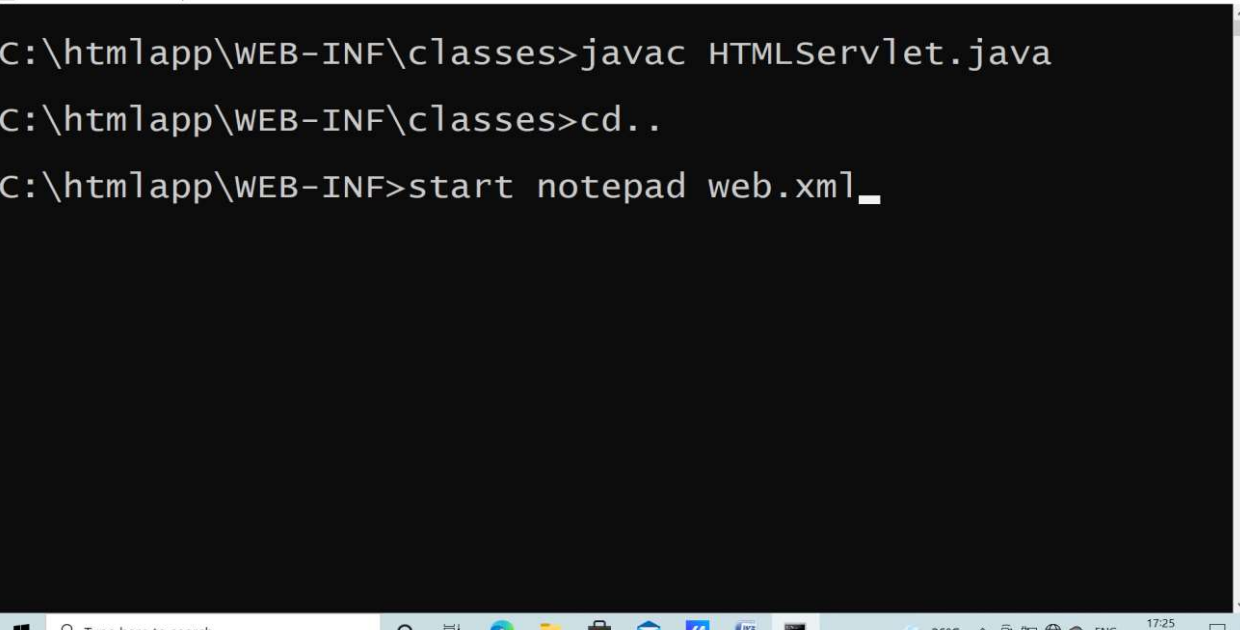
```
Administrator: Command Prompt
C:\>md htmlapp
C:\>cd htmlapp
C:\htmlapp>md WEB-INF
C:\htmlapp>cd WEB-INF
C:\htmlapp\WEB-INF>md classes
C:\htmlapp\WEB-INF>cd classes
C:\htmlapp\WEB-INF\classes>start notepad HTMLServlet.java
```

2) Create and compile web application source code(Servlet Program):

HTMLServlet.java:

```
import java.io.*;
import javax.servlet.*;
```

```
public class HTMLServlet extends GenericServlet
{
    public void service(ServletRequest req, ServletResponse res)
    {
        try{
            PrintWriter pw=res.getWriter();
            pw.println("<html><body bgcolor=yellow text=blue>");
            pw.println("<h1>Welcome.....</h1>");
            pw.println("</body></html>");
        }catch(Exception e)
        {
            System.err.println(e);
        }
    }
}
```



The screenshot shows a Windows Command Prompt window titled "Administrator: Command Prompt". The command history is as follows:

```
C:\htmlapp\WEB-INF\classes>javac HTMLServlet.java
C:\htmlapp\WEB-INF\classes>cd..
C:\htmlapp\WEB-INF>start notepad web.xml
```

The taskbar at the bottom shows the Windows search bar, task view button, and several application icons. The system tray on the right indicates a temperature of 26°C, network status, and the date and time: 17:25 on 25-07-2021.

3) Write deployment descriptor(web.xml):

web.xml:

```
<web-app>

<servlet>

<servlet-name>html</servlet-name>

<servlet-class>HTMLServlet</servlet-class>

</servlet>

<servlet-mapping>

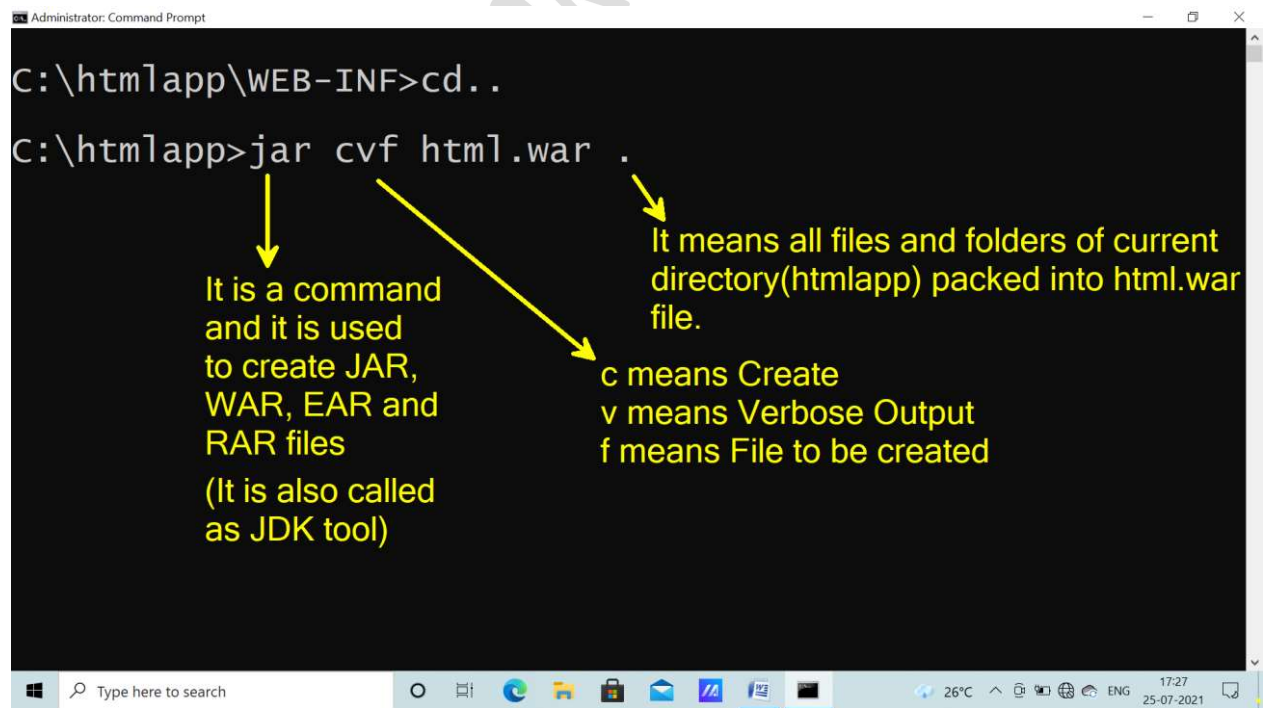
<servlet-name>html</servlet-name>

<url-pattern>/html</url-pattern>

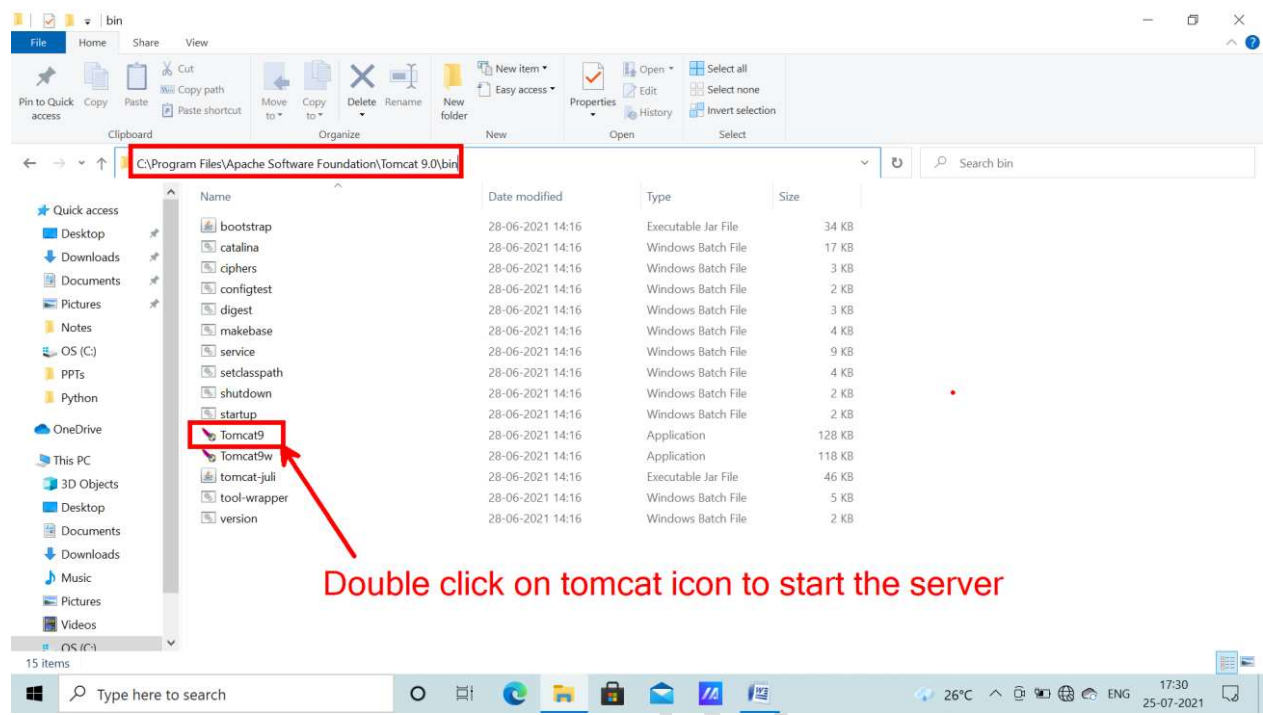
</servlet-mapping>

</web-app>
```

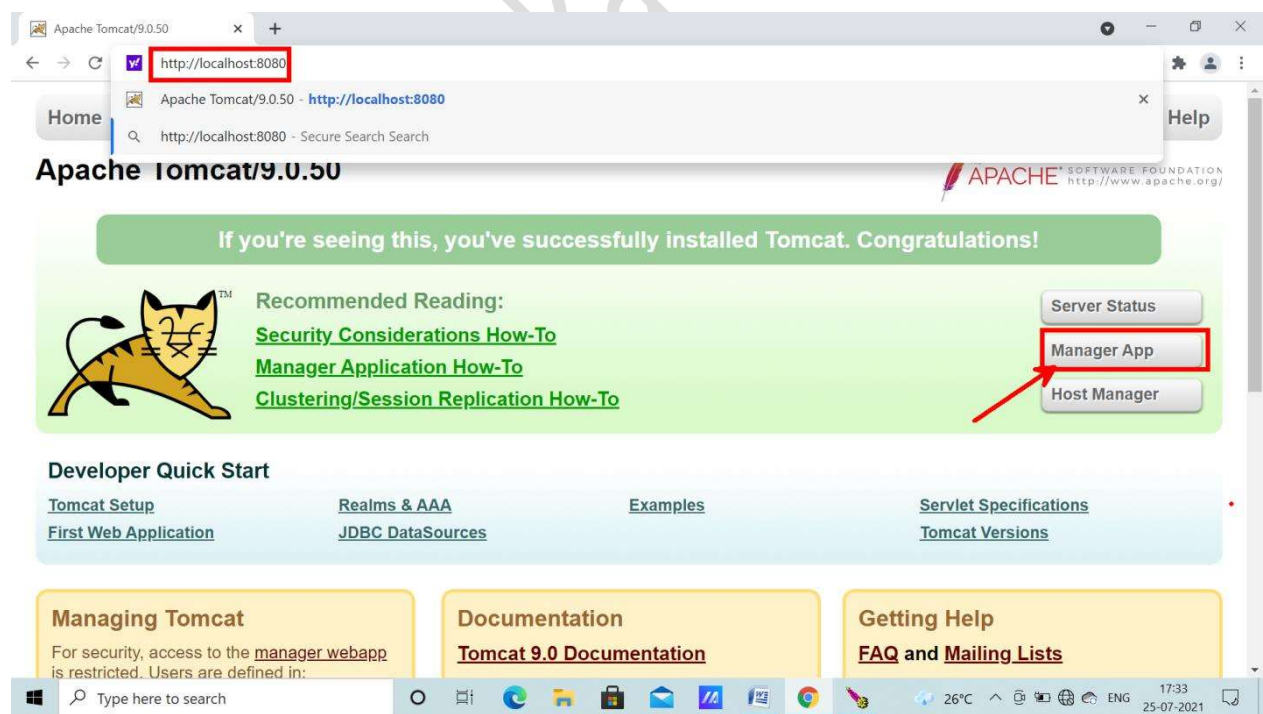
4) Create a WAR(Web ARchive) file:

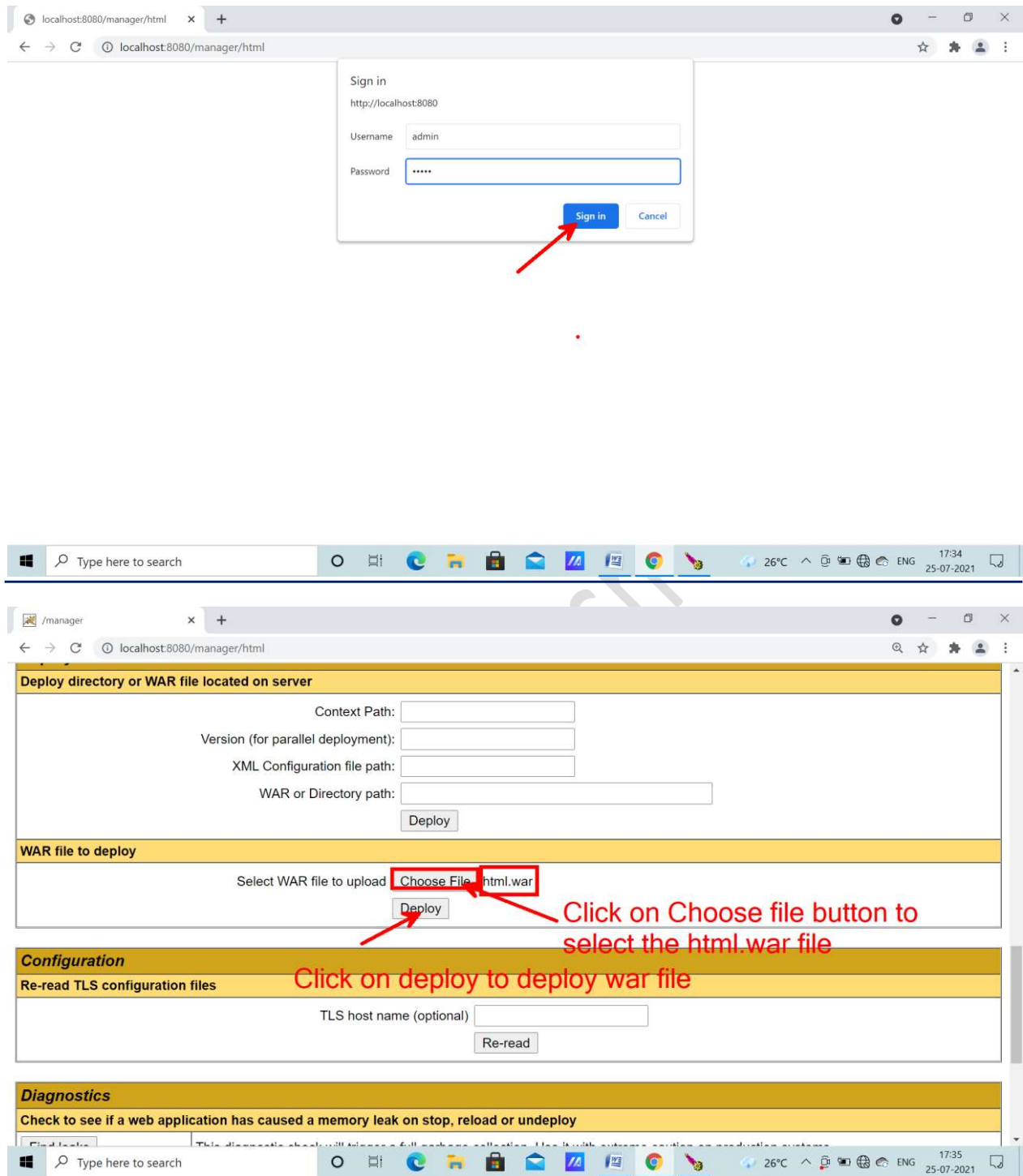


5) Start the server:



6) Deploy a war file on server:





TimeServlet.java:

```
import java.io.*;  
import java.util.*;
```

```
import javax.servlet.*;

public class TimeServlet extends GenericServlet
{
    public void service(ServletRequest req, ServletResponse res)
    {
        try{
            Date d=new Date();
            PrintWriter pw=res.getWriter();
            pw.println("<html><body bgcolor=red text=yellow><h1>");
            pw.println(d.getHours()+":"+d.getMinutes()+":"+d.getSeconds());
            pw.println("</h1></body></html>");
        }catch(Exception e)
        {
            System.err.println(e);
        }
    }
}
```

web.xml:

```
<web-app>
<servlet>
<servlet-name>time</servlet-name>
<servlet-class>TimeServlet</servlet-class>
</servlet>
<servlet-mapping>
```

```
<servlet-name>time</servlet-name>
```

```
<url-pattern>/time</url-pattern>
```

```
</servlet-mapping>
```

```
</web-app>
```

ImageServlet.java:

```
import java.io.*;
```

```
import javax.servlet.*;
```

```
public class ImageServlet extends GenericServlet
```

```
{
```

```
    public void service(ServletRequest req, ServletResponse res)
```

```
    {
```

```
        try{
```

```
            FileInputStream fis=new FileInputStream("C:/flowers.jpg"); //It opens  
a file for reading
```

```
            int n=fis.available(); //It returns file size
```

```
            byte[] b=new byte[n]; //It allocates the memory in a RAM according  
to file size
```

```
            fis.read(b); //It will read data from file.
```

```
            ServletOutputStream sos=res.getOutputStream(); //It creates a  
stream from server to browser
```

```
            sos.write(b); //It will send the image to browser
```

```
        }catch(Exception e)
```

```
        {
```

```
            System.err.println(e);
```

```
        }
```

```
}  
}
```

web.xml:

```
<web-app>  
<servlet>  
  <servlet-name>image</servlet-name>  
  <servlet-class>ImageServlet</servlet-class>  
</servlet>  
<servlet-mapping>  
  <servlet-name>image</servlet-name>  
  <url-pattern>/image</url-pattern>  
</servlet-mapping>  
</web-app>
```

TableServlet.java:

```
import java.io.*;  
import javax.servlet.*;  
public class TableServlet extends GenericServlet  
{  
    public void service(ServletRequest req, ServletResponse res)  
    {  
        try{  
            PrintWriter pw=res.getWriter();  
            pw.println("<html><body bgcolor=green text=yellow>");  
            pw.println("<table border=10>");  
        }  
    }  
}
```



```
        for(int i=1;i<=100;i++)
        {
            pw.println("<tr>");
            pw.println("<td> 5 </td>");
            pw.println("<td> x </td>");
            pw.println("<td> "+i+" </td>");
            pw.println("<td> = </td>");
            pw.println("<td> "+(5*i)+" </td>");
            pw.println("</tr>");
        }
        pw.println("</table></body></html>");
    }catch(Exception e)
    {
        System.err.println(e);
    }
}
```

web.xml:

```
<web-app>
<servlet>
<servlet-name>table</servlet-name>
<servlet-class>TableServlet</servlet-class>
</servlet>
<servlet-mapping>
```

```
<servlet-name>table</servlet-name>
```

```
<url-pattern>/table</url-pattern>
```

```
</servlet-mapping>
```

```
</web-app>
```

CounterServlet.java:

```
import java.io.*;
```

```
import javax.servlet.*;
```

```
public class CounterServlet extends GenericServlet
```

```
{
```

```
    int count=0;
```

```
    public void service(ServletRequest req, ServletResponse res)
```

```
    {
```

```
        try{
```

```
            PrintWriter pw=res.getWriter();
```

```
            pw.println("<html><body bgcolor=green text=yellow><h1>");
```

```
            pw.println("This page has been accessed "++count+" times");
```

```
            pw.println("</h1></body></html>");
```

```
        }catch(Exception e)
```

```
        {
```

```
            System.err.println(e);
```

```
        }
```

```
    }
```

```
}
```

web.xml:

```
<web-app>  
<servlet>  
<servlet-name>count</servlet-name>  
<servlet-class>CounterServlet</servlet-class>  
</servlet>  
<servlet-mapping>  
<servlet-name>count</servlet-name>  
<url-pattern>/count</url-pattern>  
</servlet-mapping>  
</web-app>
```

By

Mr. Venkatesh Mansani

Naresh i Technologies