Practical No. 18

Develop a simple servlet program which maintains a counter for the number of times it has been accessed since its loading; initialize the counter using deployment descriptor.

web.xml :

```
<?xml version="1.0" encoding="UTF-8" ?>
<web-app>
  <servlet>
    <servlet-name>Practical 18</servlet-name>
    <servlet-class>Practical 18</servlet-class>
    <init-param>
      <param-name>page_counter</param-name>
      <param-value>0</param-value>
    </init-param>
  </servlet>
  <servlet-mapping>
    <servlet-name>Practical 18</servlet-name>
    <url-pattern>/Practical18</url-pattern>
  </servlet-mapping>
</web-app>
```

Practical 18.java :

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

public class Practical18 extends HttpServlet {
    int i;
    public void init() throws ServletException {
        ServletConfig config = getServletConfig();
        String s = config.getInitParameter("page_
        counter");
        i = Integer.parseInt(s); }
```

```
protected void doGet(HttpServletRequest request,
HttpServletResponse response) throws ServletException,
IOException {
        response.setContentType("text/html");
        PrintWriter out.response.getWriter();
        it++;

        out.println("<HTML>");
        out.println("<BODY>");
        out.println("<H1>Page count is :"+it"</H1>");
        out.println("</BODY>");
        out.println("</HTML>");
    }

protected void doPost(HttpServletRequest request,
HttpServletResponse response) throw ServletException,
IOException {
        doGet(request,response);
    }
```

Conclusion /Observation:

To count the number of times pages is loaded here we haved used doGet() and doPost() method and we have initialize the counter in the web.xml file.

Practical No. 19

Write an HTML code to create login form having one submit button, two textboxes labeled as Login name and Password as respectively. Write a Servlet class named as ReadParameter to read these two parameters and display entered parameters values on the page using doGet() or doPost() method when user clicked on submit button.

Login.html :

```
<html>
  <body>
    <form action="ReadParameter" method="POST">
        Login Name : <input type="text" name="name" /><br/>
        Password : <input type="password" name="password" />
        </br>
        <input type="submit" value="Submit" />
    </form>
  </body>
</html>
```

ReadParameter.java :

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

public class ReadParameter extends HttpServlet {
      protected void doGet(HttpServletRequest request,
HttpServletResponse response) throws ServletException,
IOException {
          doPost(request,response); }

      protected void doPost(HttpServletRequest request,
HttpServletResponse response) throw ServletException,
IOException{
          response.setContentType("text/html");
```

```
PrintWriter out = response.getWriter();
String n = request.getParameter("name");
String p = request.getParameter("password");

out.println("HTML");
out.println("BODY");
out.println("Welcome " + n + "<br>");
out.print("Your password is " + p);
out.println("</BODY>");
out.println("</HTML>");
    }
}
```

web.xml :

```
<?xml version="1.0" encoding="UTF-8"?>
<web-app>
    <servlet>
        <servlet-name>S3</servlet-name>
        <servlet-class>ReadParameter</servlet-class>
    </servlet>
    <servlet-mapping>
        <servlet-name>S3</servlet-name>
        <url-pattern>/ReadParameter</url-pattern>
    </servlet-mapping>
</web-app>
```

Conclusion /observation :

Here Login.html contains form, label and textfield. By clicking on submit button it is redirecting to the ReadParameter.java page which displays the results.