

Dt : 9/6/2022

Note:

=>we use the following tags part of web.xml to execute filter program:

<web-app>

.

<filter>

<filter-name> name </filter-name>

<filter-class> class </filter-class>

<init-param>

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

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

</init-param>

</filter>

<filter-mapping>

<filter-name>name</filter-name>

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

</filter-mapping>

</web-app>

=====

Ex_Application-1:

login.html

<!DOCTYPE html>

<html>

```

<head>
<meta charset="ISO-8859-1">
<title>Insert title here</title>
</head>
<body>
  <form action="log" method="post">
    UserName:<input type="text" name="uname"><br>
    PassWord:<input type="password" name="pword"><br>
    <input type="submit" value="Login">
    <a href="register.html">NewUser?</a>
  </form>
</body>
</html>

```

DBConnection.java

```

package test;
import java.sql.*;
public class DBConnection {
    private static Connection con=null;//reference variable
    private DBConnection() {}
    static
    {
        try {
            Class.forName("oracle.jdbc.driver.OracleDriver");
            con = DriverManager.getConnection
("jdbc:oracle:thin:@localhost:1521:xe","system","manager");
        } catch (Exception e) {e.printStackTrace();}
    }
    public static Connection getCon()
    {
        return con;
    }
}

```

LoginDAO.java

```

package test;

import java.sql.*;

import javax.servlet.*;

public class LoginDAO {

```

```

public String fName=null;

public String login(ServletRequest req) {

    try {

        Connection con = DBConnection.getCon();

        PreparedStatement ps = con.prepareStatement
            ("select * from UserReg45 where uname=? and pword=?");

        ps.setString(1,req.getParameter("uname"));

        ps.setString(2,req.getParameter("pword"));

        ResultSet rs = ps.executeQuery();

        if(rs.next()) {

            fName = rs.getString(3);
        }

    }catch(Exception e) {e.printStackTrace();}

    return fName;

}
}

```

LoginFilter.java

```

package test;

import java.io.*;

import javax.servlet.*;

import javax.servlet.annotation.*;

@WebFilter("/log")

public class LoginFilter implements Filter{

    public void doFilter(ServletRequest req,ServletResponse res,

        FilterChain chain)throws ServletException,IOException{

```

```

        PrintWriter pw = res.getWriter();

        res.setContentType("text/html");

        String fName = new LoginDAO().login(req);

        if(fName==null) {

            pw.println("Invalid Login process...<br>");

            RequestDispatcher rd =

                req.getRequestDispatcher("login.html");

            rd.include(req, res);

        }else {

            req.setAttribute("fname", fName);

            chain.doFilter(req, res); //linking Servlet_program

        }

    }

}

```

WelcomeServlet.java

```

package test;

import java.io.*;

import javax.servlet.*;

import javax.servlet.http.*;

import javax.servlet.annotation.*;

@SuppressWarnings("serial")

@WebServlet("/log")

public class WelcomeServlet extends HttpServlet{

    protected void doPost(HttpServletRequest req,HttpServletResponse res)

```

```

throws ServletException,IOException{

    PrintWriter pw = res.getWriter();

    res.setContentType("text/html");

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

    pw.println("Login Successfull...<br>");

    pw.println("Welcome User "+name);

}

}

```

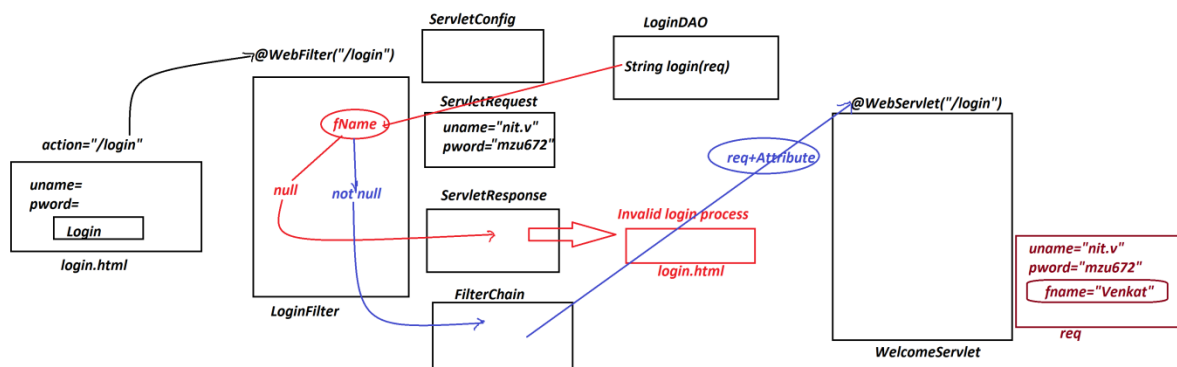
web.xml

```

<?xml version="1.0" encoding="UTF-8"?>
<web-app>
    <welcome-file-list>
        <welcome-file>login.html</welcome-file>
    </welcome-file-list>
</web-app>

```

Diagram:



Dt : 11/6/2022

Filter_program-2:

Application to demonstrate FilterConfig.

input.html

```
<!DOCTYPE html>
<html>
<head>
<meta charset="ISO-8859-1">
<title>Insert title here</title>
</head>
<body>
  <form action="dis" method="post">
    UserName:<input type="text" name="uname"><br>
    <input type="submit" value="Display">
  </form>
</body>
</html>
```

web.xml

```
<?xml version="1.0" encoding="UTF-8"?>
<web-app>
  <filter>
    <filter-name>DisplayFilter</filter-name>
    <filter-class>test.DisplayFilter</filter-class>
    <init-param>
      <param-name>a</param-name>
      <param-value>3000</param-value>
    </init-param>
  </filter>
  <filter-mapping>
    <filter-name>DisplayFilter</filter-name>
    <url-pattern>/dis</url-pattern>
  </filter-mapping>
  <welcome-file-list>
    <welcome-file>input.html</welcome-file>
  </welcome-file-list>
</web-app>
```

DisplayFilter.java

```
package test;
```

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

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

```

public class DisplayFilter implements Filter{

    public FilterConfig fc=null;//Instance Variable

    public void init(FilterConfig fc)throws ServletException{

        this.fc=fc;

    }

    public void doFilter(ServletRequest req,ServletResponse res,

        FilterChain chain)throws ServletException,IOException{

        PrintWriter pw = res.getWriter();

        res.setContentType("text/html");

        pw.println("UserName:"+req.getParameter("uname"));

        pw.println("<br>====FilterConfig====");

        pw.println("<br>FilterName:"+fc.getFilterName());

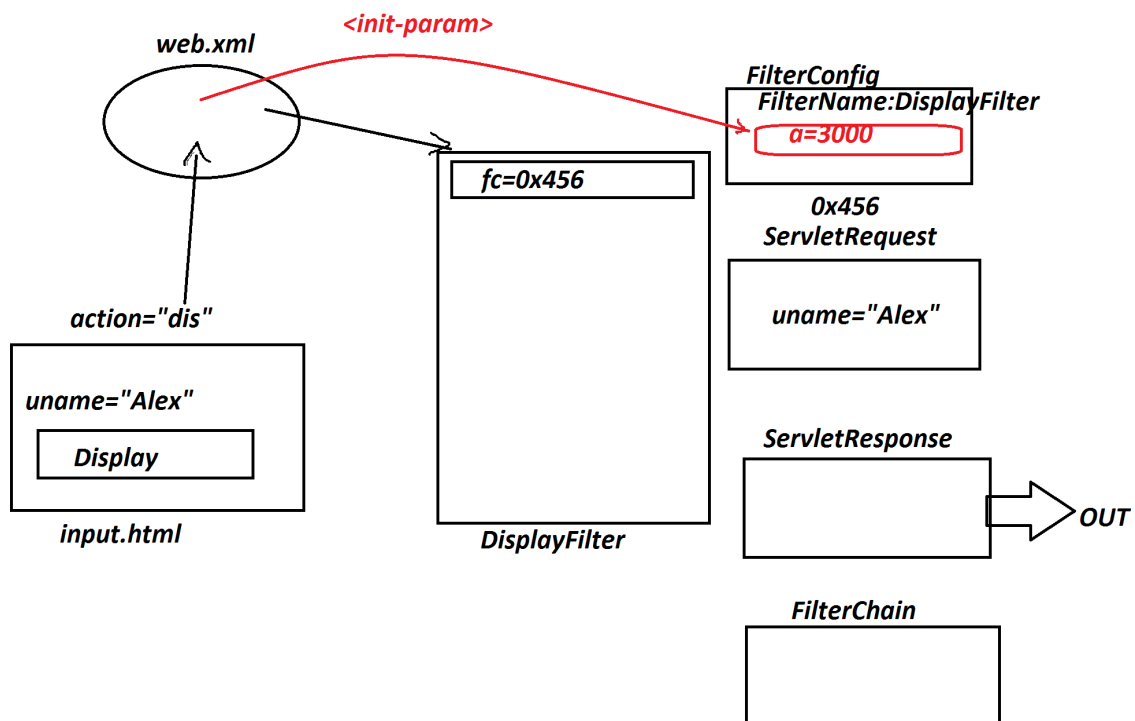
        pw.println("<br>FilterValue:"+fc.getInitParameter("a"));

    }

}

```

Diagram:



faq:

wt is the diff b/w

(i)ServletConfig

(ii)FilterConfig

(i)ServletConfig:

=>ServletConfig is instantiated automatically when the Servlet program is

loaded for execution.

=>ServletConfig object will hold Servlet name.

(ii)FilterConfig:

*=>FilterConfig is instantiated automatically when the Filter program loaded
for execution.*

=>FilterConfig object will hold Filter name

=====

Note:

=>In realtime filters are used in Security and protection layer.

=====

Venkatesh Maipathii