# 1、登录拦截器需求

Filter的定义是在请求一个资源或者从一个资源返回信息的时候执行过滤操作的插件。Servlet API中提供了一个Filter接口，开发web应用时，如果编写的Java类实现了这个接口，则把这个java类称之为过滤器Filter。通过Filter技术，开发人员可以实现用户在访问某个目标资源之前，对访问的请求和响应进行拦截,一般常用于实现URL级别的权限访问控制、过滤敏感词汇、压缩响应信息等一些高级功能。

# 2、处理步骤

## 2.1 编写登录页面

```
<%@ page contentType="text/html;charset=UTF-8" language="java" %>
<%@ taglib prefix="c" uri="http://java.sun.com/jstl/core" %>
<%
String path=request.getContextPath();
String
basePath=request.getScheme()+"://"+request.getServerName()+":"+request.getServer
Port()+path+"/";
%>
<html  class="x-admin-sm">
<head>
    <meta charset="UTF-8">
    <title>管理-X-admin2.1</title>
    <meta name="renderer" content="webkit|ie-comp|ie-stand">
    <meta http-equiv="X-UA-Compatible" content="IE=edge,chrome=1">
    <meta http-equiv="Cache-Control" content="no-siteapp" />
    <link rel="stylesheet" href="<%=basePath%>admin/css/font.css">
    <link rel="stylesheet" href="<%=basePath%>admin/css/xadmin.css">
    <script type="text/javascript"
src="https://cdn.bootcss.com/jquery/3.2.1/jquery.min.js"></script>
    <script src="<%=basePath%>admin/lib/layui/layui.js" charset="utf-8">
</script>
    <script type="text/javascript" src="<%=basePath%>admin/js/xadmin.js">
</script>
    <script type="text/javascript" src="<%=basePath%>admin/js/cookie.js">
</script>

</head>
<body class="login-bg">

    <div class="login layui-anim layui-anim-up">
        <div class="message">系统登录</div>
        <div id="darkbannerwrap"></div>

        <form method="post" class="layui-form" >
            <input name="username" placeholder="用户名"  type="text"
                   lay-verify="required" class="layui-input" >
            <hr class="hr15">
            <input name="password" lay-verify="required" placeholder="密码"
                   type="password" class="layui-input">
            <hr class="hr15">
            <input value="登录" lay-submit lay-filter="login" style="width:100%;"
type="submit">
            <hr class="hr20" >
        </form>
    </div>

    <script>
        $(function  () {
            layui.use('form', function(){
              var form = layui.form;

              //监听提交
              form.on('submit(login)', function(data){
                  //登录成功跳转主页
                  var data=data.field;
                  debugger
                  $.ajax({
                      url:"<%=basePath%>loginIn",
                      type:"post",
                      data:data,
                      success:function (result) {
                          debugger
                          var res=JSON.parse(result);
                          if(res.code==0){
                              layer.alert("登录成功", {icon: 6},function () {
                                  location.href='Index.jsp'
                              });
                          }else{
                              layer.msg("登录失败，重新登录...")
                          }
                      }
                  });
                  return false;
                // // alert(888)
                // layer.msg(JSON.stringify(data.field),function(){
                //     location.href='index.jsp'
                // });
                // return false;
              });
            });
        })


    </script>


    <!-- 底部结束 -->
</body>
</html>
```

## 2.2 后端登录接口

```
package com.wineshop.controller;

import com.alibaba.fastjson.JSON;
import com.wineshop.util.R;

import javax.servlet.ServletException;
import javax.servlet.annotation.WebServlet;
import javax.servlet.http.HttpServlet;
import javax.servlet.http.HttpServletRequest;
import javax.servlet.http.HttpServletResponse;
import javax.servlet.http.HttpSession;
import java.io.IOException;

@WebServlet( "/loginIn")
public class LoginServlet extends HttpServlet {
    @Override
    protected void service(HttpServletRequest req, HttpServletResponse resp)
throws ServletException, IOException {
        HttpSession session=req.getSession();
        //接收用户名 密码
        String name=req.getParameter("username");
        String pwd=req.getParameter("password");
        //根据用户名密码查询用户对象
        if(name.equals("kappy")){
            session.setAttribute("user","kappy");
            String jsons = JSON.toJSONString(R.ok());
            resp.getWriter().print(jsons);
        }else{
            resp.getWriter().print(JSON.toJSONString(R.fail("登录失败")));
        }



    }
}
```

## 2.3 登录拦截器编写

```
package com.wineshop.filter;

import javax.servlet.*;
import javax.servlet.annotation.WebFilter;
import javax.servlet.http.HttpServletRequest;
import javax.servlet.http.HttpServletResponse;
import javax.servlet.http.HttpSession;
import java.io.IOException;

//@WebFilter(filterName = "LoginFilter")
public class LoginFilter implements Filter {
    public void destroy() {
    }

    public void doFilter(ServletRequest request, ServletResponse response,
FilterChain chain) throws ServletException, IOException {

        HttpServletRequest req = (HttpServletRequest)request;
        HttpServletResponse res = (HttpServletResponse)response;
        HttpSession session = req.getSession();
        String  name = (String) session.getAttribute("user");
        String spath = req.getServletPath();
        System.out.println("request.getContextPath()"+req.getContextPath());
        //不需要过滤的url
        String[] urls = {"/loginIn","/json",".js",".css",".ico",".jpg",".png"};
        boolean flag = true;
        for (String str : urls) {
            if (spath.indexOf(str) != -1) {
                flag =false;
                break;
            }
        }
        if (flag) {
            if (name != null) {
                chain.doFilter(request, response);
            }else {
                res.sendRedirect(req.getContextPath()+"/admin/login.jsp");
            }
        }else{
            chain.doFilter(request, response);
        }
  }

    public void init(FilterConfig config) throws ServletException {

    }

}
```

如果还有其他的类型不需要过滤，直接放行后缀对应内容即可

### 2.4 拦截器配置

```
   <filter>
        <filter-name>loginFilter</filter-name>
        <filter-class>com.wineshop.filter.LoginFilter</filter-class>
    </filter>
    <filter-mapping>
        <filter-name>loginFilter</filter-name>
        <url-pattern>/admin/*</url-pattern>
    </filter-mapping>
```

## 3.内容完善，主页没登陆跳转

```
<%
   //如果没有登录跳转到登陆页面
   HttpSession session1= request.getSession();
   String name= (String) session1.getAttribute("user");
   if(name==null){
       response.sendRedirect("login.jsp");
   }
%>
```