MVC Architecture in JSP

**MVC** stands for Model View and Controller. It is a **design pattern** that separates the business logic, presentation logic and data.

**Controller** acts as an interface between View and Model. Controller intercepts all the incoming requests.

**Model** represents the state of the application i.e. data. It can also have business logic.

**View** represents the presentaion i.e. UI(User Interface).

951.2ow to save the iPhone 14 Plus

**Advantage of MVC Architecture**

1. Navigation Control is centralized
2. Easy to maintain the large application



**MVC Example in JSP**

In this example, we are using servlet as a controller, jsp as a view component, Java Bean class as a model.

In this example, we have created 5 pages:

* **index.jsp** a page that gets input from the user.
* **ControllerServlet.java** a servlet that acts as a controller.
* **login-success.jsp** and **login-error.jsp** files acts as view components.
* **web.xml** file for mapping the servlet.

*File: index.jsp*

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

Name:<input type="text" name="name"><br>

Password:<input type="password" name="password"><br>

<input type="submit" value="login">

</form>

*File: ControllerServlet*

**package** pkg;

**import** java.io.IOException;

**import** java.io.PrintWriter;

**import** javax.servlet.RequestDispatcher;

**import** javax.servlet.ServletException;

**import** javax.servlet.http.HttpServlet;

**import** javax.servlet.http.HttpServletRequest;

**import** javax.servlet.http.HttpServletResponse;

**public** **class** ControllerServlet **extends** HttpServlet {

**protected** **void** doPost(HttpServletRequest request, HttpServletResponse response)

**throws** ServletException, IOException {

        response.setContentType("text/html");

        PrintWriter out=response.getWriter();

        String name=request.getParameter("name");

        String password=request.getParameter("password");

        LoginBean bean=**new** LoginBean();

        bean.setName(name);

        bean.setPassword(password);

        request.setAttribute("bean",bean);

**boolean** status=bean.validate();

**if**(status){

            RequestDispatcher rd=request.getRequestDispatcher("login-success.jsp");

            rd.forward(request, response);

        }

**else**{

            RequestDispatcher rd=request.getRequestDispatcher("login-error.jsp");

            rd.forward(request, response);

        }

    }

    @Override

**protected** **void** doGet(HttpServletRequest req, HttpServletResponse resp)

**throws** ServletException, IOException {

        doPost(req, resp);

    }

}

*File: LoginBean.java*

**package** pkg;

**public** **class** LoginBean {

**private** String name,password;

**public** String getName() {

**return** name;

}

**public** **void** setName(String name) {

**this**.name = name;

}

**public** String getPassword() {

**return** password;

}

**public** **void** setPassword(String password) {

**this**.password = password;

}

**public** **boolean** validate(){

**if**(password.equals("admin")){

**return** **true**;

    }

**else**{

**return** **false**;

    }

}

}

*File: login-success.jsp*

<%@page **import**="pkg.LoginBean"%>

  <p>You are successfully logged in!</p>

<%

LoginBean bean=(LoginBean)request.getAttribute("bean");

out.print("Welcome, "+bean.getName());

%>

*File: login-error.jsp*

<p>Sorry! username or password error</p>

<%@ include file="index.jsp" %>

*File: web.xml*

**<?xml** version="1.0" encoding="UTF-8"**?>**

**<web-app** xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"

xmlns="http://java.sun.com/xml/ns/javaee" xmlns:web="http://java.sun.com/xml/ns/javaee/web-app\_2\_5.xsd"

xsi:schemaLocation="http://java.sun.com/xml/ns/javaee http://java.sun.com/xml/ns/javaee/web-app\_3\_0.xsd"

id="WebApp\_ID" version="3.0"**>**

**<servlet>**

**<servlet-name>**s1**</servlet-name>**

**<servlet-class>**pkg.ControllerServlet**</servlet-class>**

**</servlet>**

**<servlet-mapping>**

**<servlet-name>**s1**</servlet-name>**

**<url-pattern>**/ControllerServlet**</url-pattern>**

**</servlet-mapping>**

**</web-app>**