

Name: **Piyusha Rajendra Supe**

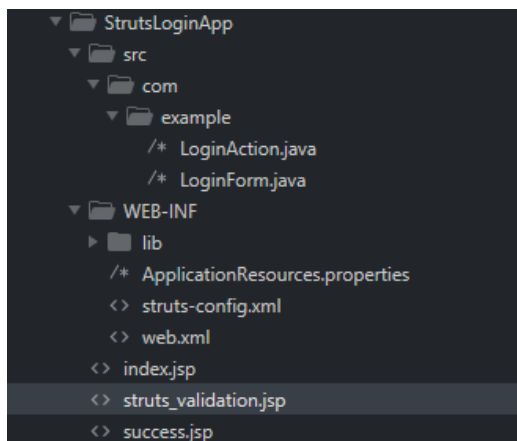
Enrolment Number: **23CO315**

Web Technology Practical 8

Title: Design a login page with entries for name, mobile number email id and login button. Use struts and perform following validations

- a. Validation for correct names
- b. Validation for mobile numbers
- c. Validation for email id
- d. Validation if no entered any value
- e. Re-display for wrongly entered values with message
- f. Congratulations and welcome page upon successful entries

The folder structure is as follows:



The code for each file is as follows:

LoginAction.java

```
package com.example;

import javax.servlet.http.HttpServletRequest;
import javax.servlet.http.HttpServletResponse;
import org.apache.struts.action.*;

public class LoginAction extends Action {

    public ActionForward execute(ActionMapping mapping, ActionForm form,
                                HttpServletRequest request, HttpServletResponse response) {

        LoginForm loginForm = (LoginForm) form;
        request.setAttribute("name", loginForm.getName());
```

```

    request.setAttribute("email", loginForm.getEmail());
    request.setAttribute("mobile", loginForm.getMobile());
    return mapping.findForward("success");
}
}

```

LoginForm.java

```

package com.example;

import org.apache.struts.action.ActionForm;
import org.apache.struts.action.ActionErrors;
import org.apache.struts.action.ActionMessage;
import org.apache.struts.action.ActionMapping;
import javax.servlet.http.HttpServletRequest;

public class LoginForm extends ActionForm {
    private String name;
    private String mobile;
    private String email;

    public String getName() { return name; }
    public void setName(String name) { this.name = name; }
    public String getMobile() { return mobile; }
    public void setMobile(String mobile) { this.mobile = mobile; }
    public String getEmail() { return email; }
    public void setEmail(String email) { this.email = email; }

    @Override
    public ActionErrors validate(ActionMapping mapping, HttpServletRequest request) {
        ActionErrors errors = new ActionErrors();
        if (name == null || name.trim().isEmpty()) {
            errors.add("name", new ActionMessage("error.name.required"));
        } else if (!name.matches("[A-Za-z ]+")) {
            errors.add("name", new ActionMessage("error.name.invalid"));
        }
        if (mobile == null || mobile.trim().isEmpty()) {
            errors.add("mobile", new ActionMessage("error.mobile.required"));
        } else if (!mobile.matches("\\d{10}")) {
            errors.add("mobile", new ActionMessage("error.mobile.invalid"));
        }
    }
}

```

```

    }
    if (email == null || email.trim().isEmpty()) {
        errors.add("email", new ActionMessage("error.email.required"));
    } else if (!email.matches("^\\S+@\\S+\\.\\S+$")) {
        errors.add("email", new ActionMessage("error.email.invalid"));
    }
    return errors;
}
}

```

ApplicationResources.properties

error.name.required = Name is required.

error.name.invalid = Name must contain only letters and spaces.

error.mobile.required = Mobile number is required.

error.mobile.invalid = Mobile number must be 10 digits.

error.email.required = Email is required.

error.email.invalid = Invalid email format.

Struts-config.xml

```

<struts-config>
    <form-beans>
        <form-bean name="loginForm" type="com.example.LoginForm" />
    </form-beans>
    <action-mappings>
        <action path="/login" type="com.example.LoginAction" name="loginForm" scope="request"
            validate="true" input="/index.jsp">
            <forward name="success" path="/success.jsp"/>
        </action>
    </action-mappings>
</struts-config>

```

Web.xml

```

<web-app>
    <welcome-file-list>
        <welcome-file>index.jsp</welcome-file>
    </welcome-file-list>

```

```

<display-name>StrutsLoginApp</display-name>

<servlet>

  <servlet-name>action</servlet-name>

  <servlet-class>org.apache.struts.action.ActionServlet</servlet-class>

  <init-param>

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

    <param-value>/WEB-INF/struts-config.xml</param-value>

  </init-param>

  <load-on-startup>1</load-on-startup>

</servlet>

<servlet-mapping>

  <servlet-name>action</servlet-name>

  <url-pattern>*.do</url-pattern>

</servlet-mapping>

<taglib>

  <taglib-uri>http://struts.apache.org/tags-html</taglib-uri>

  <taglib-location>/WEB-INF/struts-html.tld</taglib-location>

</taglib>

</web-app>

```

Success.jsp

```

<% @ page contentType="text/html; charset=UTF-8" %>

<html>

<head><title>Success</title></head>

<body>

<h2>Welcome!</h2>

<p>Congratulations <b><%= request.getAttribute("name") %></b></p>

<p>Email: <%= request.getAttribute("email") %></p>

<p>Mobile: <%= request.getAttribute("mobile") %></p>

</body>

</html>

```

Struts_validation.jsp

```

<% @ page import="java.util.regex.*" %>

```

```

<% @ page contentType="text/html; charset=UTF-8" language="java" %>
<%
    String name = request.getParameter("name");
    String mobile = request.getParameter("mobile");
    String email = request.getParameter("email");

    boolean isSubmitted = request.getParameter("submit") != null;
    boolean isValid = true;
    String nameError = "", mobileError = "", emailError = "";

    if (isSubmitted) {
        if (name == null || name.trim().equals("")) {
            nameError = "Name is required.";
            isValid = false;
        } else if (!name.matches("^[A-Za-z\\s]+$")) {
            nameError = "Name must contain only letters.";
            isValid = false;
        }
    }

    if (mobile == null || mobile.trim().equals("")) {
        mobileError = "Mobile number is required.";
        isValid = false;
    } else if (!mobile.matches("[0-9]{10}$")) {
        mobileError = "Mobile number must be 10 digits.";
        isValid = false;
    }

    if (email == null || email.trim().equals("")) {
        emailError = "Email is required.";
        isValid = false;
    } else {
        Pattern emailPattern = Pattern.compile("^[A-Za-z0-9+_.-]+@[A-Za-z0-9.-]+$");

```

```
Matcher matcher = emailPattern.matcher(email);
if (!matcher.matches()) {
    emailError = "Invalid email format.";
    isValid = false;
}
}
}
%>
<html>
<head>
<title>Login Page with JSP Validation</title>
<style>
body {
    font-family: Arial, sans-serif;
    background: #e8f0fe;
    display: flex;
    justify-content: center;
    align-items: center;
    height: 100vh;
}
.form-container {
    background: white;
    padding: 25px;
    border-radius: 12px;
    box-shadow: 0 0 15px rgba(0,0,0,0.1);
    width: 350px;
}
h2 {
    text-align: center;
    margin-bottom: 20px;
    color: #3366cc;
}
```

```
input[type="text"] {  
    width: 100%;  
    padding: 10px;  
    margin-bottom: 5px;  
    border-radius: 6px;  
    border: 1px solid #ccc;  
}  
.error {  
    color: red;  
    font-size: 12px;  
    margin-bottom: 12px;  
}  
input[type="submit"] {  
    width: 100%;  
    padding: 12px;  
    background: #3366cc;  
    border: none;  
    border-radius: 6px;  
    color: white;  
    font-weight: bold;  
    cursor: pointer;  
}  
.welcome {  
    background: #d4edda;  
    color: #155724;  
    padding: 15px;  
    border-radius: 8px;  
    text-align: center;  
    font-size: 18px;  
    font-weight: bold;  
}  
</style>
```

```

</head>

<body>

<div class="form-container">

    <%
        if (isSubmitted && isValid) {
    %>

        <div class="welcome">

            Congratulations <%= name %>!  
<br>Welcome to the system.

        </div>

    <%
        } else {
    %>

    <form method="post">

        <h2>Login Page</h2>

        <input type="text" name="name" placeholder="Enter Name" value="<%= name != null ? name : ""
            %>">

        <div class="error"><%= nameError %></div>

        <input type="text" name="mobile" placeholder="Enter Mobile Number" value="<%= mobile != null ?
            mobile : "" %>">

        <div class="error"><%= mobileError %></div>

        <input type="text" name="email" placeholder="Enter Email ID" value="<%= email != null ? email : ""
            %>">

        <div class="error"><%= emailError %></div>

        <input type="submit" name="submit" value="Login">

    </form>

    <%
        }
    %>

</div>

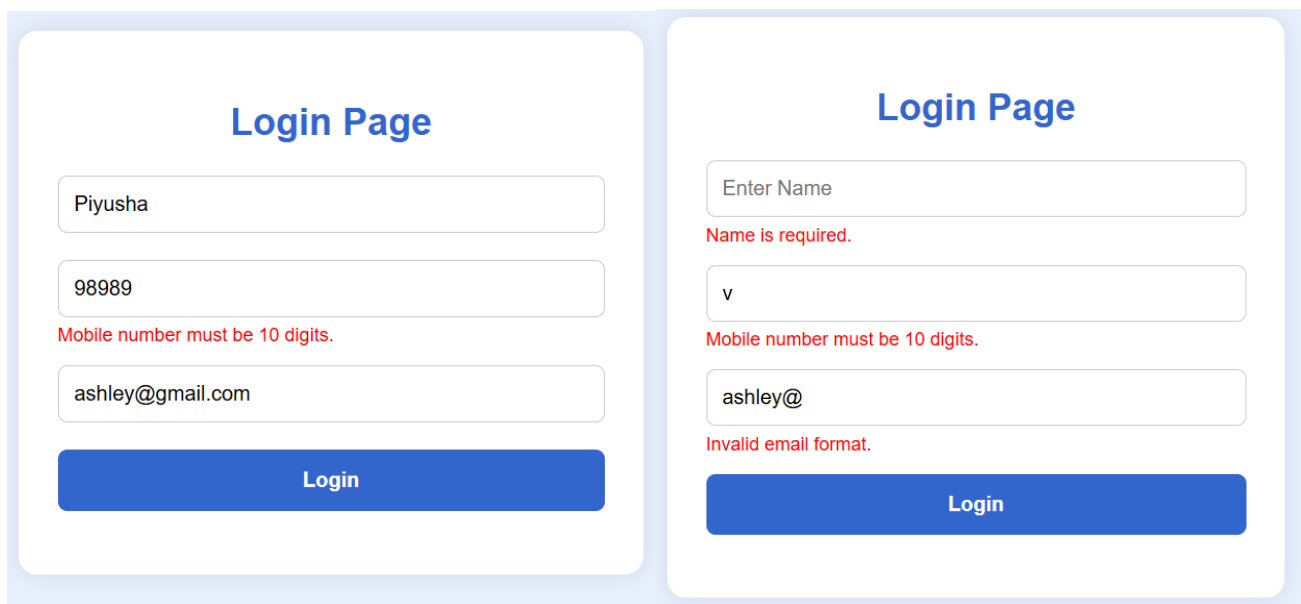
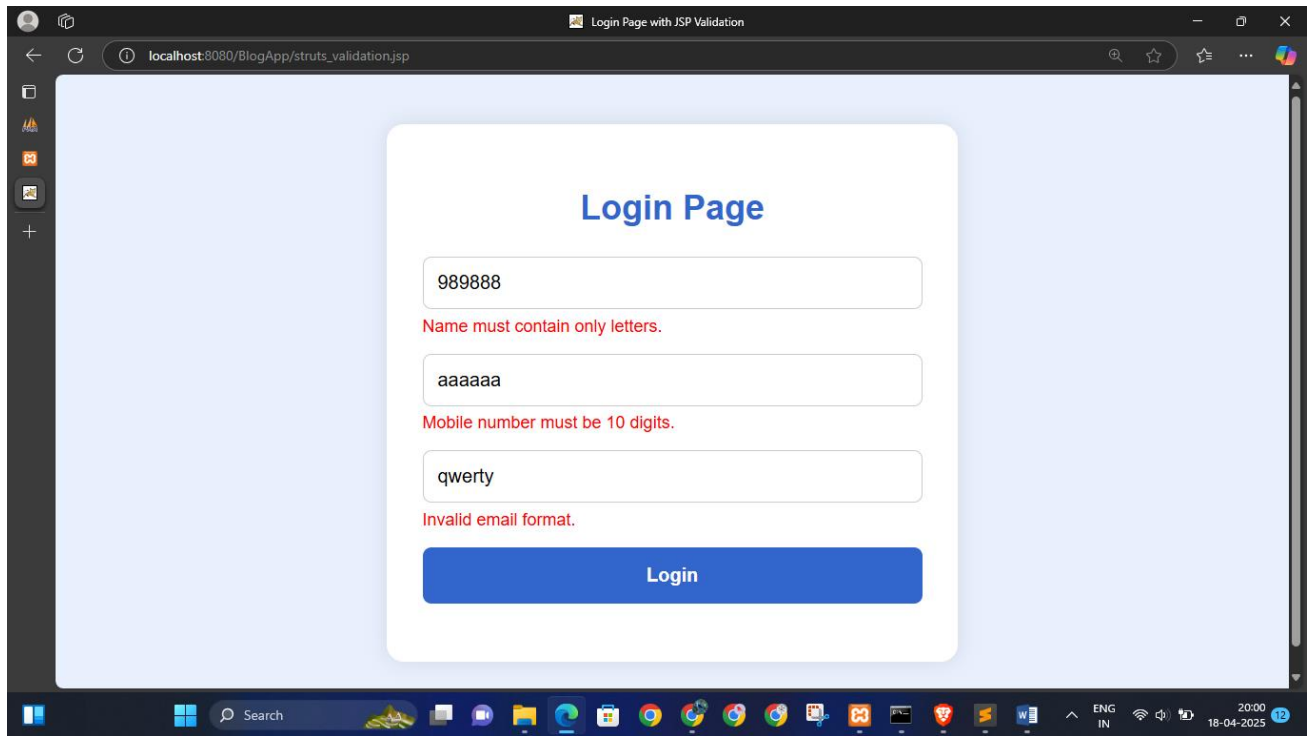
```

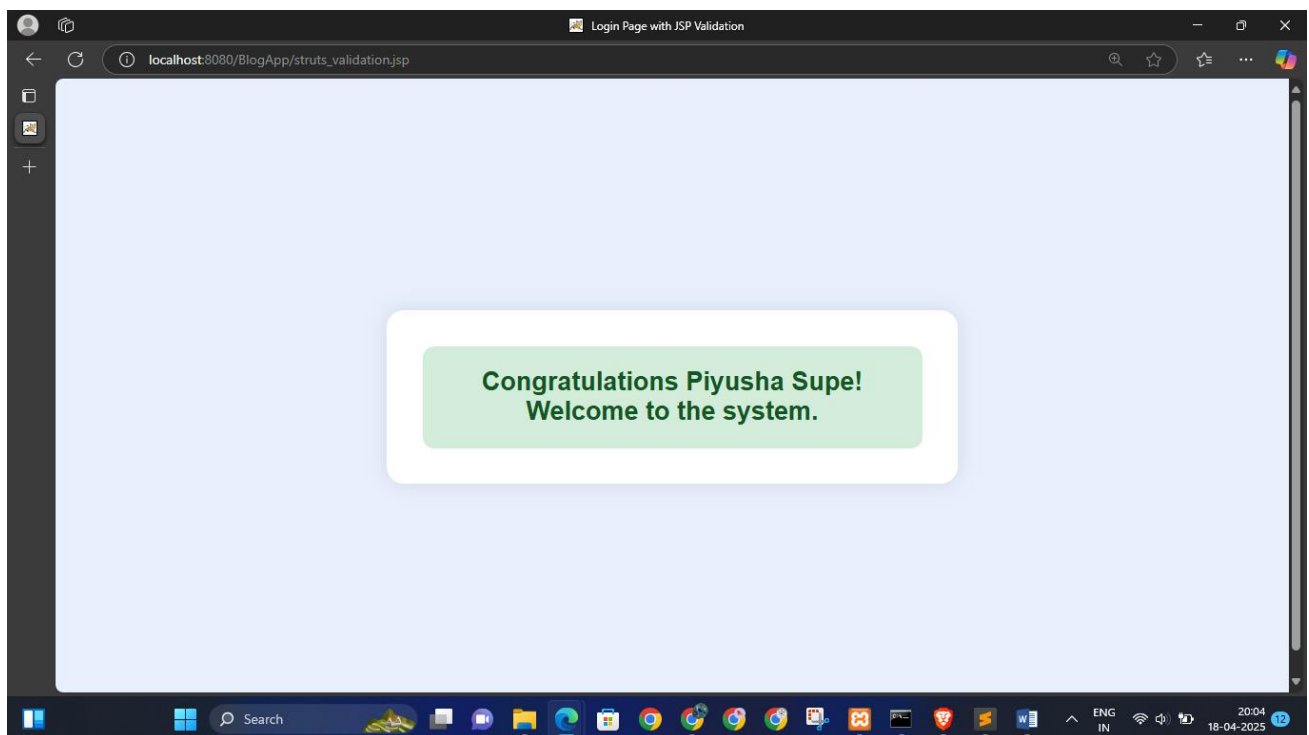
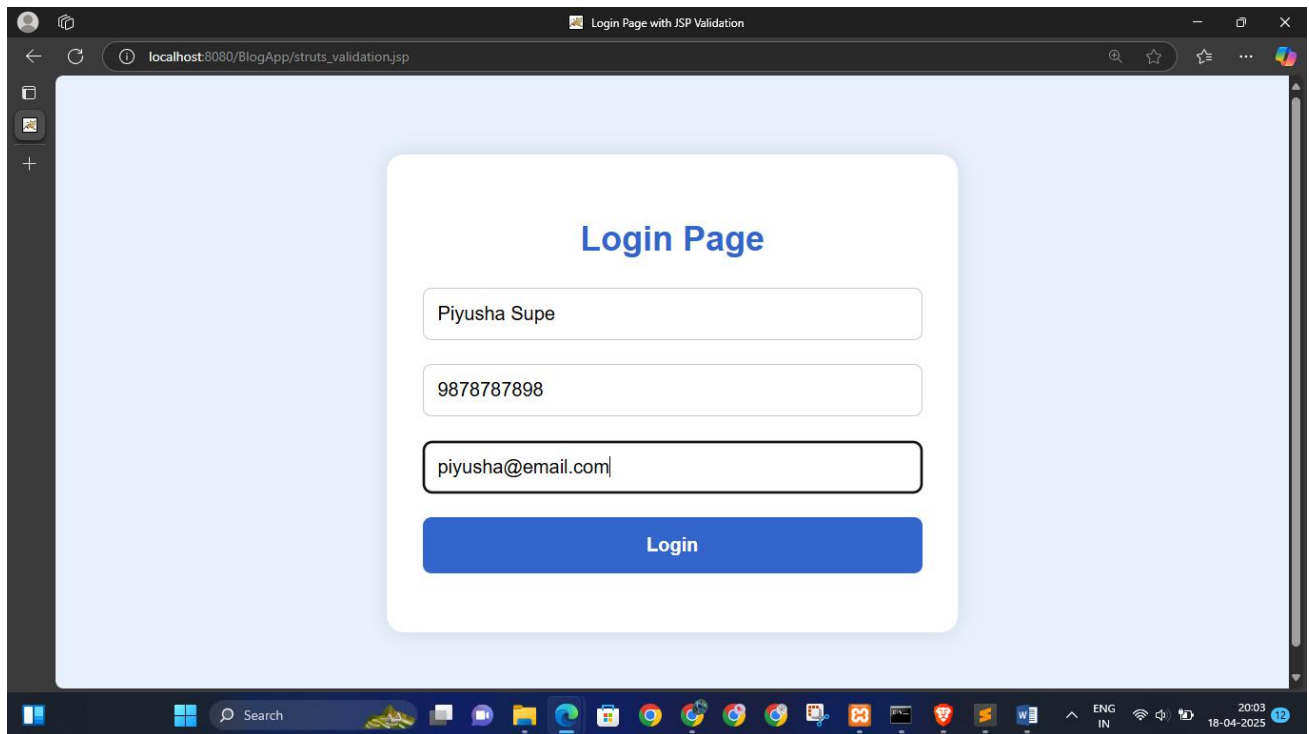


```
</body>
```

```
</html>
```

The output is as follows:





Conclusion: Thus in this way we have successfully configured the struts validation using the Apache struts framework