

JAVA SERVER PAGES (JSP)

1. GIỚI THIỆU	2
2. Bài tập 0:	3
3. Bài tập 1: Thực hiện với 3 cách	5
4. Bài tập 4	10
5. Bài tập 5:	14
6. JSP Action Tags	16
7. Java bean	17
8. Bài tập 6:	19

1. GIỚI THIỆU

Mã JSP (Scriptlets):

```
<% code java %>
```

Ví dụ in văn bản:

```
<% out.println("Welcome to JSP world"); %>
```

```
<% out.write("How are you?\n");%>
```

Biểu thức (Expressions):

```
<%= expression %>
```

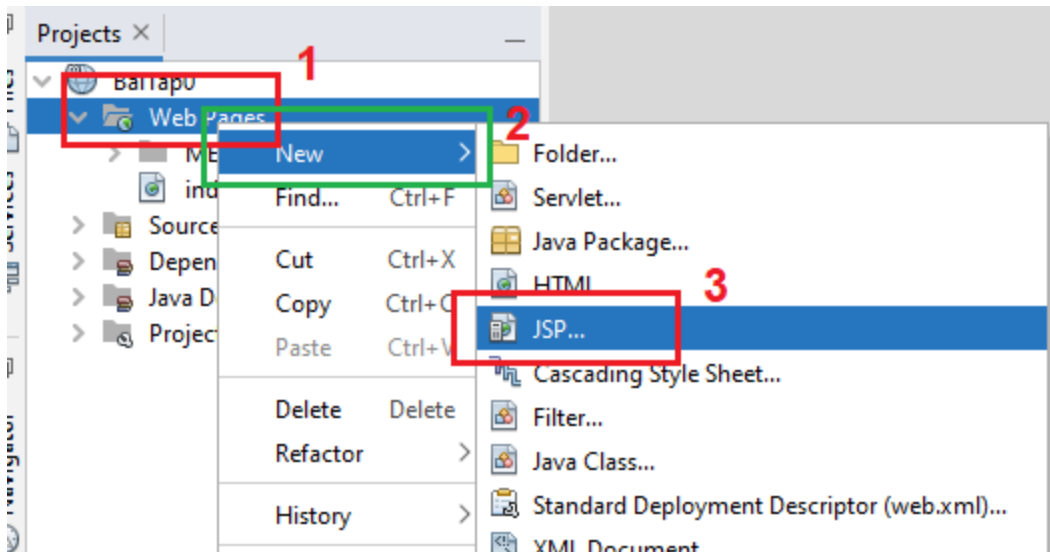
Bảng sau liệt kê 9 đối tượng ẩn mà JSP hỗ trợ:

No.	Đối tượng & Mô tả
1	request Đây là đối tượng HttpServletRequest được liên kết với request.
2	response Đây là đối tượng HttpServletResponse được liên kết với response tới client.
3	out Đây là đối tượng PrintWriter được sử dụng để gửi dữ liệu tới client.
4	session Đây là đối tượng HttpSession được liên kết với request.
5	application Đây là đối tượng ServletContext được liên kết với application context.
6	config Đây là đối tượng ServletConfig được liên kết với page.
7	pageContext Điều này đóng gói việc sử dụng các tính năng cụ thể của server như JspWriters với hiệu suất cao hơn.
8	page Đơn giản là một từ đồng nghĩa với this , được sử dụng để gọi các phương thức được định nghĩa bởi lớp serv được biên dịch.

No.	Đối tượng & Mô tả
9	exception Đối tượng exception cho phép các dữ liệu ngoại lệ (Exception) được truy cập bằng JSP được chỉ định.

2. Bài tập 0:

Thực hành đơn giản các lệnh in, gọi phương thức, phép tính cơ bản



Index.jsp

```
<body>
    <%
        String name = "Nguyễn Văn Tèo";
        int a = 10;
        int b = 20;
        out.println("<p>Xin chào, " + name + "!</p>");
        out.println("<p>a = " + a + "</p>");
        out.println("<p>b = " + b + "</p>");
        out.println("<p>Phép cộng " + a + " + " + b + " = " + (a + b) +
    "</p>");
        out.println("<p>Min: " + Math.min(a,b) + "</p>");
    %>
</body>
```

<http://localhost:9999/BaiTap0/index.jsp>

Xin chào, Nguyễn Văn Tèo!

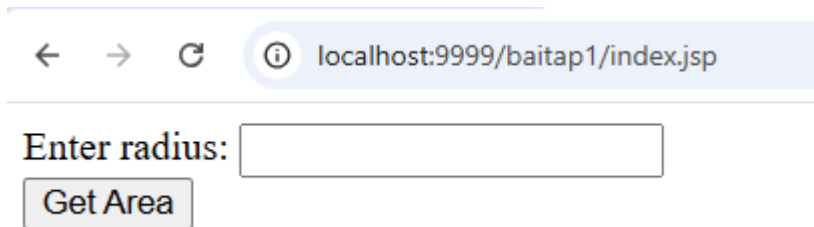
a = 10

b = 20

Phép cộng $10 + 20 = 30$

Min: 10

3. Bài tập 1: Thực hiện với 3 cách



Enter radius:

Get Area

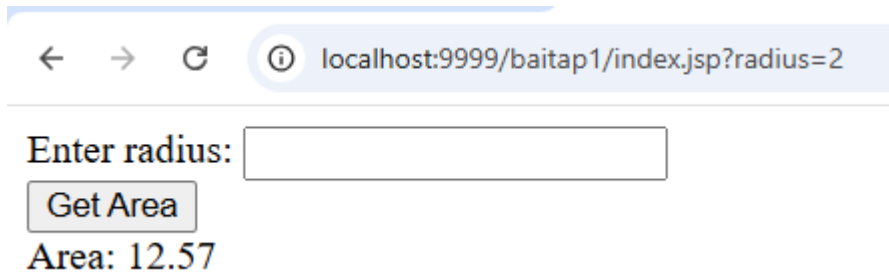
Cách 1: access to page via url: /index.jsp (dùng form + xử lý backend đều dùng jsp)

Tạo index.jsp

```
<%@page contentType="text/html" pageEncoding="UTF-8"%>
<%@page import = "java.text.DecimalFormat" %>

<!DOCTYPE html>
<html>
    <head>
        <meta http-equiv="Content-Type" content="text/html; charset=UTF-8">
        <title>JSP Page</title>
    </head>
    <body>
        <form>
            Enter radius:
            <input type="text" name="radius"><br/>
            <input type="submit" value="Get Area">
        </form>
        <%
            if(request.getParameter("radius") != null){
                String r = request.getParameter("radius");
                DecimalFormat f = new DecimalFormat("###.##");
                double radius, area;
                try {
                    radius = Double.parseDouble(r);
                    area = Math.PI * radius * radius;
                    out.println("Area: " + f.format(area));
                } catch (NumberFormatException e) {
                    System.out.println(e.getMessage());
                }
            }
        %>
    </body>
</html>
```

Kết quả:



← → ↻ ⓘ localhost:9999/baitap1/index.jsp?radius=2

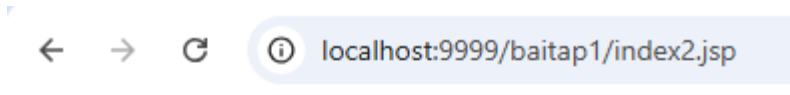
Enter radius:

Get Area

Area: 12.57

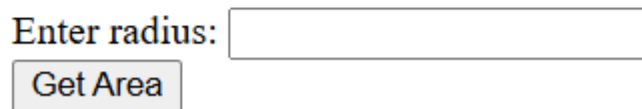
Cách 2: access to page via url: /index2.jsp (dùng form dùng jsp, xử lý backend dùng servlet)

Tạo index2.jsp, lưu ý thuộc tính action trong form



← → ↻ ⓘ localhost:9999/baitap1/index2.jsp

Cách 2



Enter radius:

Get Area

```
<%@page contentType="text/html" pageEncoding="UTF-8"%>
<%@page import = "java.text.DecimalFormat" %>
<!DOCTYPE html>
<html>
  <head>
    <meta http-equiv="Content-Type" content="text/html; charset=UTF-8">
    <title>JSP Page</title>
  </head>
  <body>
    <h1>Cách 2</h1>
    <form action="new2" method="post">
      Enter radius:
      <input type="text" name="radius"><br/>
      <input type="submit" value="Get Area">
    </form>
    <%
      if(request.getAttribute("a") != null){
        double area = (double)request.getAttribute("a");
        DecimalFormat f = new DecimalFormat("###.##");
        out.println("Area: " + f.format(area));
      }
    %>
  </body>
</html>
```

```

    }

    %>
</body>
</html>

```

Tạo Servlet:

☐ Add information to deployment descriptor (web.xml)

Class Name:

Servlet Name:

URL Pattern(s):

Trong doGet() thực hiện:

```

@Override
protected void doGet(HttpServletRequest request, HttpServletResponse
response)
    throws ServletException, IOException {
    request.getRequestDispatcher("index2.jsp").forward(request, response);
}

```

Trong doPost() thực hiện:

```

@Override
protected void doPost(HttpServletRequest request, HttpServletResponse
response)
    throws ServletException, IOException {
    String r = request.getParameter("radius");
    DecimalFormat f = new DecimalFormat("###.##");
    double radius, area;
    try {
        radius = Double.parseDouble(r);
        area = Math.PI * radius * radius;
        request.setAttribute("a", area);
        request.getRequestDispatcher("index2.jsp").forward(request,
response);
    } catch (NumberFormatException e) {
        System.out.println(e.getMessage());
    }
}

```

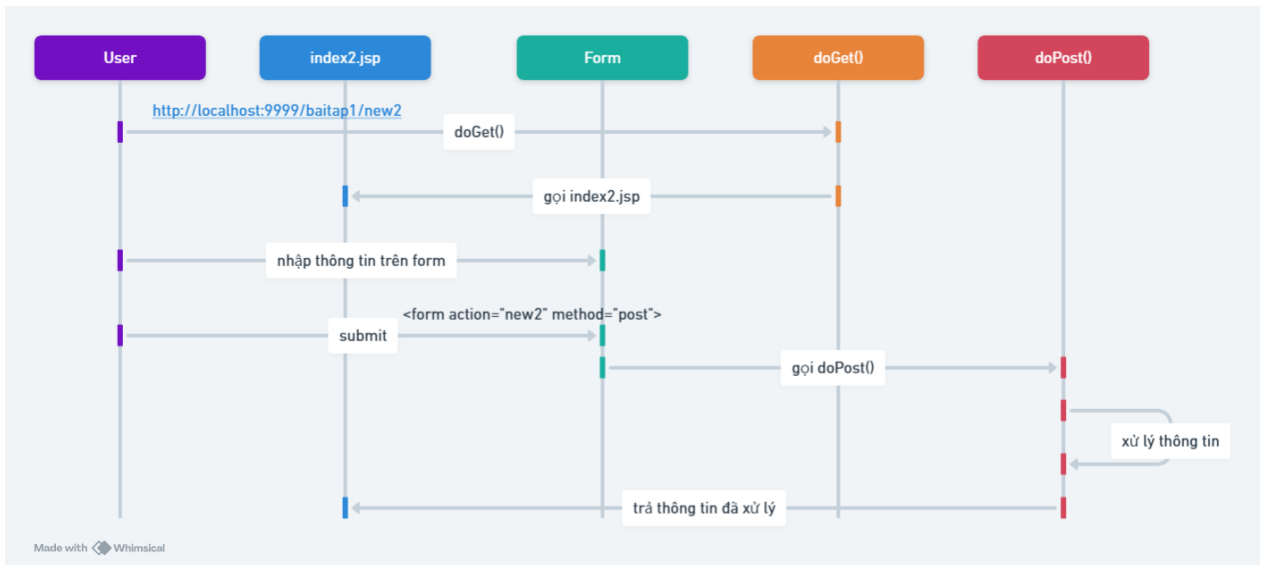
Test:

Có 2 cách truy cập index2.jsp

Cách 1: truy cập trực tiếp: <http://localhost:9999/baitap1/index2.jsp>

Cách 2: truy cập thông qua servlet: <http://localhost:9999/baitap1/new2>

(đầu tiên sẽ gọi doGet() → gọi index2.jsp → nhập thông tin trên form → submit → gọi doPost() → xử lý thông tin → index2.jsp)



Cách 3: access to page via url: /new2 (làm hoàn toàn bằng servlet)

Không tạo file .jsp nào hết (ví xử lý hoàn toàn bằng servlet)

Tạo servlet:

☐ Add information to deployment descriptor (web.xml)

Class Name:

Servlet Name:

URL Pattern(s):

Initialization Parameters:

Cập nhật doGet(), lưu ý thuộc tính action="new3" trong form

```
@Override
protected void doGet(HttpServletRequest request, HttpServletResponse response)
throws ServletException, IOException {
    PrintWriter out = response.getWriter();
}
```



```

        out.println("<!DOCTYPE html>");
        out.println("<html>");
        out.println("<head>");
        out.println("<meta http-equiv=\"Content-Type\" content=\"text/html; charset=UTF-8\">");
        out.println("<title>Servlet NewServlet3</title>");
        out.println("</head>");
        out.println("<body>");
        out.println("<h1>Cách 3</h1>\n"
            + "        <form action=\"new3\" method=\"post\">\n"
            + "            Enter radius:\n"
            + "            <input type=\"text\" name=\"radius\"><br/>\n"
            + "            <input type=\"submit\" value=\"Get Area\">\n"
            + "        </form>");

        if (request.getAttribute("a") != null) {
            double radius = (double) request.getAttribute("radius");
            double area = (double) request.getAttribute("a");

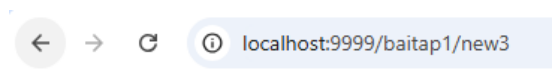
            out.println("<h2>Radius: " + radius + "</h2>");
            out.println("<h2>Area: " + new DecimalFormat("###.##").format(area) +
"</h2>");
        }

        if (request.getAttribute("error") != null) {
            out.println("<h3 style='color:red;'>" + request.getAttribute("error")
+ "</h3>");
        }

        out.println("</body>");
        out.println("</html>");
    }
}

```

Test gọi form: <http://localhost:9999/baitap1/new3>



Cách 3

Enter radius:

Cập nhật doPost()

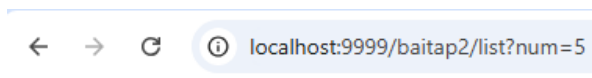
```

@Override
protected void doPost(HttpServletRequest request, HttpServletResponse
response)
    throws ServletException, IOException {
    String r = request.getParameter("radius");
    double radius, area;
    try {
        radius = Double.parseDouble(r);
        area = Math.PI * radius * radius;
        request.setAttribute("a", area);
        request.setAttribute("radius", radius);
        // Chuyển tiếp sang doGet để hiển thị
        doGet(request, response);
    } catch (NumberFormatException e) {
        // Đặt thông báo lỗi vào request
        request.setAttribute("error", "Invalid radius value. Please enter a
valid number.");
        // Chuyển tiếp sang doGet để hiển thị
        doGet(request, response);
    }
}






```

4. Bài tập 4

Tạo giao diện như sau:



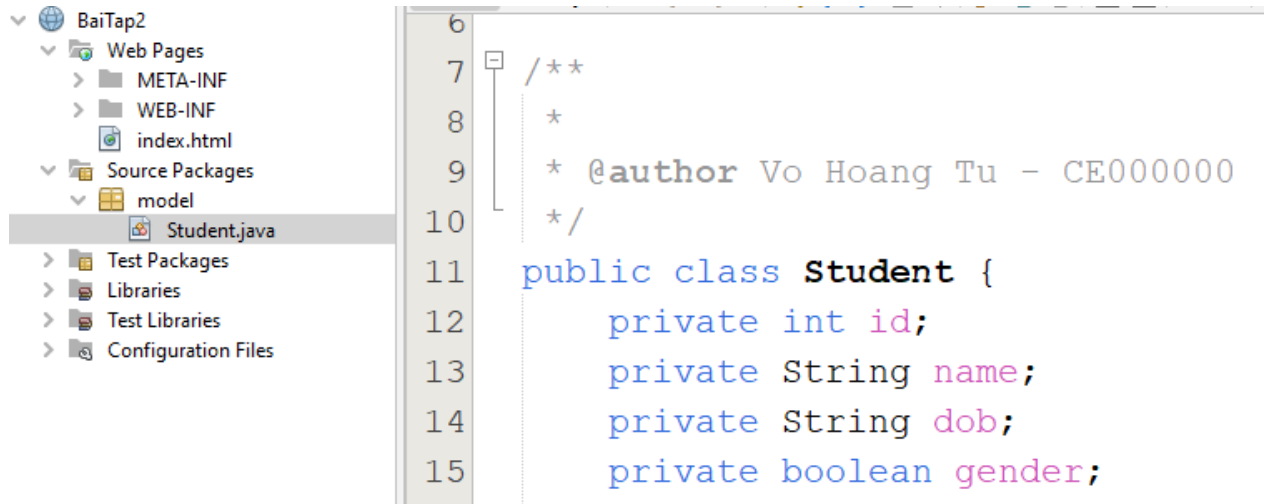
List of Students

Id	Name	Dob	Gender	Image
1	Mss.1	2024	false	
2	Mss.2	2024	false	
3	Mr.3	2024	true	
4	Mss.4	2024	false	
5	Mss.5	2024	false	

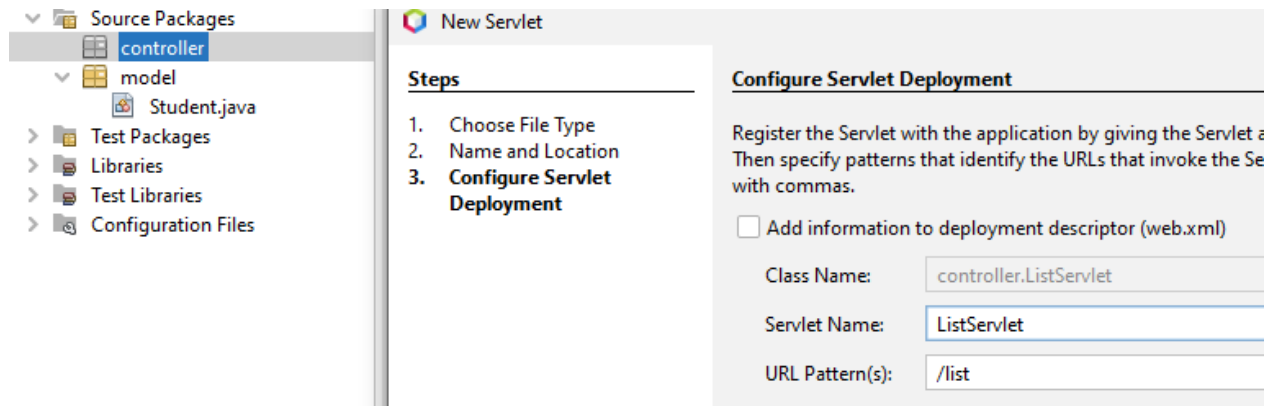
Danh sách được tạo ngẫu nhiên dựa vào tham số num trên url

Tạo model (model)

Tạo Class có tên Student và lưu trong package model



Tạo Servlet trong package controller



Cập nhật doGet()

```
@Override
protected void doGet(HttpServletRequest request, HttpServletResponse
response)
throws ServletException, IOException {
    String numRaw = request.getParameter("num");
    int num;
    try {
        num = Integer.parseInt(numRaw);
        List<Student> list = new ArrayList<>();
        int id;
        String name, dob;
        boolean gender;
        Random ran = new Random();
        for (int i = 0; i < num; i++) {
            id = i + 1;
            gender = ran.nextBoolean();
```

```

        if(gender){
            name = "Mr." + id;
        }else{
            name = "Mss." + id;
        }
        dob = "2024";
        list.add(new Student(id, name, dob, gender));
    }
    request.setAttribute("data", list);
    request.getRequestDispatcher("list.jsp").forward(request, response);
} catch (NumberFormatException e) {
    System.out.println(e.getMessage());
}
}

```

Tạo trang list.jsp

```

<%@page contentType="text/html" pageEncoding="UTF-8"%>
<%@page import="java.util.List" %>
<%@page import="model.Student" %>

<!DOCTYPE html>
<html>
    <head>
        <meta http-equiv="Content-Type" content="text/html; charset=UTF-8">
        <title>JSP Page</title>
    </head>
    <body>
        <h1>List of Students</h1>
        <table border="1">
            <tr>
                <td>Id</td>
                <td>Name</td>
                <td>Dob</td>
                <td>Gender</td>
                <td>Image</td>
            </tr>
            <%
                List<Student> list = (List<Student>)
request.getAttribute("data");

                for (Student i : list) {
                    %>
                    <tr>
                        <td><%= i.getId() %></td>
                        <td><%= i.getName() %></td>

```

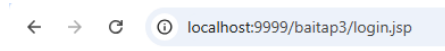
```
        <td><%= i.getDob() %></td>
        <td><%= i.isGender() %></td>
        <td>
            " width="30px" height="30px">
        </td>
    </tr>
    <%
    }
    %>

</table>
</body>
</html>
```

5. Bài tập 5:

Tạo form Login với username và password, các giá trị check cho chúng được khởi tạo trước: tên tài khoản của mình, password là 123. Nếu đăng nhập sai thì thông báo nhập sai (chữ màu đỏ) phía dưới form nhập. Nếu nhập đúng chuyển sang trang welcome.jsp

Tạo login.jsp



Login

Username:

Password:

```
<body>
  <h1>Login</h1>
  <form action="login" method="post">
    <label for="username">Username:</label>
    <input type="text" id="username" name="username" required><br><br>
    <label for="password">Password:</label>
    <input type="password" id="password" name="password"
required><br><br>
    <button type="submit">Login</button>
  </form>
  <%
String errorMessage = (String) request.getAttribute("errorMessage");
if (errorMessage != null) {
%>
  <p style="color: red;"><%= errorMessage %></p>
  <% } %>
</body>
```

Tạo Servlet:

☒ Add information to deployment descriptor (web.xml)

Class Name:	<input type="text" value="controller.LoginServlet"/>
Servlet Name:	<input type="text" value="LoginServlet"/>
URL Pattern(s):	<input type="text" value="/login"/>

Cập nhật doPost()

```

@Override
protected void doPost(HttpServletRequest request, HttpServletResponse
response)
throws ServletException, IOException {
    String username = request.getParameter("username");
    String password = request.getParameter("password");
    // Lấy thông tin username và password từ web.xml
    String correctUsername = "tuvh6";
    String correctPassword = "abc123";
    // Kiểm tra thông tin đăng nhập
    if (correctUsername.equals(username) && correctPassword.equals(password))
    {
        // Đăng nhập thành công
        response.sendRedirect("welcome.jsp");
    } else {
        // Đăng nhập thất bại
        request.setAttribute("errorMessage", "Sai username hoặc password!");
        request.getRequestDispatcher("login.jsp").forward(request, response);
    }
}
}

```

Tạo trang **welcome.jsp**

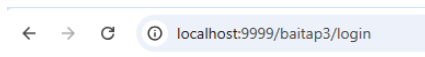
```

<!DOCTYPE html>
<html>
    <head>
        <meta http-equiv="Content-Type" content="text/html; charset=UTF-8">
        <title>JSP Page</title>
    </head>
    <body>
        <h1>Welcome, <%=
getServletContext().getInitParameter("username") %>!</h1>
        <p>You have successfully logged in.</p>
    </body>
</html>

```

Test: url: <http://localhost:9999/baitap3/login.jsp>

Test 1: Đăng nhập không thành công



Login

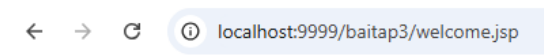
Username:

Password:

Login

Sai username hoặc password!

Test 2: Đăng nhập thành công



Welcome, tuvh6!

You have successfully logged in.

6. JSP Action Tags

JSP Action Tags	Description
jsp:forward	forwards the request and response to another resource.
jsp:include	includes another resource.
jsp:useBean	creates or locates bean object.
jsp:setProperty	sets the value of property in bean object.
jsp:getProperty	prints the value of property of the bean.
jsp:plugin	embeds another components such as applet.
jsp:param	sets the parameter value. It is used in forward and include mostly.

7. Java bean

JavaBean là một class có quy ước (convention) và được dùng với một mục đích đặc biệt, thường trong các ứng dụng cần giao tiếp dữ liệu. Trong JSP (JavaServer Pages): JSP có thể sử dụng JavaBean để dễ dàng truy cập và hiển thị dữ liệu từ server lên giao diện web.

```
<jsp:useBean id="student" class="HocSinh" scope="session" />
<jsp:setProperty name="student" property="ten" value="Nguyen Van A" />
<p>Học sinh: <jsp:getProperty name="student" property="ten" /></p>
```

Khi nào dùng class thường, khi nào dùng JavaBean?

- Dùng class thông thường khi chỉ cần xử lý logic hoặc xây dựng các chức năng không phụ thuộc vào chuẩn hóa dữ liệu.
- Dùng JavaBean khi cần lưu trữ và truyền dữ liệu, hoặc làm việc với các công cụ/tầng framework yêu cầu sử dụng JavaBean.

Một JavaBean là một lớp Java được xây dựng một cách đặc biệt trong Java và được mã hóa theo JavaBeans API Specifications. Sau đây là các đặc trưng duy nhất giúp phân biệt một JavaBean với các lớp khác trong Java:

- Nó cung cấp một constructor mặc định, không có tham số.
- Implements Serializable interface (để truyền hoặc lưu dữ liệu).
- Nó có thể có một số đặc tính mà có thể được đọc và được viết.
- Nó có thể có một số phương thức "getter" và "setter" cho các đặc tính.

<jsp:useBean> Action trong JSP

useBean action trong JSP là khá linh hoạt. Đầu tiên nó tìm kiếm một đối tượng đang tồn tại đang sử dụng thuộc tính id và scope. Nếu một đối tượng không được tìm thấy, thì nó cố gắng tạo đối tượng đã xác định đó.

Cách đơn giản nhất để tải một Bean như sau:

```
<jsp:useBean id="name" class="package.class" scope="page" />
```

Trong đó scope="page" là mặc định, ngoài ra “application”, “session” hoặc “request”

Ví dụ:

```
<jsp:useBean id="students" class = "bean.StudentsBean"/>  
Tương đương  
<% bean.StudentsBean students = new bean.StudentsBean(); %>
```

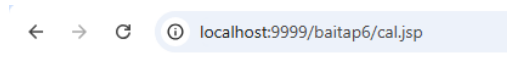
Khi lớp Bean đã được tải, có thể sử dụng các `<jsp:setProperty>` action và `<jsp:getProperty>` action để sửa đổi và thu nhận các đặc tính của Bean.

```
<jsp:getProperty name="students" property="firstName"/>  
Tương đương <%= students.getFirstName() %>  
<jsp:getProperty name="students" property="*" />  
<jsp:setProperty name="students" property="firstName" value="Ha"/>  
Tương đương <%= students.setFirstName("Ha") %>  
<jsp:setProperty name="students" property="*" />
```

Trong JavaBean, nếu một form đã thiết lập giá trị cho tất cả các thuộc tính của object, chúng ta chỉ cần dùng `property="*"` để tự động gán giá trị cho tất cả các thuộc tính đó.

8. Bài tập 6:

Tạo cal.jsp



A Simple Calculator

Enter Operator 1:

Enter Operator 2:

Select Operation + ▼

- ☒ +
- ☐ -
- ☐ *
- ☐ /

```
<%@page contentType="text/html" pageEncoding="UTF-8"%>
<!DOCTYPE html>
<html>
  <head>
    <meta http-equiv="Content-Type" content="text/html; charset=UTF-8">
    <title>JSP Page</title>
  </head>
  <body>
    <h1>A Simple Calculator</h1>
    <form>
      <table>
        <tr>
          <td>Enter Operator 1: </td>
          <td><input type="text" name="num1" /></td>
        </tr>
        <tr>
          <td>Enter Operator 2: </td>
          <td><input type="text" name="num2" /></td>
        </tr>
        <tr>
          <td>Select Operation </td>
          <td>
            <select name="op">
              <option value="+">+</option>
              <option value="-">-</option>
              <option value="*">*</option>
              <option value="/">/</option>
            </select>
          </td>
        </tr>
        <tr>
          <td><input type="submit" value="Calculate" /></td>
        </tr>
      </table>
    </form>
  </body>
</html>
```

```

        </tr>
    </table>
</form>
<jsp:useBean id = "math" class="models.Math" scope="request" />

<jsp:setProperty name="math" property="num1" param="num1" />
<jsp:setProperty name="math" property="num2" param="num2" />
<jsp:setProperty name="math" property="op" param="op" />

<h2><jsp:getProperty name="math" property="result" /></h2>

</body>
</html>

```

Khi sử dụng thẻ `<jsp:setProperty>` để gán giá trị từ form vào một thuộc tính của JavaBean, JSP sẽ cố gắng tự động chuyển đổi kiểu dữ liệu từ `String` sang kiểu của thuộc tính trong JavaBean.

Cách 2: sử dụng `property="*"`

```

<jsp:useBean id = "math" class="models.Math" scope="request" />

<jsp:setProperty name="math" property="*" />

<h2><jsp:getProperty name="math" property="result" /></h2>

```

Cách 3:

```

<%@page contentType="text/html" pageEncoding="UTF-8"%>
<!DOCTYPE html>
<html>
    <head>
        <meta http-equiv="Content-Type" content="text/html; charset=UTF-8">
        <title>JSP Page</title>
    </head>
    <body>
        <h1>A Simple Calculator</h1>
        <form>
            <table>
                <tr>
                    <td>Enter Operator 1: </td>
                    <td><input type= "text" name = "num1" /></td>
                </tr>
                <tr>

```

```

        <td>Enter Operator 2: </td>
        <td><input type= "text" name = "num2" /></td>
    </tr>
    <tr>
        <td>Select Operation </td>
        <td>
            <select name= "op">
                <option value="+">+</option>
                <option value="-">-</option>
                <option value="*">*</option>
                <option value="/">/</option>
            </select>
        </td>
    </tr>
    <tr>
        <td><input type="submit" value="Calculate" /></td>
    </tr>
</table>
</form>

<jsp:useBean id = "math" class="models.Math" scope="request" />

<%
    String rawNum1 = request.getParameter("num1");
    String rawNum2 = request.getParameter("num2");
    double num1 = 0;
    double num2 = 0;
    boolean check = true;

    // Kiểm tra cả num1 và num2 đều không phải null và không phải chuỗi
    rỗng
    if ((rawNum1 != null && !rawNum1.trim().isEmpty()) && (rawNum2 !=
    null && !rawNum2.trim().isEmpty())) {

        try {
            num1 = Double.parseDouble(rawNum1);
        } catch (NumberFormatException e) {
            out.println("Invalid input for num1. Please enter a valid
            number.");
            check = false;
        }

        try {
            num2 = Double.parseDouble(rawNum2);
        } catch (NumberFormatException e) {
            out.println("Invalid input for num2. Please enter a valid
            number.");

```

```

        check = false;
    }

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

    // Gán giá trị vào JavaBean
    math.setNum1(num1);
    math.setNum2(num2);
    math.setOp(op);
} else {
    out.println("Please provide valid inputs for both num1 and
num2.");
    check = false;
}
%>

<%
// Kiểm tra nếu các tham số hợp lệ và không có lỗi, mới hiển thị kết
quả
if (check) {
%>
    <h2><jsp:getProperty name="math" property="result" /></h2>
<%
}
%>

</body>
</html>

```

Hoặc muốn in ra các thuộc tính:

```

<h2>
    <jsp:getProperty name="math" property="num1" />
    <jsp:getProperty name="math" property="op" />
    <jsp:getProperty name="math" property="num2" />
</h2>

```

4.0+5.0