



Національний технічний університет України
“Київський політехнічний інститут імені Ігоря Сікорського”
Факультет інформатики та обчислювальної техніки
Кафедра інформаційні систем та технологій

ЛАБОРАТОРНА РОБОТА №3

з дисципліни «Основи програмування - 2»

Тема: «JSP»

Виконали:

студенти групи ІА-31
Клим'юк В.Л, Самелюк А.С,
Дук М.Д, Сакун Д.С

Перевірив:

асистент кафедри ІСТ
Степанов А. С.

Тема: JSP

Мета: Розробка розуміння технології JavaServer Pages (JSP) та її застосування для створення динамічних веб-сторінок у середовищі Java EE.

Хід роботи

1. Пригадати як використовувати JSP для створення динамічних веб-сторінок.
2. Для свого варіанту(№1) з л/р №2 першого семестру створити JSP-сторінку з HTML-формою для введення значень параметрів для обчислення формули (початкове значення, кінцеве значення, крок). При першому відкритті сторінки в полях параметрів мають стояти значення за замовчуванням. При другому та наступних запусках – значення параметрів, що вводились минулого разу. При натисканні кнопки «обчислити» на сторінці має з'явитись таблиця, яка містить значення параметрів та результат обчислення функції. В процесі виконання завдання потрібно продемонструвати використання таких JSP-елементів: scriptlets, expressions, declarations.

```
<%@ page contentType="text/html; charset=UTF-8" pageEncoding="UTF-8" %>
<!DOCTYPE html>
<html>
<head>
    <title>Form 1</title>
    <link rel="stylesheet" href="css/form.css">
</head>
<body>

<%!
    public double calculateResult(double a, double b, double c, double d) {
        return Math.sqrt(Math.abs(Math.sin(a) - 4.0 * Math.log(b) / Math.pow(c, d)));
    }
%>
<%
    String from_a = "0";
    String to_a = "0";
    String step_a = "0";

    String from_b = "0";
    String to_b = "0";
    String step_b = "0";

    String from_c = "0";
    String to_c = "0";
    String step_c = "0";
```

```

String from_d = "0";
String to_d = "0";
String step_d = "0";

String newValueFA = request.getParameter("from_a");
String newValueTA = request.getParameter("to_a");
String newValueSA = request.getParameter("step_a");

String newValueFB = request.getParameter("from_b");
String newValueTB = request.getParameter("to_b");
String newValueSB = request.getParameter("step_b");

String newValueFC = request.getParameter("from_c");
String newValueTC = request.getParameter("to_c");
String newValueSC = request.getParameter("step_c");

String newValueFD = request.getParameter("from_d");
String newValueTD = request.getParameter("to_d");
String newValueSD = request.getParameter("step_d");

if (newValueFA != null && !newValueFA.isEmpty()) {
    from_a = newValueFA;
    Cookie cookieFA = new Cookie("from_a", from_a);
    response.addCookie(cookieFA);
}
if (newValueTA != null && !newValueTA.isEmpty()) {
    to_a = newValueTA;
    Cookie cookieTA = new Cookie("to_a", to_a);
    response.addCookie(cookieTA);
}
if (newValueSA != null && !newValueSA.isEmpty()) {
    step_a = newValueSA;
    Cookie cookieSA = new Cookie("step_a", step_a);
    response.addCookie(cookieSA);
}

if (newValueFB != null && !newValueFB.isEmpty()) {
    from_b = newValueFB;
    Cookie cookieFB = new Cookie("from_b", from_b);
    response.addCookie(cookieFB);
}
if (newValueTB != null && !newValueTB.isEmpty()) {
    to_b = newValueTB;
    Cookie cookieTB = new Cookie("to_b", to_b);
    response.addCookie(cookieTB);
}
if (newValueSB != null && !newValueSB.isEmpty()) {
    step_b = newValueSB;
    Cookie cookieSB = new Cookie("step_b", step_b);
    response.addCookie(cookieSB);
}

if (newValueFC != null && !newValueFC.isEmpty()) {
    from_c = newValueFC;
    Cookie cookieFC = new Cookie("from_c", from_c);
    response.addCookie(cookieFC);
}
if (newValueTC != null && !newValueTC.isEmpty()) {
    to_c = newValueTC;
    Cookie cookieTC = new Cookie("to_c", to_c);
    response.addCookie(cookieTC);
}
if (newValueSC != null && !newValueSC.isEmpty()) {
    step_c = newValueSC;

```

```

        Cookie cookieSC = new Cookie("step_c", step_c);
        response.addCookie(cookieSC);
    }

    if (newValueFD != null && !newValueFD.isEmpty()) {
        from_d = newValueFD;
        Cookie cookieFD = new Cookie("from_d", from_d);
        response.addCookie(cookieFD);
    }
    if (newValueTD != null && !newValueTD.isEmpty()) {
        to_d = newValueTD;
        Cookie cookieTD = new Cookie("to_d", to_d);
        response.addCookie(cookieTD);
    }
    if (newValueSD != null && !newValueSD.isEmpty()) {
        step_d = newValueSD;
        Cookie cookieSD = new Cookie("step_d", step_d);
        response.addCookie(cookieSD);
    }
}

%>

<div class="main-container">
    <div class="image-container">
        
    </div>
    <form class="calculator-container" method="post">
        <div class="calculator-field">
            <div class="calculator-field-name">a</div>
            <div class="calculator-field-from">from:</div>
            <input class="calculator-field-input-from" type="number"
value="<%=from_a%>" step="0.01" name="from_a"/>
            <div class="calculator-field-to">to:</div>
            <input class="calculator-field-input-to" type="number"
value="<%=to_a%>" step="0.01" name="to_a"/>
            <div class="calculator-field-step">step:</div>
            <input class="calculator-field-input-step" type="number"
value="<%=step_a%>" step="0.01" name="step_a"/>
        </div>
        <div class="calculator-field">
            <div class="calculator-field-name">b</div>
            <div class="calculator-field-from">from:</div>
            <input class="calculator-field-input-from" type="number"
value="<%=from_b%>" step="0.01" name="from_b"/>
            <div class="calculator-field-to">to:</div>
            <input class="calculator-field-input-to" type="number"
value="<%=to_b%>" step="0.01" name="to_b"/>
            <div class="calculator-field-step">step:</div>
            <input class="calculator-field-input-step" type="number"
value="<%=step_b%>" step="0.01" name="step_b"/>
        </div>
        <div class="calculator-field">
            <div class="calculator-field-name">c</div>
            <div class="calculator-field-from">from:</div>
            <input class="calculator-field-input-from" type="number"
value="<%=from_c%>" step="0.01" name="from_c"/>
            <div class="calculator-field-to">to:</div>
            <input class="calculator-field-input-to" type="number"
value="<%=to_c%>" step="0.01" name="to_c"/>
            <div class="calculator-field-step">step:</div>
            <input class="calculator-field-input-step" type="number"
value="<%=step_c%>" step="0.01" name="step_c"/>
        </div>
        <div class="calculator-field">
            <div class="calculator-field-name">d</div>

```

```

        <div class="calculator-field-from">from:</div>
        <input class="calculator-field-input-from" type="number"
value="<%=from_d%>" step="0.01" name="from_d"/>
        <div class="calculator-field-to">to:</div>
        <input class="calculator-field-input-to" type="number"
value="<%=to_d%>" step="0.01" name="to_d"/>
        <div class="calculator-field-step">step:</div>
        <input class="calculator-field-input-step" type="number"
value="<%=step_d%>" step="0.01" name="step_d"/>
    </div>
    <div class="calculator-field-submit">
        <button class="calculator-submit-button" type="submit">
            Calculate
        </button>
    </div>
</form>
<div class="results-container">
    <table>
        <thead>
            <tr>
                <th>a</th>
                <th>b</th>
                <th>c</th>
                <th>d</th>
                <th>res</th>
            </tr>
        <tbody>
<%
        if (request.getMethod().equalsIgnoreCase("post")) {
            double from_A ;
            double from_B ;
            double from_C ;
            double from_D ;

            double to_A ;
            double to_B ;
            double to_C ;
            double to_D ;

            double step_A ;
            double step_B ;
            double step_C ;
            double step_D ;

            try {
                from_A = Double.parseDouble(request.getParameter("from_a"));
                from_B = Double.parseDouble(request.getParameter("from_b"));
                from_C = Double.parseDouble(request.getParameter("from_c"));
                from_D = Double.parseDouble(request.getParameter("from_d"));

                to_A = Double.parseDouble(request.getParameter("to_a"));
                to_B = Double.parseDouble(request.getParameter("to_b"));
                to_C = Double.parseDouble(request.getParameter("to_c"));
                to_D = Double.parseDouble(request.getParameter("to_d"));

                step_A = Double.parseDouble(request.getParameter("step_a"));
                step_B = Double.parseDouble(request.getParameter("step_b"));
                step_C = Double.parseDouble(request.getParameter("step_c"));
                step_D = Double.parseDouble(request.getParameter("step_d"));

            } catch (NumberFormatException e) {
                response.sendError(400, "Дані введені не коректно");
                return;
            }

```

```

        if (from_C==0 || to_C==0) {
            response.sendError(400, "Дані введені не коректно");
            return;
        }

        for (double a = from_A; a <= to_A; a += step_A) {
            for (double b = from_B; b <= to_B; b += step_B) {
                for (double c = from_C; c <= to_C; c += step_C) {
                    for (double d = from_D; d <= to_D; d += step_D) {
                        double res = calculateResult( a, b, c, d);
                    }
                }
            }
        }

        %>
        <tr>
            <td><%= a %></td>
            <td><%= b %></td>
            <td><%= c %></td>
            <td><%= d %></td>
            <td><%= res %></td>
        </tr>

    <%
    }
    }
    }
    }
    %>
</tbody>
</table>
</div>
</div>
</body>
</html>

```

Код 1.1 – form1.jsp

```

<%@ page contentType="text/html; charset=UTF-8" pageEncoding="UTF-8" %>
<!DOCTYPE html>
<html>
<head>
    <title>Form 2</title>
    <link rel="stylesheet" href="css/form.css">
</head>
<body>

<%!
    public double calculateResult(double a, double b, double c, double d) {
        return (Math.exp(a) + 3 * Math.log10(c) * Math.abs(Math.atan(d))) /
(Math.pow(b, 1 / c));
    }
%>
<%
    String from_a = "0";
    String to_a = "0";
    String step_a = "0";

    String from_b = "0";
    String to_b = "0";
    String step_b = "0";

    String from_c = "0";
    String to_c = "0";

```

```

String step_c = "0";

String from_d = "0";
String to_d = "0";
String step_d = "0";

String newValueFA = request.getParameter("from_a");
String newValueTA = request.getParameter("to_a");
String newValueSA = request.getParameter("step_a");

String newValueFB = request.getParameter("from_b");
String newValueTB = request.getParameter("to_b");
String newValueSB = request.getParameter("step_b");

String newValueFC = request.getParameter("from_c");
String newValueTC = request.getParameter("to_c");
String newValueSC = request.getParameter("step_c");

String newValueFD = request.getParameter("from_d");
String newValueTD = request.getParameter("to_d");
String newValueSD = request.getParameter("step_d");

if (newValueFA != null && !newValueFA.isEmpty()) {
    from_a = newValueFA;
    Cookie cookieFA = new Cookie("from_a", from_a);
    response.addCookie(cookieFA);
}
if (newValueTA != null && !newValueTA.isEmpty()) {
    to_a = newValueTA;
    Cookie cookieTA = new Cookie("to_a", to_a);
    response.addCookie(cookieTA);
}
if (newValueSA != null && !newValueSA.isEmpty()) {
    step_a = newValueSA;
    Cookie cookieSA = new Cookie("step_a", step_a);
    response.addCookie(cookieSA);
}

if (newValueFB != null && !newValueFB.isEmpty()) {
    from_b = newValueFB;
    Cookie cookieFB = new Cookie("from_b", from_b);
    response.addCookie(cookieFB);
}
if (newValueTB != null && !newValueTB.isEmpty()) {
    to_b = newValueTB;
    Cookie cookieTB = new Cookie("to_b", to_b);
    response.addCookie(cookieTB);
}
if (newValueSB != null && !newValueSB.isEmpty()) {
    step_b = newValueSB;
    Cookie cookieSB = new Cookie("step_b", step_b);
    response.addCookie(cookieSB);
}

if (newValueFC != null && !newValueFC.isEmpty()) {
    from_c = newValueFC;
    Cookie cookieFC = new Cookie("from_c", from_c);
    response.addCookie(cookieFC);
}
if (newValueTC != null && !newValueTC.isEmpty()) {
    to_c = newValueTC;
    Cookie cookieTC = new Cookie("to_c", to_c);
    response.addCookie(cookieTC);
}

```

```

    if (newValueSC != null && !newValueSC.isEmpty()) {
        step_c = newValueSC;
        Cookie cookieSC = new Cookie("step_c", step_c);
        response.addCookie(cookieSC);
    }

    if (newValueFD != null && !newValueFD.isEmpty()) {
        from_d = newValueFD;
        Cookie cookieFD = new Cookie("from_d", from_d);
        response.addCookie(cookieFD);
    }
    if (newValueTD != null && !newValueTD.isEmpty()) {
        to_d = newValueTD;
        Cookie cookieTD = new Cookie("to_d", to_d);
        response.addCookie(cookieTD);
    }
    if (newValueSD != null && !newValueSD.isEmpty()) {
        step_d = newValueSD;
        Cookie cookieSD = new Cookie("step_d", step_d);
        response.addCookie(cookieSD);
    }
%>
<div class="main-container">
    <div class="image-container">
        
    </div>
    <form class="calculator-container" method="post">
        <div class="calculator-field">
            <div class="calculator-field-name">a</div>
            <div class="calculator-field-from">from:</div>
            <input class="calculator-field-input-from" type="number"
value="<%=from_a%>" step="0.01" name="from_a"/>
            <div class="calculator-field-to">to:</div>
            <input class="calculator-field-input-to" type="number" value="<%=to_a%>"
step="0.01" name="to_a"/>
            <div class="calculator-field-step">step:</div>
            <input class="calculator-field-input-step" type="number"
value="<%=step_a%>" step="0.01" name="step_a"/>
        </div>
        <div class="calculator-field">
            <div class="calculator-field-name">b</div>
            <div class="calculator-field-from">from:</div>
            <input class="calculator-field-input-from" type="number"
value="<%=from_b%>" step="0.01" name="from_b"/>
            <div class="calculator-field-to">to:</div>
            <input class="calculator-field-input-to" type="number" value="<%=to_b%>"
step="0.01" name="to_b"/>
            <div class="calculator-field-step">step:</div>
            <input class="calculator-field-input-step" type="number"
value="<%=step_b%>" step="0.01" name="step_b"/>
        </div>
        <div class="calculator-field">
            <div class="calculator-field-name">c</div>
            <div class="calculator-field-from">from:</div>
            <input class="calculator-field-input-from" type="number"
value="<%=from_c%>" step="0.01" name="from_c"/>
            <div class="calculator-field-to">to:</div>
            <input class="calculator-field-input-to" type="number" value="<%=to_c%>"
step="0.01" name="to_c"/>
            <div class="calculator-field-step">step:</div>
            <input class="calculator-field-input-step" type="number"
value="<%=step_c%>" step="0.01" name="step_c"/>
        </div>
        <div class="calculator-field">

```



```

        <div class="calculator-field-name">d</div>
        <div class="calculator-field-from">from:</div>
        <input class="calculator-field-input-from" type="number"
value="<%=from_d%>" step="0.01" name="from_d"/>
        <div class="calculator-field-to">to:</div>
        <input class="calculator-field-input-to" type="number" value="<%=to_d%>"
step="0.01" name="to_d"/>
        <div class="calculator-field-step">step:</div>
        <input class="calculator-field-input-step" type="number"
value="<%=step_d%>" step="0.01" name="step_d"/>
    </div>
    <div class="calculator-field-submit">
        <button class="calculator-submit-button" type="submit">
            Calculate
        </button>
    </div>
</form>
<div class="results-container">
    <table>
        <thead>
            <tr>
                <th>a</th>
                <th>b</th>
                <th>c</th>
                <th>d</th>
                <th>res</th>
            </tr>
        </thead>
        <tbody>
<%=
    if (request.getMethod().equalsIgnoreCase("post")) {
        double from_A ;
        double from_B ;
        double from_C ;
        double from_D ;

        double to_A ;
        double to_B ;
        double to_C ;
        double to_D ;

        double step_A ;
        double step_B ;
        double step_C ;
        double step_D ;

        try {
            from_A = Double.parseDouble(request.getParameter("from_a"));
            from_B = Double.parseDouble(request.getParameter("from_b"));
            from_C = Double.parseDouble(request.getParameter("from_c"));
            from_D = Double.parseDouble(request.getParameter("from_d"));

            to_A = Double.parseDouble(request.getParameter("to_a"));
            to_B = Double.parseDouble(request.getParameter("to_b"));
            to_C = Double.parseDouble(request.getParameter("to_c"));
            to_D = Double.parseDouble(request.getParameter("to_d"));

            step_A = Double.parseDouble(request.getParameter("step_a"));
            step_B = Double.parseDouble(request.getParameter("step_b"));
            step_C = Double.parseDouble(request.getParameter("step_c"));
            step_D = Double.parseDouble(request.getParameter("step_d"));

        }catch(NumberFormatException e){
            response.sendError(400, "Дані введені не коректно");
            return;

```

```

    }

    if (from_C==0 || to_C==0) {
        response.sendError(400, "Дані введені не коректно");
        return;
    }
    if (from_B==0 || to_B==0) {
        response.sendError(400, "Дані введені не коректно");
        return;
    }

    for (double a = from_A; a <= to_A; a += step_A) {
        for (double b = from_B; b <= to_B; b += step_B) {
            for (double c = from_C; c <= to_C; c += step_C) {
                for (double d = from_D; d <= to_D; d += step_D) {
                    double res = calculateResult(a, b, c, d);
                }
            }
        }
    }

    %>

    <tr>
        <td><%= a %></td>
        <td><%= b %></td>
        <td><%= c %></td>
        <td><%= d %></td>
        <td><%= res %></td>
    </tr>

    <%
    }
    }
    }
    }
    %>
    </tbody>
    </table>
</div>
</div>
</body>
</html>

```

Код 1.2 – form2.jsp

```

<%@ page contentType="text/html; charset=UTF-8" pageEncoding="UTF-8" %>
<!DOCTYPE html>
<html>
<head>
    <title>Form 3</title>
    <link rel="stylesheet" href="css/form.css">
</head>
<body>

<%!
    public double calculateResult(double a, double b, double c, double d) {
        return Math.pow(2.0 * Math.sin(a) + Math.cos(Math.abs(b * Math.sqrt(c))), d);
    }
%>

<%
    String from_a = "0";
    String to_a = "0";
    String step_a = "0";

```

```

String from_b = "0";
String to_b = "0";
String step_b = "0";

String from_c = "0";
String to_c = "0";
String step_c = "0";

String from_d = "0";
String to_d = "0";
String step_d = "0";

String newValueFA = request.getParameter("from_a");
String newValueTA = request.getParameter("to_a");
String newValueSA = request.getParameter("step_a");

String newValueFB = request.getParameter("from_b");
String newValueTB = request.getParameter("to_b");
String newValueSB = request.getParameter("step_b");

String newValueFC = request.getParameter("from_c");
String newValueTC = request.getParameter("to_c");
String newValueSC = request.getParameter("step_c");

String newValueFD = request.getParameter("from_d");
String newValueTD = request.getParameter("to_d");
String newValueSD = request.getParameter("step_d");

if (newValueFA != null && !newValueFA.isEmpty()) {
    from_a = newValueFA;
    Cookie cookieFA = new Cookie("from_a", from_a);
    response.addCookie(cookieFA);
}
if (newValueTA != null && !newValueTA.isEmpty()) {
    to_a = newValueTA;
    Cookie cookieTA = new Cookie("to_a", to_a);
    response.addCookie(cookieTA);
}
if (newValueSA != null && !newValueSA.isEmpty()) {
    step_a = newValueSA;
    Cookie cookieSA = new Cookie("step_a", step_a);
    response.addCookie(cookieSA);
}

if (newValueFB != null && !newValueFB.isEmpty()) {
    from_b = newValueFB;
    Cookie cookieFB = new Cookie("from_b", from_b);
    response.addCookie(cookieFB);
}
if (newValueTB != null && !newValueTB.isEmpty()) {
    to_b = newValueTB;
    Cookie cookieTB = new Cookie("to_b", to_b);
    response.addCookie(cookieTB);
}
if (newValueSB != null && !newValueSB.isEmpty()) {
    step_b = newValueSB;
    Cookie cookieSB = new Cookie("step_b", step_b);
    response.addCookie(cookieSB);
}

if (newValueFC != null && !newValueFC.isEmpty()) {
    from_c = newValueFC;
    Cookie cookieFC = new Cookie("from_c", from_c);
    response.addCookie(cookieFC);
}

```

```

    }
    if (newValueTC != null && !newValueTC.isEmpty()) {
        to_c = newValueTC;
        Cookie cookieTC = new Cookie("to_c", to_c);
        response.addCookie(cookieTC);
    }
    if (newValueSC != null && !newValueSC.isEmpty()) {
        step_c = newValueSC;
        Cookie cookieSC = new Cookie("step_c", step_c);
        response.addCookie(cookieSC);
    }

    if (newValueFD != null && !newValueFD.isEmpty()) {
        from_d = newValueFD;
        Cookie cookieFD = new Cookie("from_d", from_d);
        response.addCookie(cookieFD);
    }
    if (newValueTD != null && !newValueTD.isEmpty()) {
        to_d = newValueTD;
        Cookie cookieTD = new Cookie("to_d", to_d);
        response.addCookie(cookieTD);
    }
    if (newValueSD != null && !newValueSD.isEmpty()) {
        step_d = newValueSD;
        Cookie cookieSD = new Cookie("step_d", step_d);
        response.addCookie(cookieSD);
    }
}
%>
<div class="main-container">
    <div class="image-container">
        
    </div>
    <form class="calculator-container" method="post">
        <div class="calculator-field">
            <div class="calculator-field-name">a</div>
            <div class="calculator-field-from">from:</div>
            <input class="calculator-field-input-from" type="number"
value="<%=from_a%>" step="0.01" name="from_a"/>
            <div class="calculator-field-to">to:</div>
            <input class="calculator-field-input-to" type="number" value="<%=to_a%>"
step="0.01" name="to_a"/>
            <div class="calculator-field-step">step:</div>
            <input class="calculator-field-input-step" type="number"
value="<%=step_a%>" step="0.01" name="step_a"/>
            </div>
            <div class="calculator-field">
                <div class="calculator-field-name">b</div>
                <div class="calculator-field-from">from:</div>
                <input class="calculator-field-input-from" type="number"
value="<%=from_b%>" step="0.01" name="from_b"/>
                <div class="calculator-field-to">to:</div>
                <input class="calculator-field-input-to" type="number" value="<%=to_b%>"
step="0.01" name="to_b"/>
                <div class="calculator-field-step">step:</div>
                <input class="calculator-field-input-step" type="number"
value="<%=step_b%>" step="0.01" name="step_b"/>
            </div>
            <div class="calculator-field">
                <div class="calculator-field-name">c</div>
                <div class="calculator-field-from">from:</div>
                <input class="calculator-field-input-from" type="number"
value="<%=from_c%>" step="0.01" name="from_c"/>
                <div class="calculator-field-to">to:</div>
                <input class="calculator-field-input-to" type="number" value="<%=to_c%>"

```

```

step="0.01" name="to_c"/>
    <div class="calculator-field-step">step:</div>
    <input class="calculator-field-input-step" type="number"
value="<%=step_c%>" step="0.01" name="step_c"/>
</div>
    <div class="calculator-field">
        <div class="calculator-field-name">d</div>
        <div class="calculator-field-from">from:</div>
        <input class="calculator-field-input-from" type="number"
value="<%=from_d%>" step="0.01" name="from_d"/>
        <div class="calculator-field-to">to:</div>
        <input class="calculator-field-input-to" type="number" value="<%=to_d%>"
step="0.01" name="to_d"/>
        <div class="calculator-field-step">step:</div>
        <input class="calculator-field-input-step" type="number"
value="<%=step_d%>" step="0.01" name="step_d"/>
    </div>
    <div class="calculator-field-submit">
        <button class="calculator-submit-button" type="submit">
            Calculate
        </button>
    </div>
</form>
<div class="results-container">
    <table>
        <thead>
            <tr>
                <th>a</th>
                <th>b</th>
                <th>c</th>
                <th>d</th>
                <th>res</th>
            </tr>
        </thead>
        <tbody>
            <%=
                if (request.getMethod().equalsIgnoreCase("post")) {
                    double from_A ;
                    double from_B ;
                    double from_C ;
                    double from_D ;

                    double to_A ;
                    double to_B ;
                    double to_C ;
                    double to_D ;

                    double step_A ;
                    double step_B ;
                    double step_C ;
                    double step_D ;

                    try {
                        from_A = Double.parseDouble(request.getParameter("from_a"));
                        from_B = Double.parseDouble(request.getParameter("from_b"));
                        from_C = Double.parseDouble(request.getParameter("from_c"));
                        from_D = Double.parseDouble(request.getParameter("from_d"));

                        to_A = Double.parseDouble(request.getParameter("to_a"));
                        to_B = Double.parseDouble(request.getParameter("to_b"));
                        to_C = Double.parseDouble(request.getParameter("to_c"));
                        to_D = Double.parseDouble(request.getParameter("to_d"));

                        step_A = Double.parseDouble(request.getParameter("step_a"));
                        step_B = Double.parseDouble(request.getParameter("step_b"));

```

```

        step_C = Double.parseDouble(request.getParameter("step_c"));
        step_D = Double.parseDouble(request.getParameter("step_d"));

    } catch (NumberFormatException e) {
        response.sendError(400, "Дані введені не коректно");
        return;
    }

    for (double a = from_A; a <= to_A; a += step_A) {
        for (double b = from_B; b <= to_B; b += step_B) {
            for (double c = from_C; c <= to_C; c += step_C) {
                for (double d = from_D; d <= to_D; d += step_D) {
                    double res = calculateResult(a, b, c, d);

%>

                <tr>
                    <td><%= a %></td>
                    <td><%= b %></td>
                    <td><%= c %></td>
                    <td><%= d %></td>
                    <td><%= res %></td>
                </tr>

            <%
                }
            }
        }
    }

%>

</tbody>
</table>
</div>
</div>
</body>
</html>

```

Код 1.3 – form3.jsp

```

package com.example.lab_03;

import jakarta.servlet.ServletException;
import jakarta.servlet.http.HttpServlet;
import jakarta.servlet.http.HttpServletRequest;
import jakarta.servlet.http.HttpServletResponse;

import java.io.IOException;

public abstract class AbstractVladServlet extends HttpServlet {
    protected String _path = "";
    protected String _formFile = "";

    public void init() {

    }

    //Calculate
    public abstract double calculate(double a, double b, double c, double d);

```

```

//Forward request to jsp file
public void forwardToJspFile(HttpServletRequest request, HttpServletResponse
response) throws IOException, ServletException {
    request.getRequestDispatcher(_formFile).forward(request, response);
}

//GET handler
public void doGet(
    HttpServletRequest request,
    HttpServletResponse response
) throws IOException, ServletException
{
    forwardToJspFile(request, response);
}

//POST handler
public void doPost(
    HttpServletRequest request,
    HttpServletResponse response
) throws IOException, ServletException
{
    forwardToJspFile(request, response);
}

public void destroy() {
}
}

```

Код 1.4 – AbstractVladServlet

```

package com.example.lab_03;

import jakarta.servlet.annotation.WebServlet;

@WebServlet(name = "vladServlet1", value = "/form1")
public class VladServlet1 extends AbstractVladServlet {

    public VladServlet1(){
        _formFile = "form1.jsp";
        _path = "/lab_03_war_exploded/vlad-form1";
    }

    @Override
    public double calculate(double a, double b, double c, double d) {
        return Math.sqrt(Math.abs(Math.sin(a) - 4.0 * Math.log(b) / Math.pow(c, d)));
    }
}

```

Код 1.5 – VladServlet1

```

package com.example.lab_03;

import jakarta.servlet.annotation.WebServlet;

@WebServlet(name = "vladServlet2", value = "/form2")
public class VladServlet2 extends AbstractVladServlet {
    public VladServlet2(){
        _formFile = "form2.jsp";
        _path = "/lab_03_war_exploded/vlad-form2";
    }
}

```

```

@Override
public double calculate(double a, double b, double c, double d) {
    return (Math.exp(a) + 3 * Math.log10(c) * Math.abs(Math.atan(d))) /
(Math.pow(b, 1 / c));
}
}

```

Код 1.6 – VladServlet2

```

package com.example.lab_03;
import jakarta.servlet.annotation.WebServlet;

@WebServlet(name = "vladServlet3", value = "/form3")
public class VladServlet3 extends AbstractVladServlet {
    public VladServlet3() {
        _formFile = "form3.jsp";
        _path = "/lab_03_war_exploded/vlad-form3";
    }

    @Override
    public double calculate(double a, double b, double c, double d) {
        return Math.pow(2.0 * Math.sin(a) + Math.cos(Math.abs(b * Math.sqrt(c))), d);
    }
}

```

Код 1.7 – VladServlet3

```

<!DOCTYPE html>
<html lang="en">
<head>
    <meta charset="UTF-8">
    <title>Lab 3</title>
</head>
<body>
<p><a href="./form1">Formula 1</a></p>
<p><a href="./form2">Formula 2</a></p>
<p><a href="./form3">Formula 3</a></p>
</body>
</html>

```

Код 1.8 – Реалізація головної сторінки

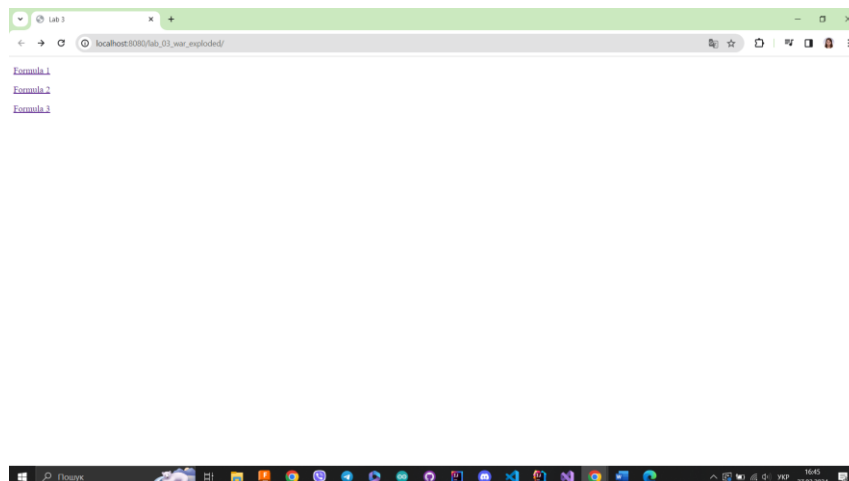


Рисунок 1.1 – Головна сторінка сайту

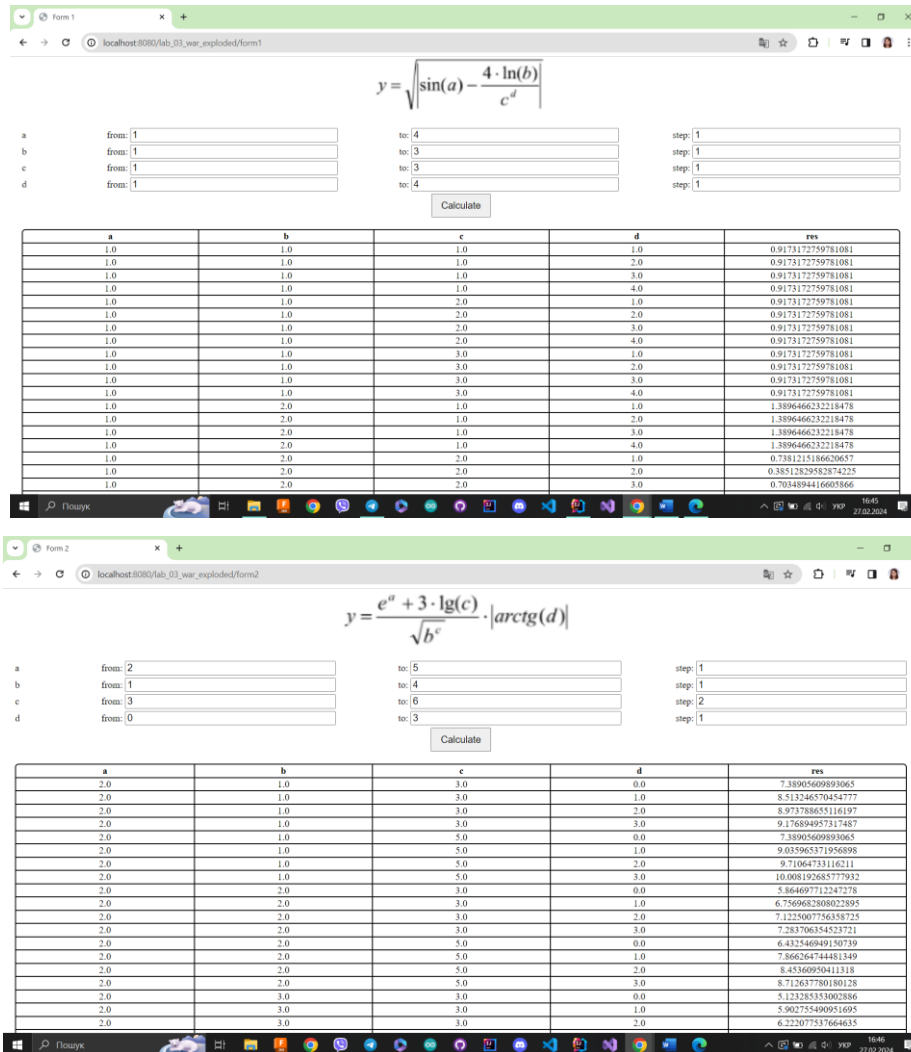


Рисунок 1.2(3) – Приклади інших сторінок сайту

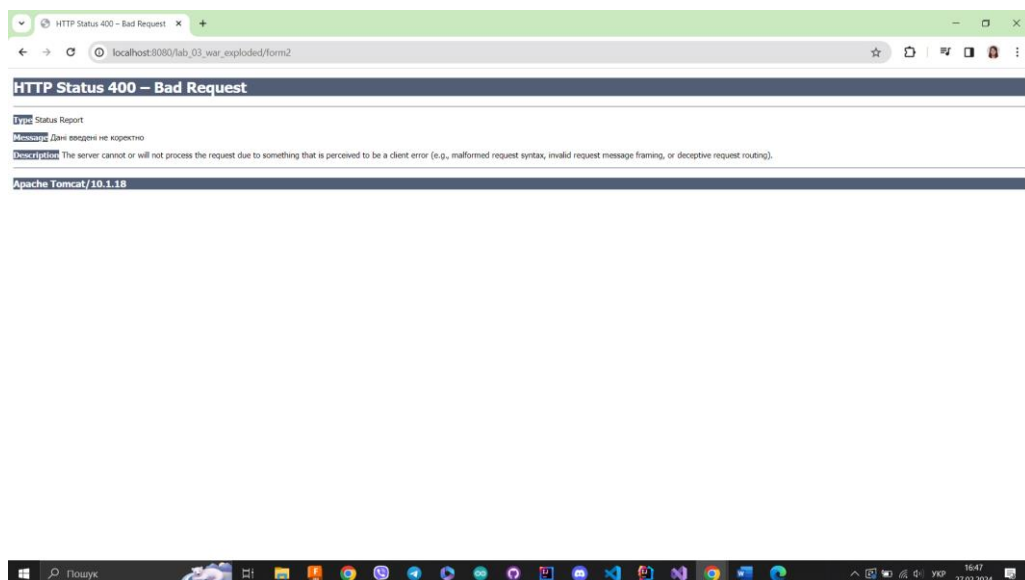


Рисунок 1.4 – Повернення помилки на некоректно введені дані

3. Відповісти на контрольні питання.

Висновки: У процесі виконання лабораторної роботи ми успішно ознайомились з технологією JavaServer Pages (JSP) та її можливостями для створення динамічних веб-сторінок. Розуміння принципів роботи JSP дозволить нам більш ефективно розробляти веб-застосунки у майбутньому.