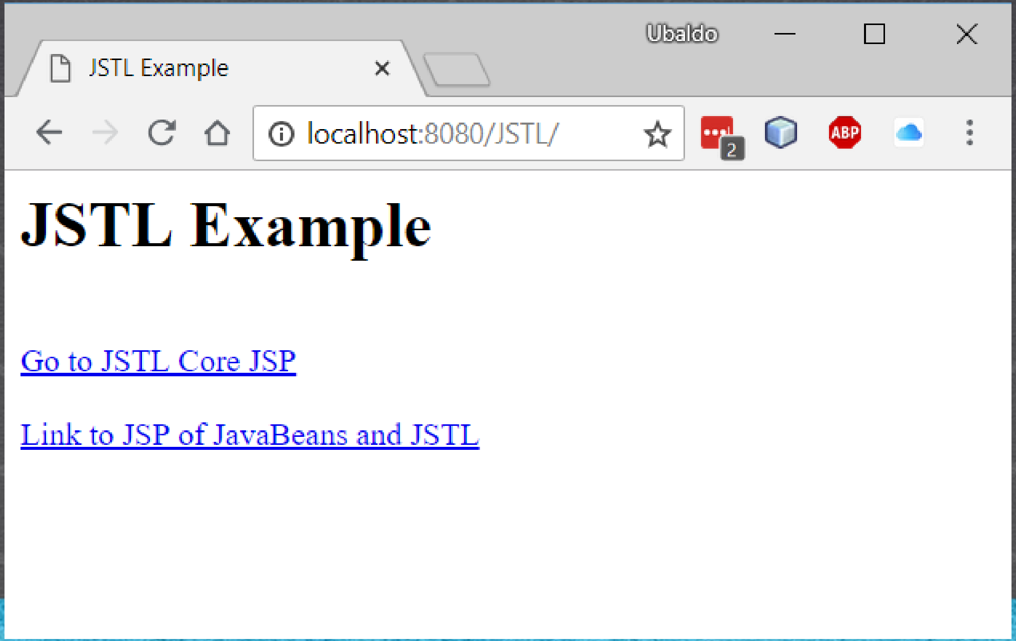
SERVLETS AND JSP COURSE

EXERCISE

JSTL WITH JSP’S

EXERCISE OBJECTIVE

Create an exercise to implement the use of JSTL with JSPs. At the end we should observe the following:



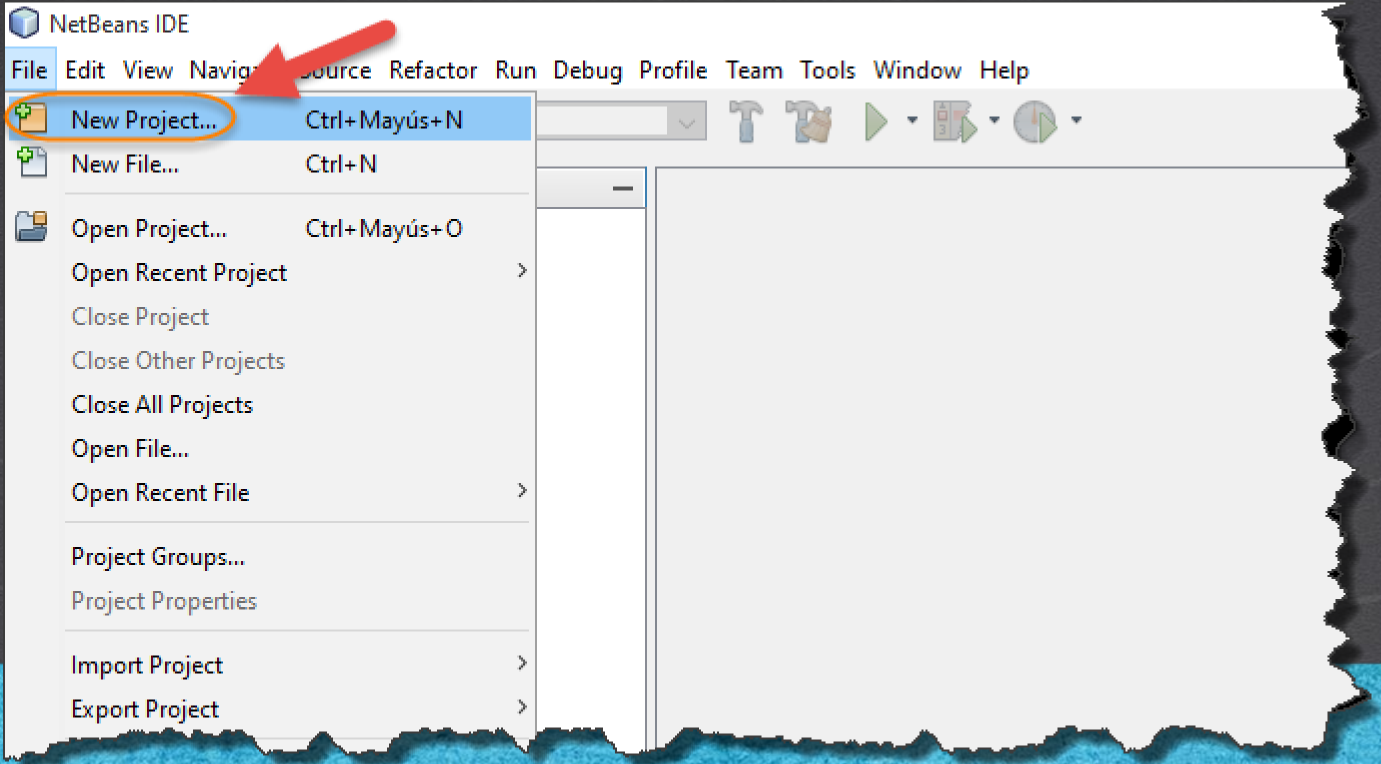
JSTL Example

Go to JSTL Core JSP

Link to JSP of JavaBeans and JSTL

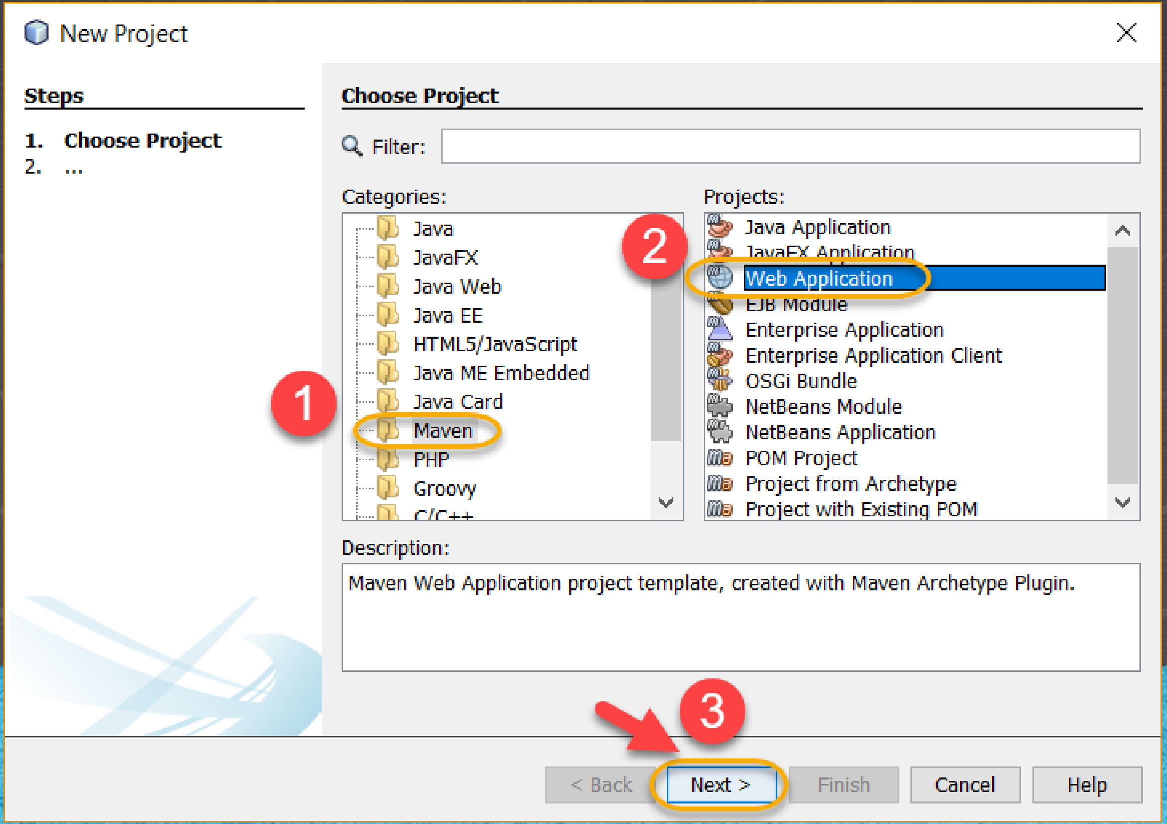
1. CREATE A NEW PROJECT

Create a new project:



1. CREATE A NEW PROJECT (CONT)

Create a new project:



New Project

Steps

1. **Choose Project**
2. ...

Choose Project

Filter:

Categories:

- Java
- JavaFX
- Java Web
- Java EE
- HTML5/JavaScript
- Java ME Embedded
- Java Card
- Maven
- PHP
- Groovy
- C/C++

Projects:

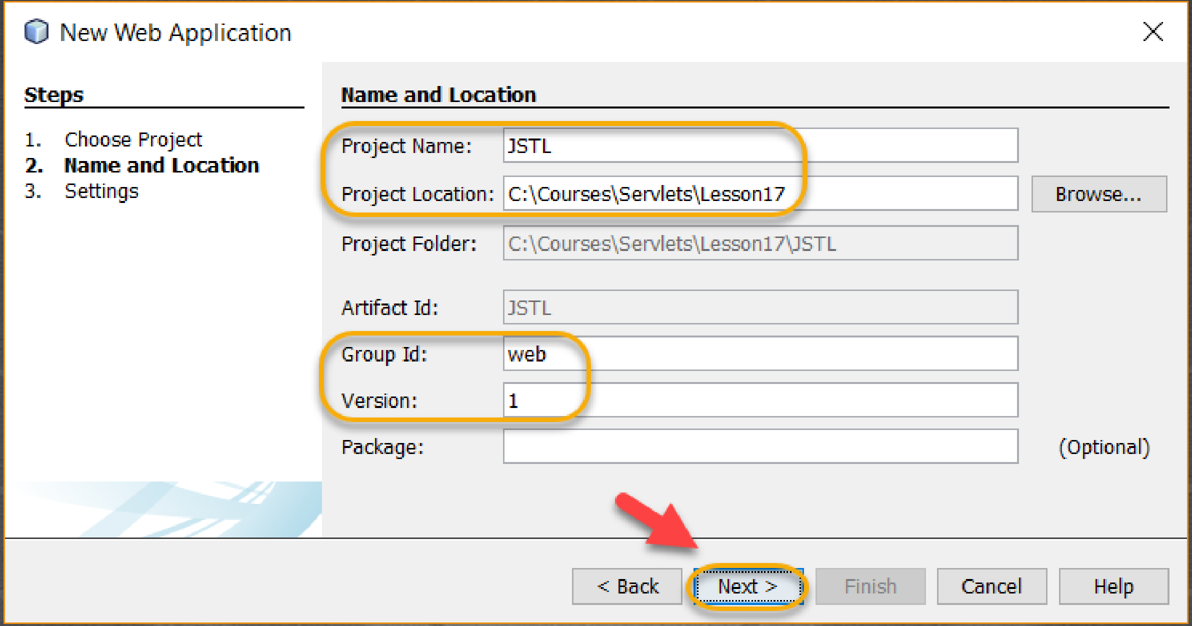
- Java Application
- JavaFX Application
- Web Application
- EJB Module
- Enterprise Application
- Enterprise Application Client
- OSGi Bundle
- NetBeans Module
- NetBeans Application
- POM Project
- Project from Archetype
- Project with Existing POM

Description:

Maven Web Application project template, created with Maven Archetype Plugin.

1. CREATE A NEW PROJECT (CONT)

Create a new project :



New Web Application

Steps

1. Choose Project
2. **Name and Location**
3. Settings

Name and Location

Project Name: JSTL

Project Location: C:\Courses\Servlets\Lesson17

Project Folder: C:\Courses\Servlets\Lesson17\JSTL

Artifact Id: JSTL

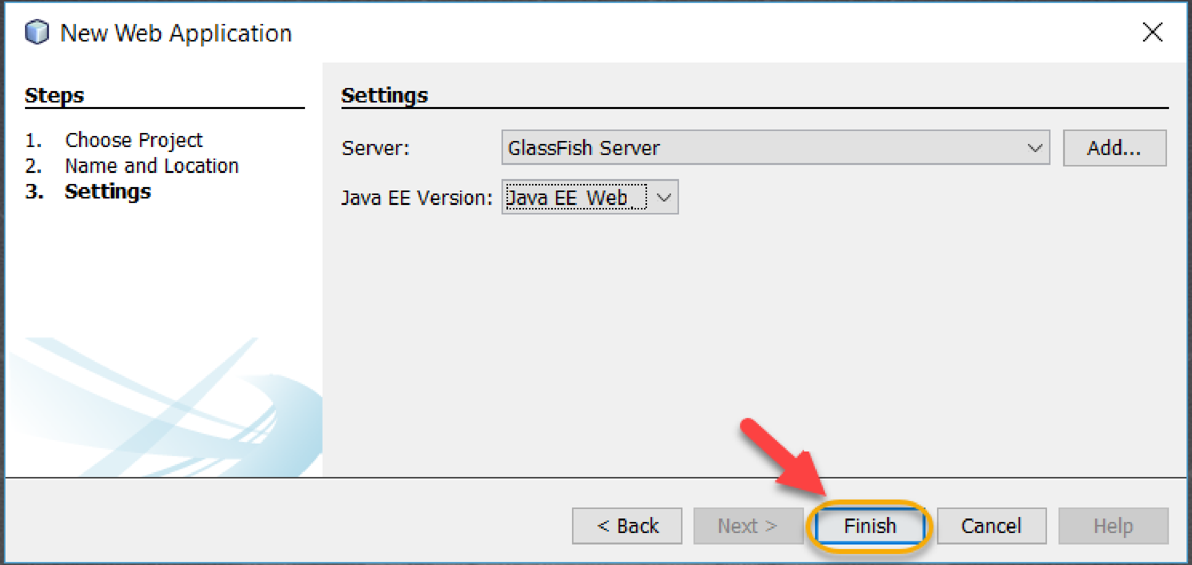
Group Id: web

Version: 1

Package: (Optional)

1. CREATE A NEW PROJECT (CONT)

Create a new project :



New Web Application

Steps

1. Choose Project
2. Name and Location
3. **Settings**

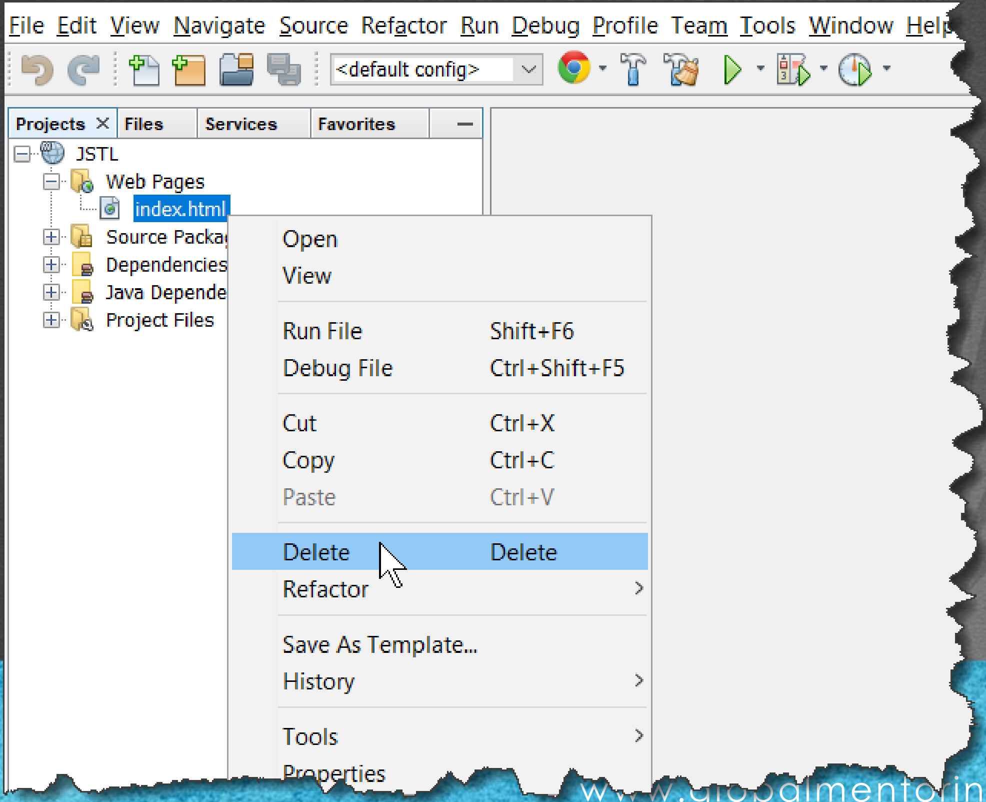
Settings

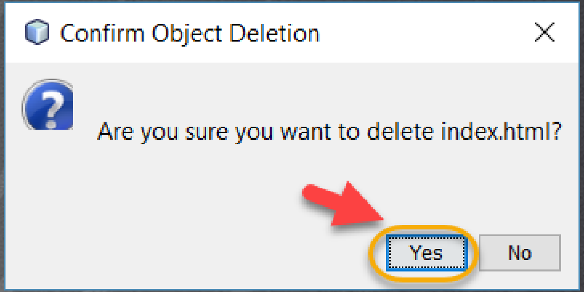
Server: GlassFish Server

Java EE Version: Java EE Web

2. DELETE A FILE

Delete the index.html file:

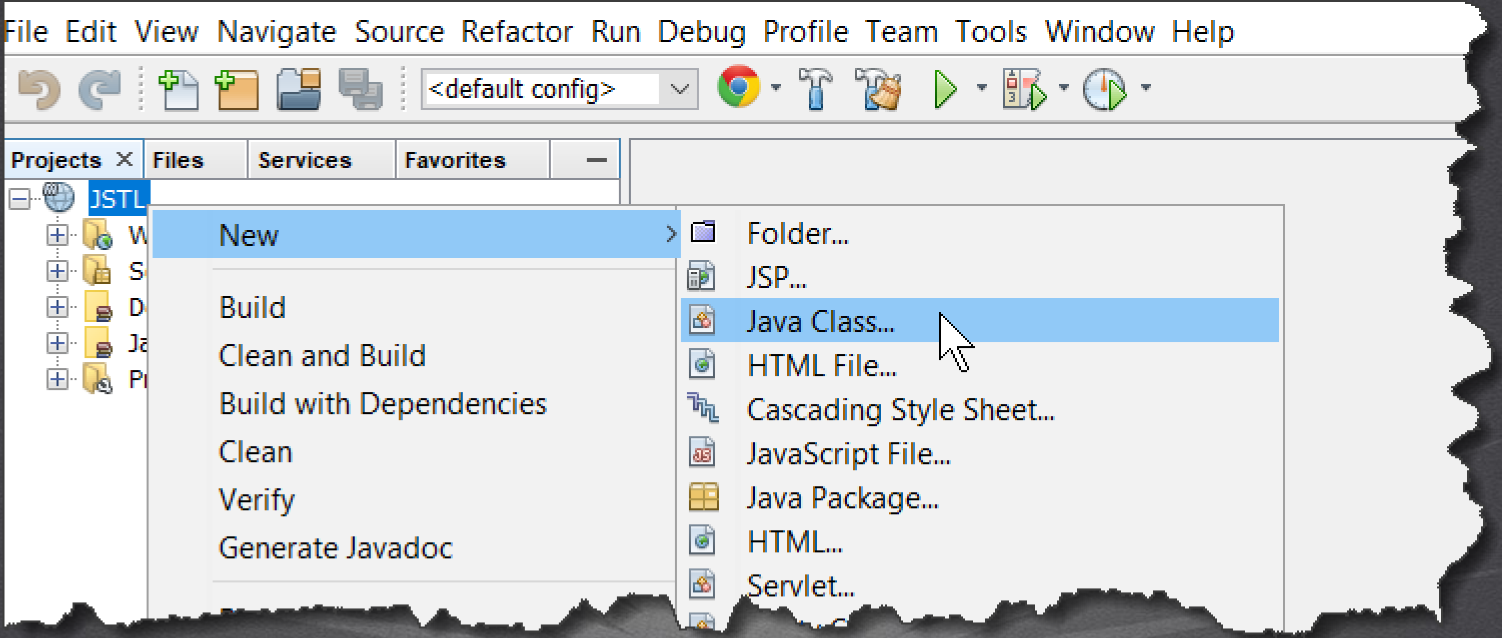




Are you sure you want to delete index.html?

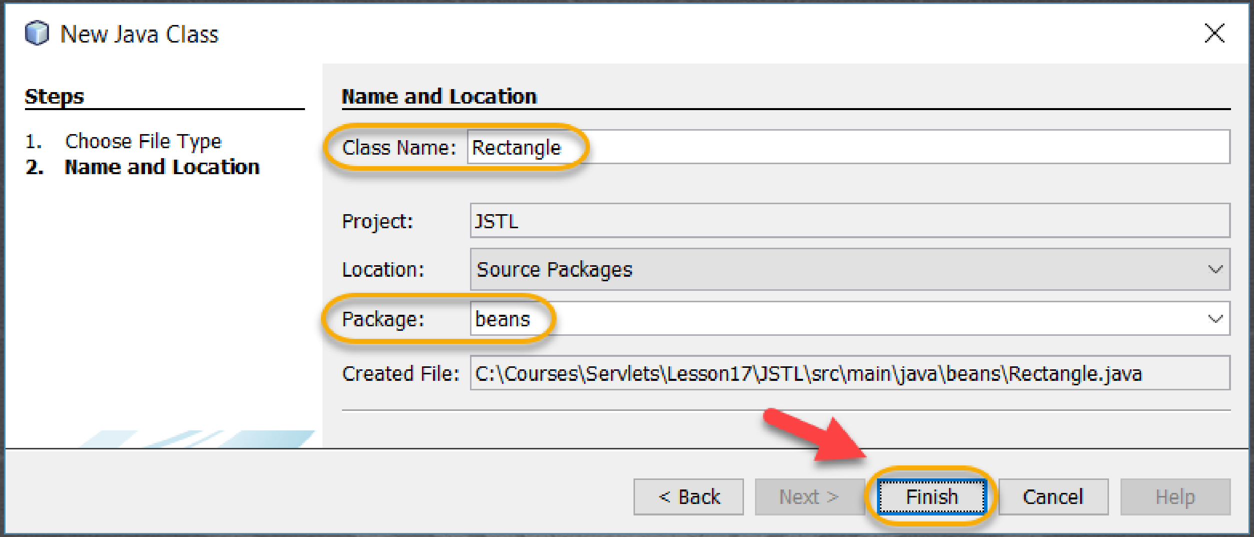
3. CREATE A CLASS

Create the Rectangle.java class:



3. CREATE A CLASS

Create the Rectangle.java class:



New Java Class

Steps

1. Choose File Type
2. **Name and Location**

Name and Location

Class Name: Rectangle

Project: JSTL

Location: Source Packages

Package: beans

Created File: C:\Courses\Servlets\Lesson17\JSTL\src\main\java\beans\Rectangle.java

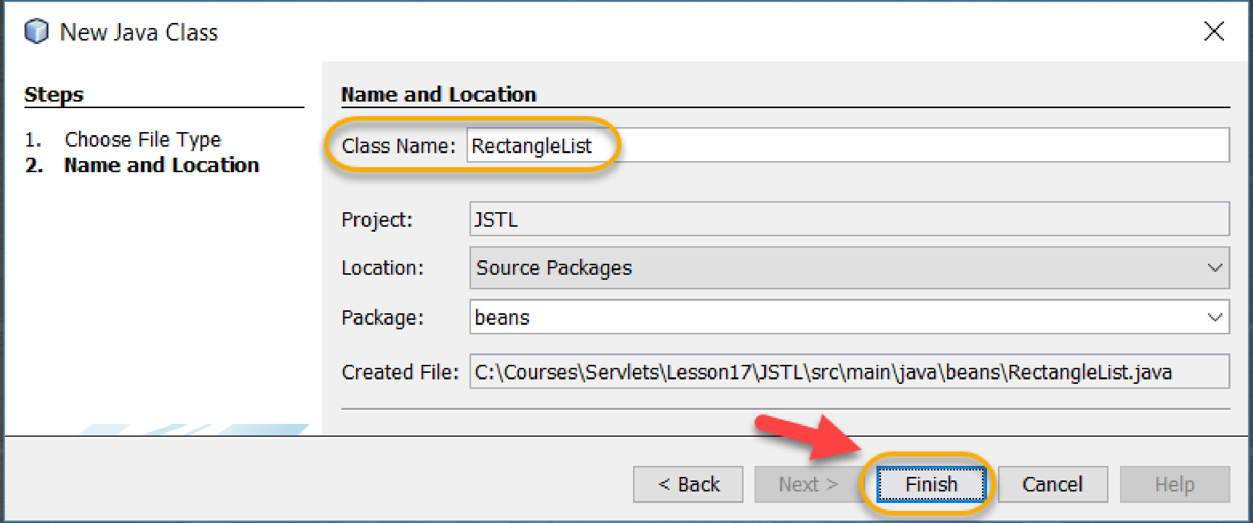
4. MODIFY THE FILE

Rectangle.java:

```java
package beans;

public class Rectangle {

  private int width;
  private int length;

  public int getLength() {
    return length;
  }

  public void setLength(int length) {
    this.length = length;
  }

  public int getWidth() {
    return width;
  }

  public void setWidth(int width) {
    this.width = width;
  }

  public int getArea(){
    return this.width * this.length;
  }
}
```

5. CREATE A NEW CLASS

Create the RectangleList.java class:



New Java Class

Steps

1. Choose File Type
2. **Name and Location**

Name and Location

Class Name: RectangleList

Project: JSTL

Location: Source Packages

Package: beans

Created File: C:\Courses\Servlets\Lesson17\JSTL\src\main\java\beans\RectangleList.java

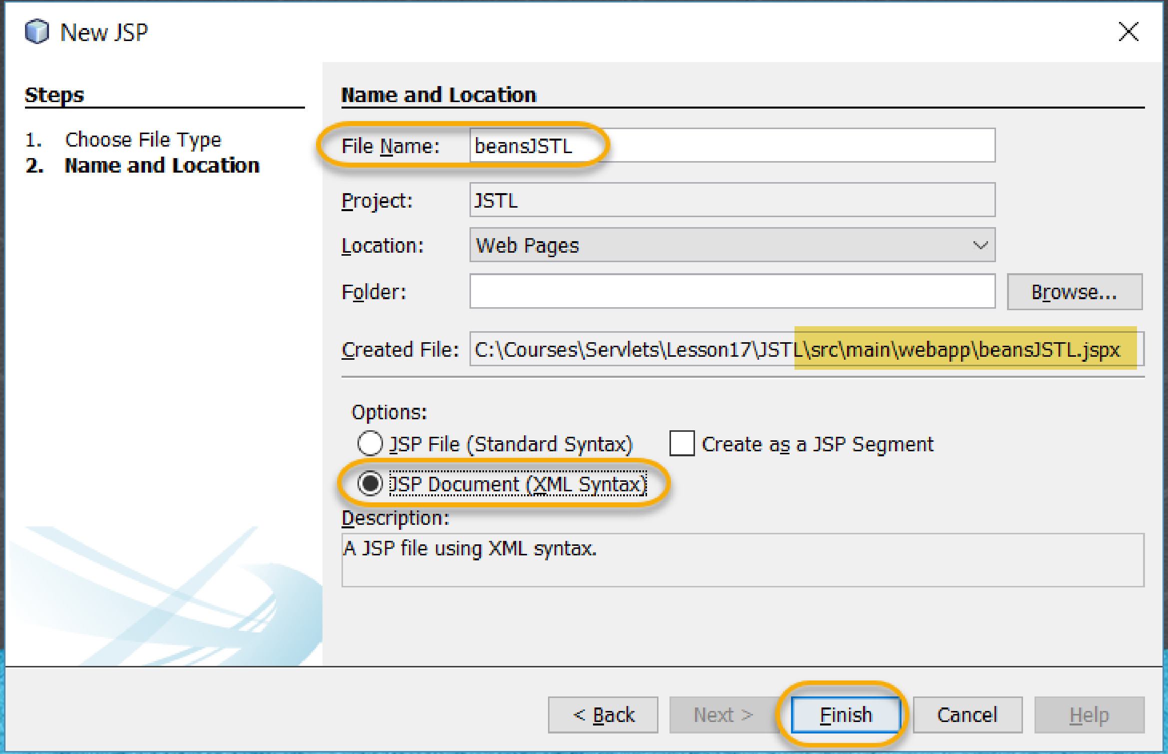
6. MODIFY THE FILE

RectangleList.java:

```
package beans;
import java.util.ArrayList;
import java.util.List;
/**
 * Class to generate a JavaBean from the list of rectangles
 *  and that can be used with jsp:useBean from a JSP
 * @author Ubaldo
 */
public class RectangleList {
  //Attribute that will store the rectangle objects
  private List<Rectangle> list = new ArrayList<>();

  /**
   * This method allows us to simulate a property called
   * "rectangle" from the action jsp:setProperty in a JSP
   * @param rec Rectangle
   */
  public void setRectangle(Rectangle rec){
    this.list.add( rec );
  }

  public List<Rectangle> getList() {
    return list;
  }

  public void setList(List<Rectangle> list) {
    this.list = list;
  }
}
```

7. CREATE A NEW JSPX

Create the beansJSTL.jspx file:



New JSP

Steps

1. Choose File Type
2. **Name and Location**

Name and Location

File Name: beansJSTL

Project: JSTL

Location: Web Pages

Folder:

Created File: C:\Courses\Servlets\Lesson17\JSTL\src\main\webapp\beansJSTL.jspx

Options:

○ JSP File (Standard Syntax) □ Create as a JSP Segment

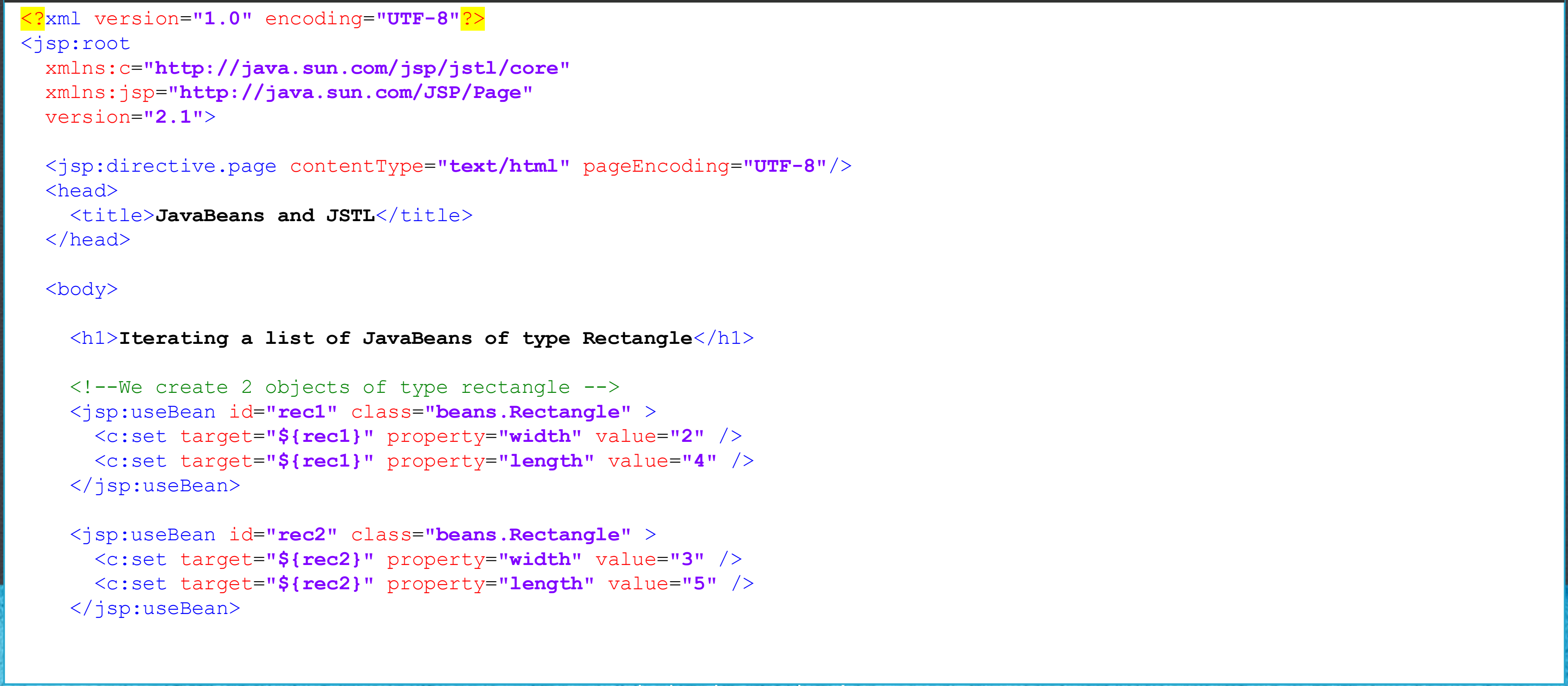
● JSP Document (XML Syntax)

Description:

A JSP file using XML syntax.

8. MODIFY THE FILE

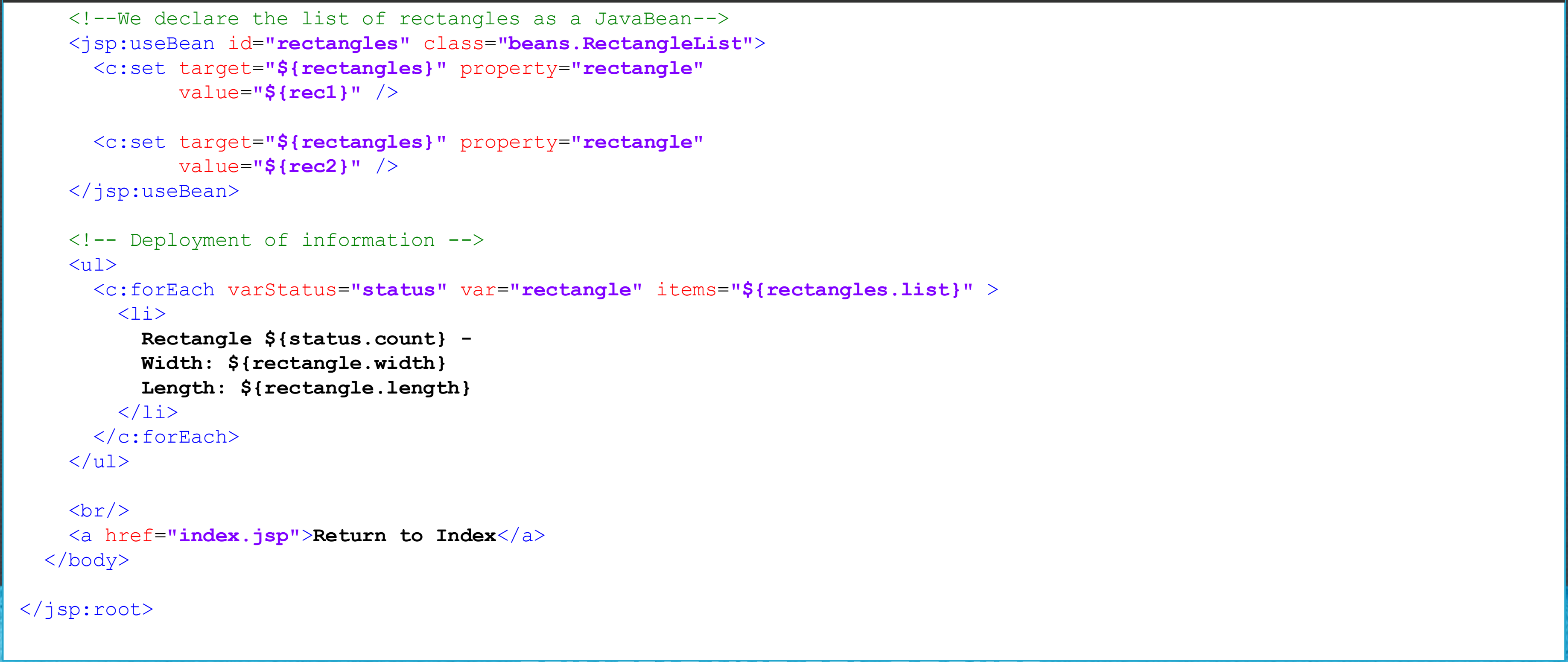
beansJSTL.jspx:



```
<?xml version="1.0" encoding="UTF-8"?>
<jsp:root
 xmlns:c="http://java.sun.com/jsp/jstl/core"
 xmlns:jsp="http://java.sun.com/JSP/Page"
 version="2.1">

 <jsp:directive.page contentType="text/html" pageEncoding="UTF-8"/>
 <head>
  <title>JavaBeans and JSTL</title>
 </head>

 <body>

  <h1>Iterating a list of JavaBeans of type Rectangle</h1>

  <!--We create 2 objects of type rectangle -->
  <jsp:useBean id="rec1" class="beans.Rectangle" >
   <c:set target="${rec1}" property="width" value="2" />
   <c:set target="${rec1}" property="length" value="4" />
  </jsp:useBean>

  <jsp:useBean id="rec2" class="beans.Rectangle" >
   <c:set target="${rec2}" property="width" value="3" />
   <c:set target="${rec2}" property="length" value="5" />
  </jsp:useBean>
```

8. MODIFY THE FILE

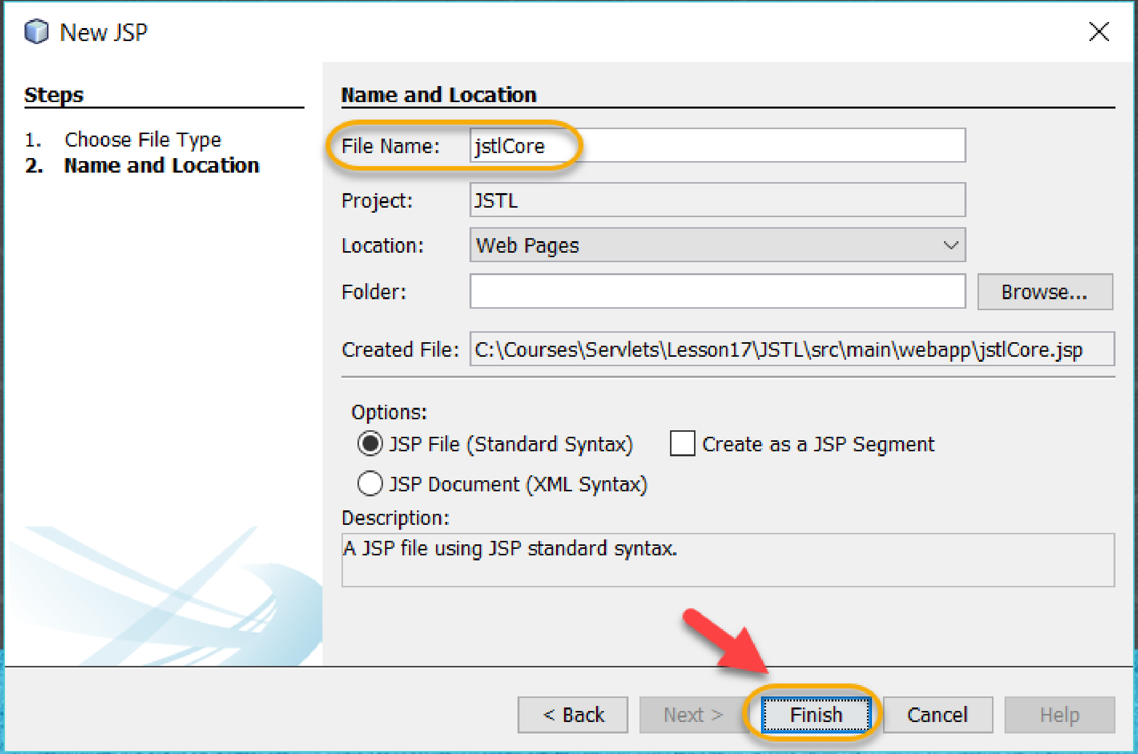
beansJSTL.jspx:



```
    <!--We declare the list of rectangles as a JavaBean-->
    <jsp:useBean id="rectangles" class="beans.RectangleList">
      <c:set target="${rectangles}" property="rectangle"
             value="${rec1}" />

      <c:set target="${rectangles}" property="rectangle"
             value="${rec2}" />
    </jsp:useBean>

    <!-- Deployment of information -->
    <ul>
      <c:forEach varStatus="status" var="rectangle" items="${rectangles.list}" >
        <li>
          Rectangle ${status.count} -
          Width: ${rectangle.width}
          Length: ${rectangle.length}
        </li>
      </c:forEach>
    </ul>

    <br/>
    <a href="index.jsp">Return to Index</a>
  </body>

</jsp:root>
```

9. CREATE A NEW JSP

Create the jstlCore.jsp file:



New JSP

Steps

1. Choose File Type
2. **Name and Location**

Name and Location

File Name: jstlCore

Project: JSTL

Location: Web Pages

Folder:

Created File: C:\Courses\Servlets\Lesson17\JSTL\src\main\webapp\jstlCore.jsp

Options:

● JSP File (Standard Syntax) □ Create as a JSP Segment

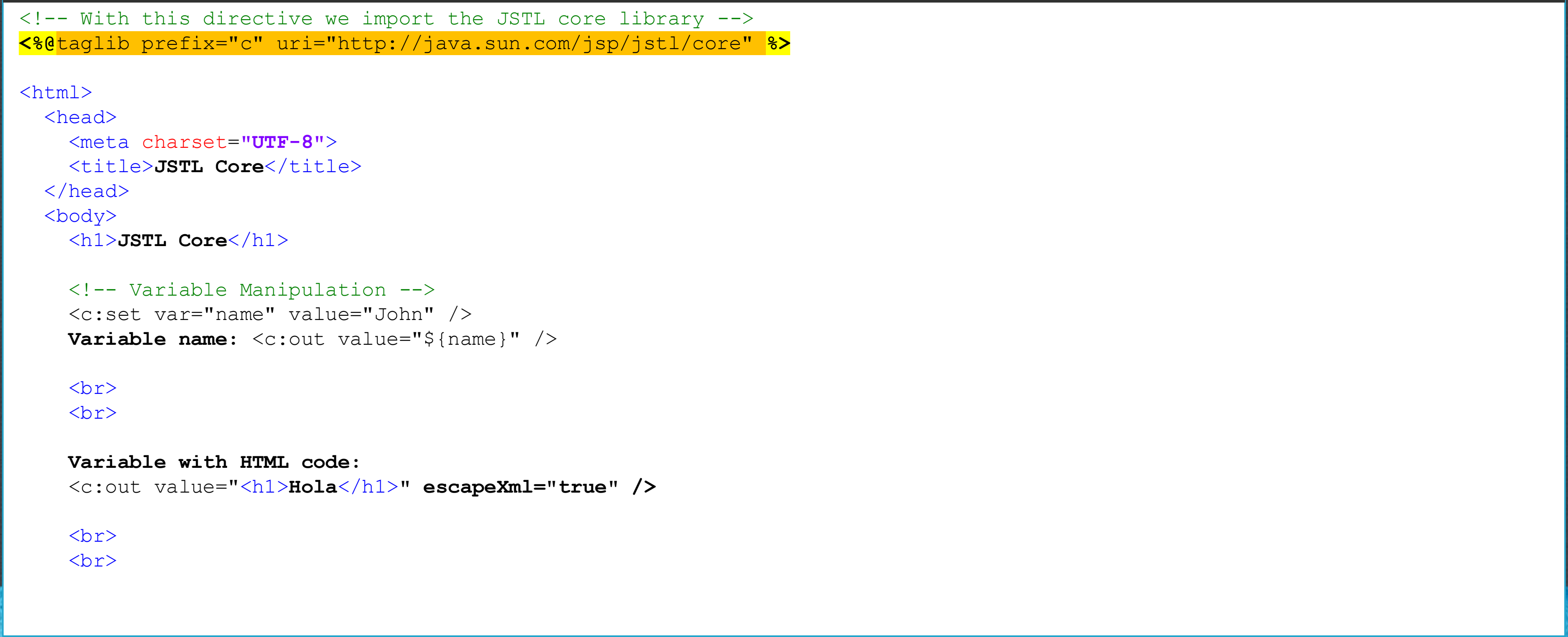
○ JSP Document (XML Syntax)

Description:

A JSP file using JSP standard syntax.

10. MODIFY THE CODE

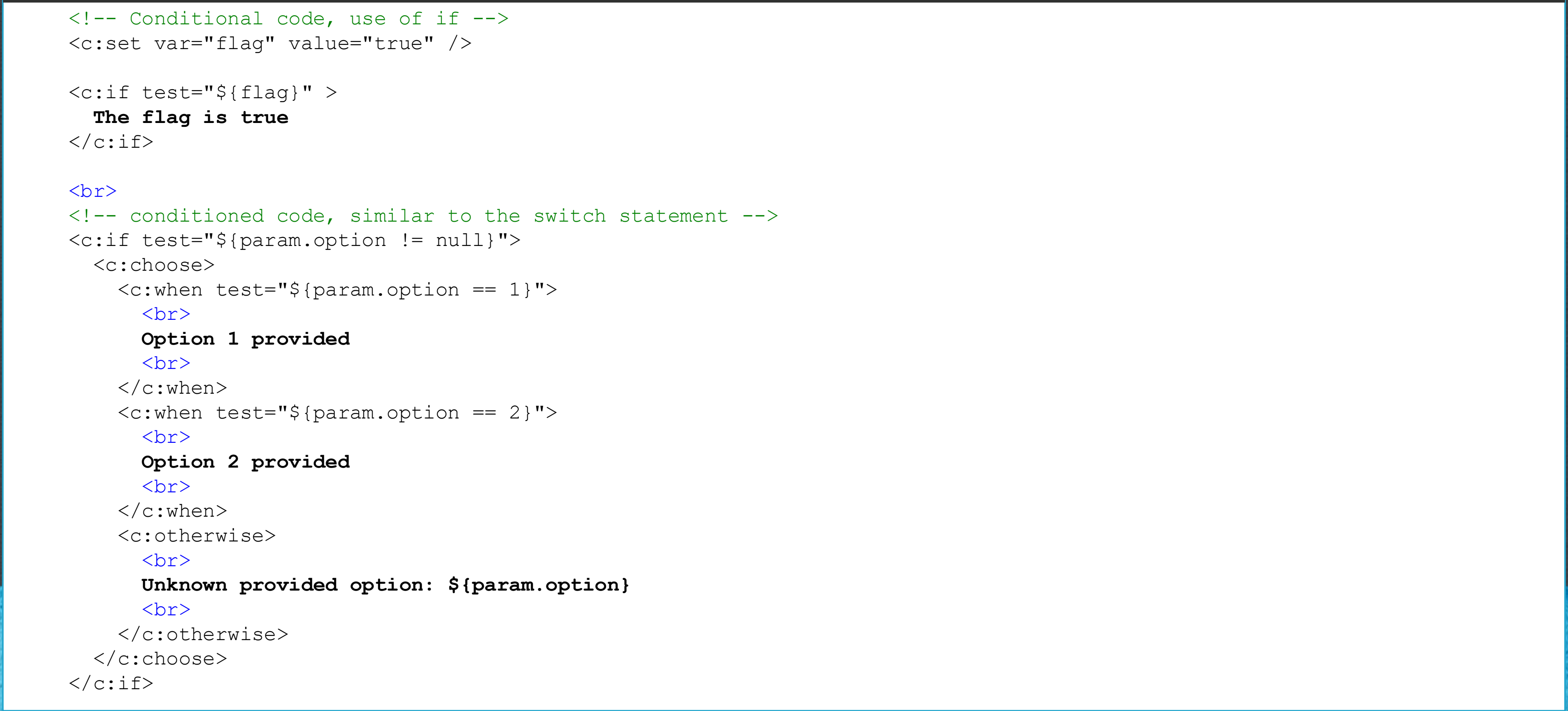
jstlCore.jsp:



```
<!-- With this directive we import the JSTL core library -->
<%@taglib prefix="c" uri="http://java.sun.com/jsp/jstl/core" %>

<html>
  <head>
    <meta charset="UTF-8">
    <title>JSTL Core</title>
  </head>
  <body>
    <h1>JSTL Core</h1>

    <!-- Variable Manipulation -->
    <c:set var="name" value="John" />
    Variable name: <c:out value="${name}" />

    <br>
    <br>

    Variable with HTML code:
    <c:out value="<h1>Hola</h1>" escapeXml="true" />

    <br>
    <br>
```

10. MODIFY THE CODE

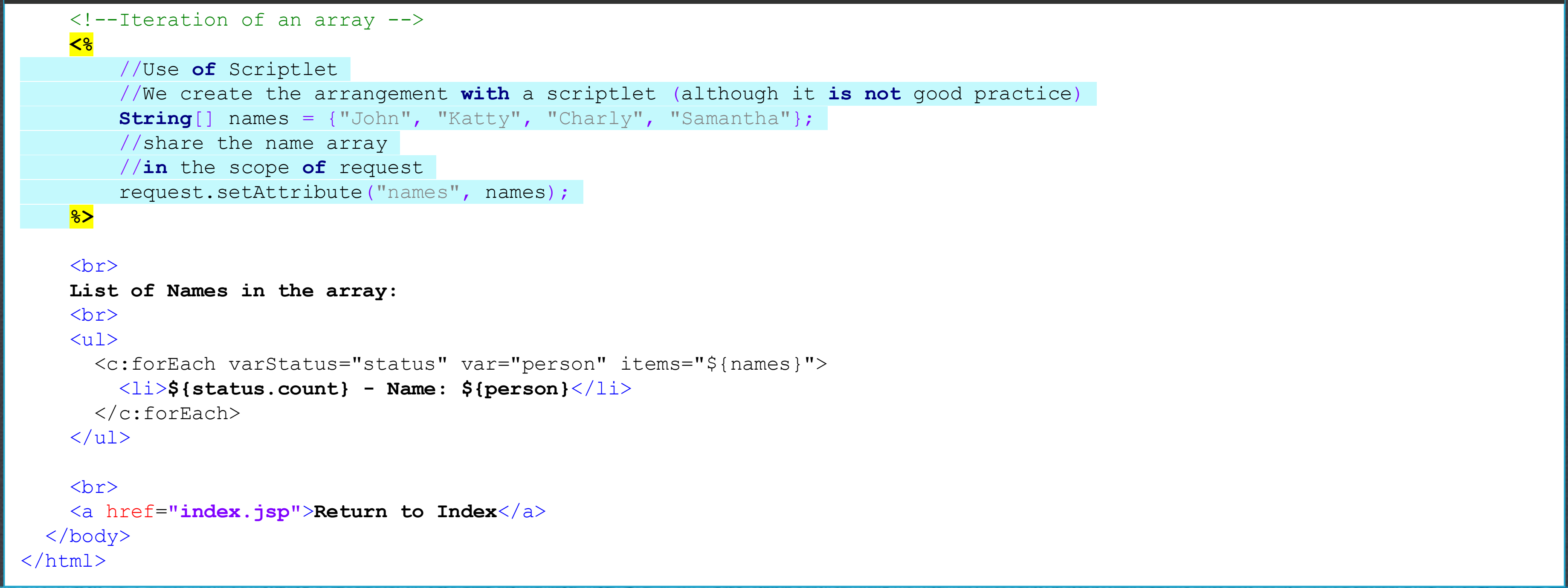
jstlCore.jsp:



```
<!-- Conditional code, use of if -->
<c:set var="flag" value="true" />

<c:if test="${flag}" >
  The flag is true
</c:if>

<br>
<!-- conditioned code, similar to the switch statement -->
<c:if test="${param.option != null}">
  <c:choose>
    <c:when test="${param.option == 1}">
      <br>
      Option 1 provided
      <br>
    </c:when>
    <c:when test="${param.option == 2}">
      <br>
      Option 2 provided
      <br>
    </c:when>
    <c:otherwise>
      <br>
      Unknown provided option: ${param.option}
      <br>
    </c:otherwise>
  </c:choose>
</c:if>
```

10. MODIFY THE CODE

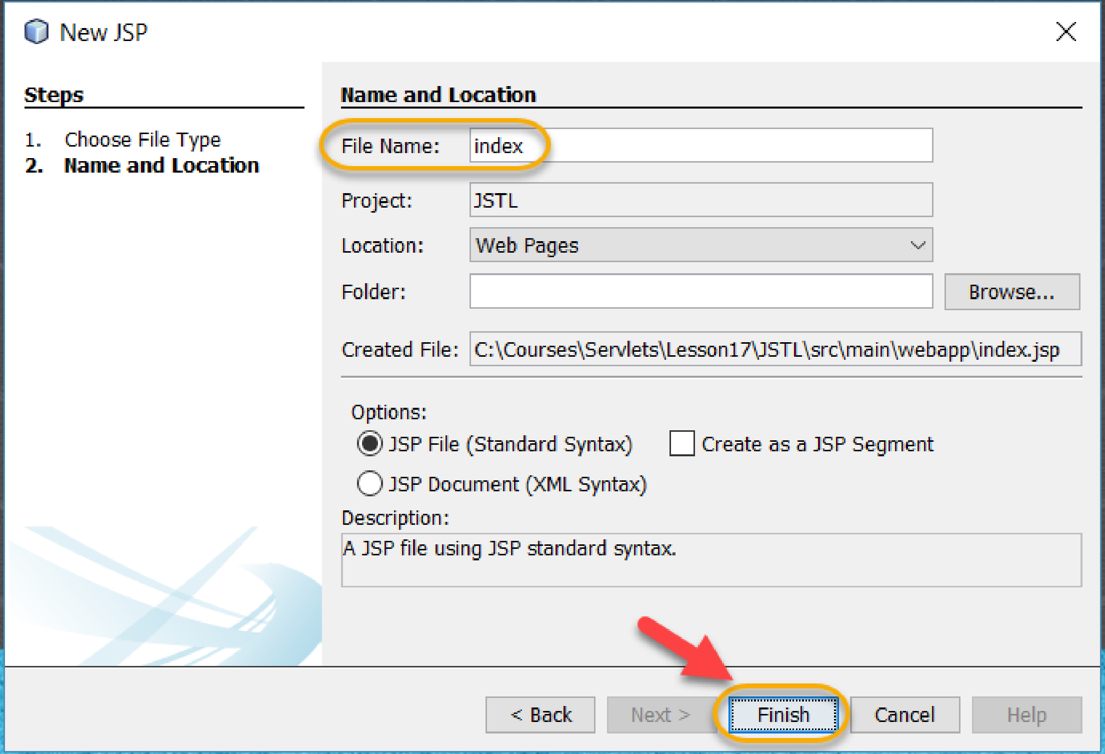
jstlCore.jsp:



```
  <!--Iteration of an array -->
  <%
    //Use of Scriptlet
    //We create the arrangement with a scriptlet (although it is not good practice)
    String[] names = {"John", "Katty", "Charly", "Samantha"};
    //share the name array
    //in the scope of request
    request.setAttribute("names", names);
  %>

  <br>
  List of Names in the array:
  <br>
  <ul>
   <c:forEach varStatus="status" var="person" items="${names}">
    <li>${status.count} - Name: ${person}</li>
   </c:forEach>
  </ul>

  <br>
  <a href="index.jsp">Return to Index</a>
 </body>
</html>
```

11. CREATE A NEW JSP

Create the index.jsp file:



New JSP

Steps

1. Choose File Type
2. **Name and Location**

Name and Location

File Name: index

Project: JSTL

Location: Web Pages

Folder:

Created File: C:\Courses\Servlets\Lesson17\JSTL\src\main\webapp\index.jsp

Options:

● JSP File (Standard Syntax) □ Create as a JSP Segment

○ JSP Document (XML Syntax)

Description:

A JSP file using JSP standard syntax.

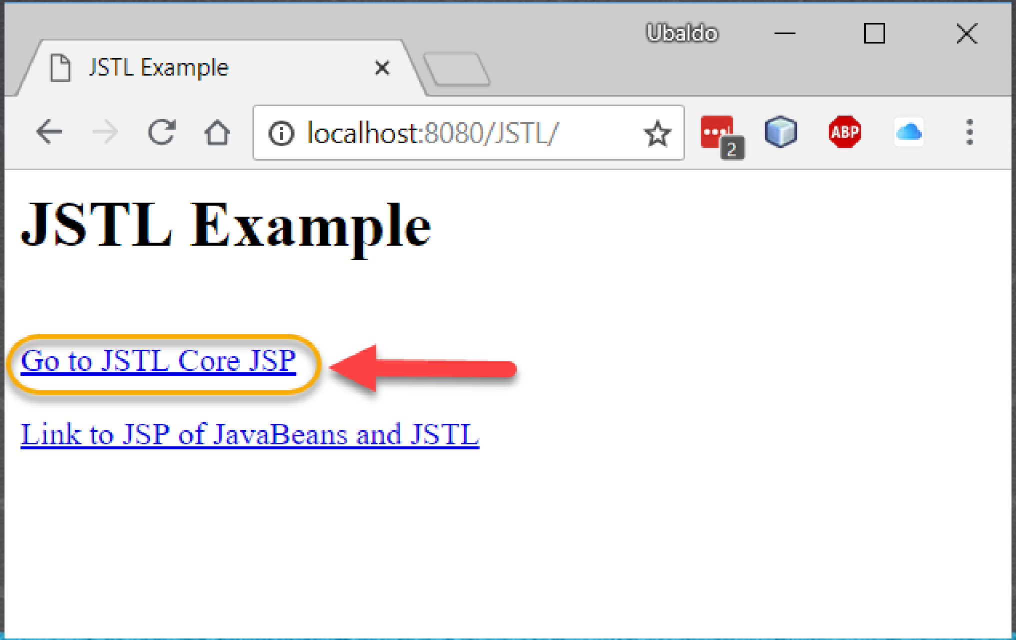
12. MODIFY THE FILE

index.jsp:

```
<html>
    <head>
         <meta charset="UTF-8">
        <title>JSTL Example</title>
    </head>
    <body>
        <h1>JSTL Example</h1>
        <br>
        <!-- Access to JSTL Core -->
        <a href="jstlCore.jsp?option=1">Go to JSTL Core JSP</a>
        <br>
        <br>
        <!-- Access to JavaBeans-->
        <a href="beansJSTL.jspx">Link to JSP of JavaBeans and JSTL</a>
    </body>
</html>
```

13. EXECUTE THE PROJECT

The result is as follows:



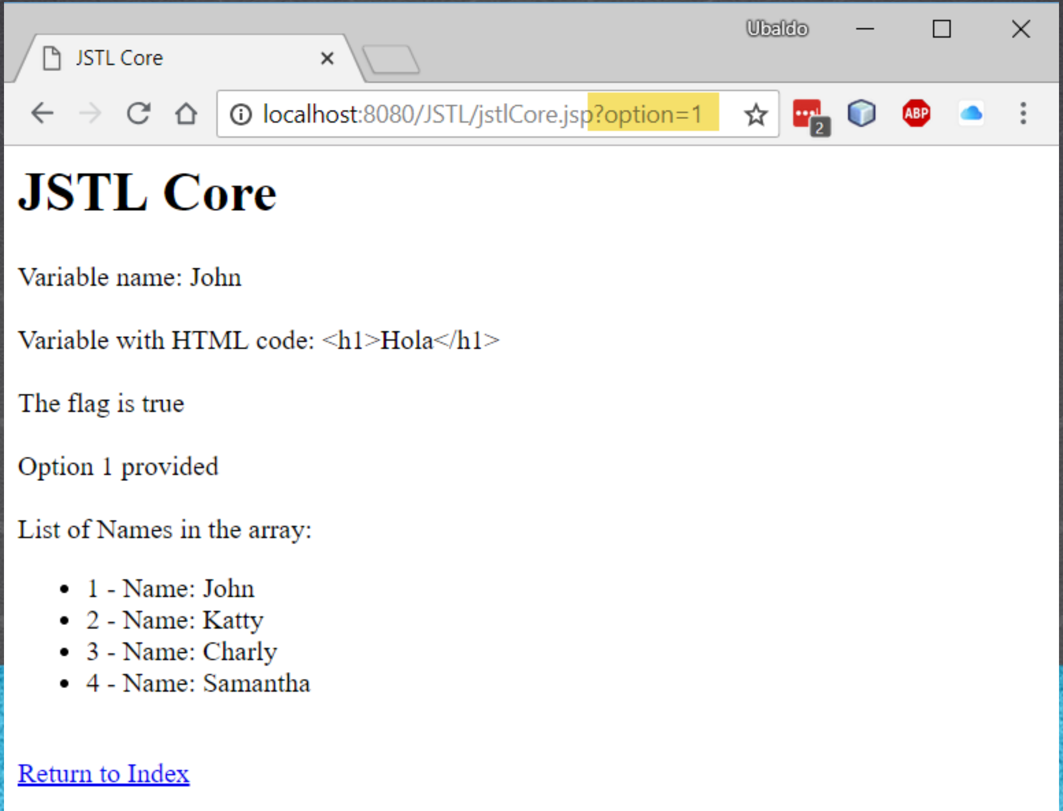
JSTL Example

Go to JSTL Core JSP

Link to JSP of JavaBeans and JSTL

13. EXECUTE THE PROJECT

The result is as follows:



JSTL Core

Variable name: John

Variable with HTML code: <h1>Hola</h1>

The flag is true

Option 1 provided

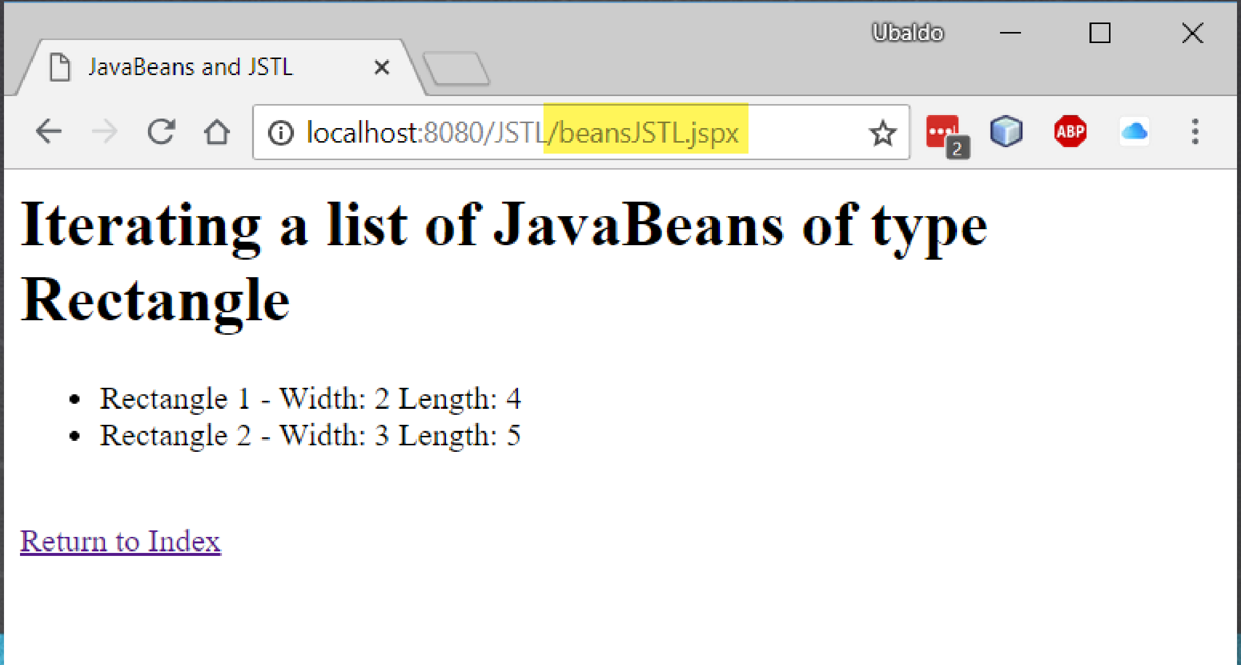
List of Names in the array:

- 1 - Name: John
- 2 - Name: Katty
- 3 - Name: Charly
- 4 - Name: Samantha

Return to Index

13. EXECUTE THE PROJECT

The result is as follows:



Iterating a list of JavaBeans of type Rectangle

- Rectangle 1 - Width: 2 Length: 4
- Rectangle 2 - Width: 3 Length: 5

Return to Index

EXERCISE CONCLUSION

- With this exercise we have put into practice the use of JSTL, both in jsp and jspx files.
- We observe how we must configure our JSPs to make use of this library.
- And finally we saw how combining JSTL with EL we have a cleaner code and easier to maintain in our JSPs.

ONLINE COURSE

SERVLETS & JSP' S

By: Eng. Ubaldo Acosta

Global Mentoring

SERVLETS AND JSP COURSE

www.globalmentoring.com.mx