

JAVA SERVLETS

Taula de Continguts

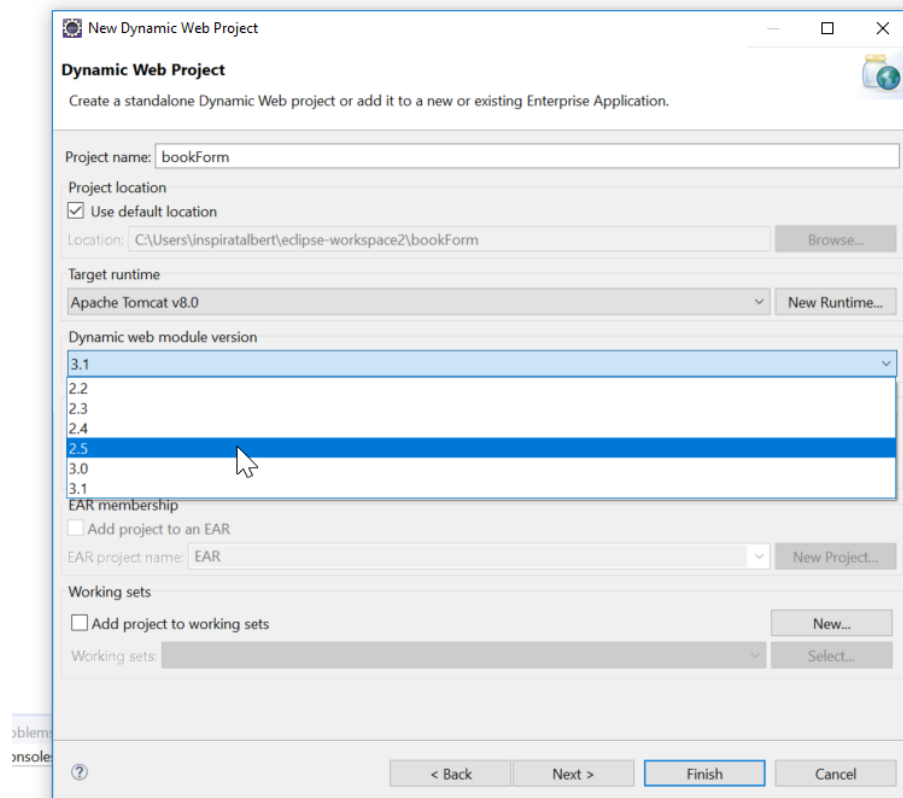
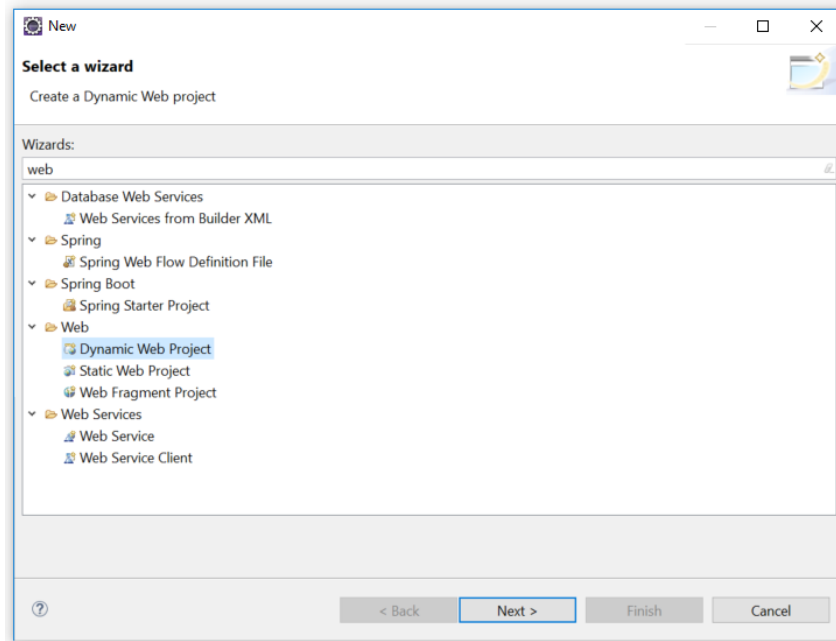
Taula de Continguts	2
Sevlets Exemple1 Book	3
SetUp Project Dynamic Web Project	3
create servlets 2.5	3
New Project first look	5
Create servlet	5
New Project with servlet	6
Configuration files	6
web.xml	6
Web files	7
index.html	7
Java files	9
servlet	9
Sevlets Exemple2 BookForm	11
Project	11
Web files	12
index.html	12
formBook.html	13
Java files	14
servlet	14
Configuration files	15
web.xml	15
Sevlets Exemple3 StudentForm (NO HAY WEB.XML)	17
SetUp Project Dynamic Web Project	17
create servlets 3.1	17
project	18
Java files	19
servlet	19
Web files	21
index.html	21

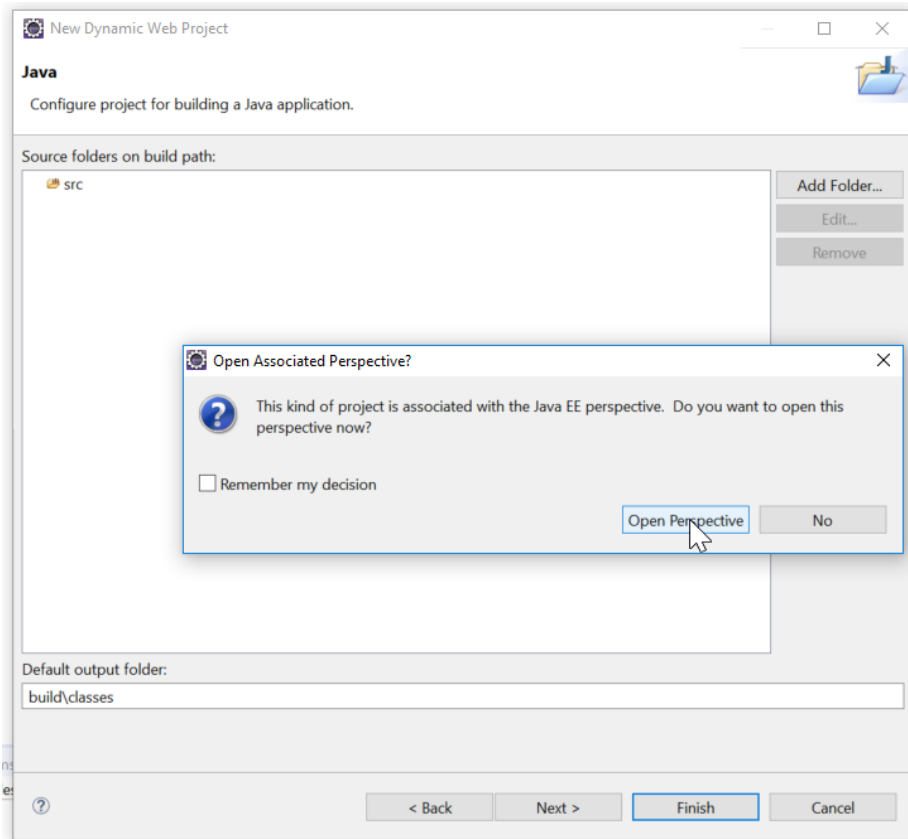
output	22
Sevlets Exemple4 Login (NO HAY WEB.XML)	23
SetUp Project Dynamic Web Project	23
project	23
Java files	24
servlet	24
Web files	26
index.html	26
Output	27

Servlets Exemple1 Book

SetUp Project Dynamic Web Project

create servlets 2.5

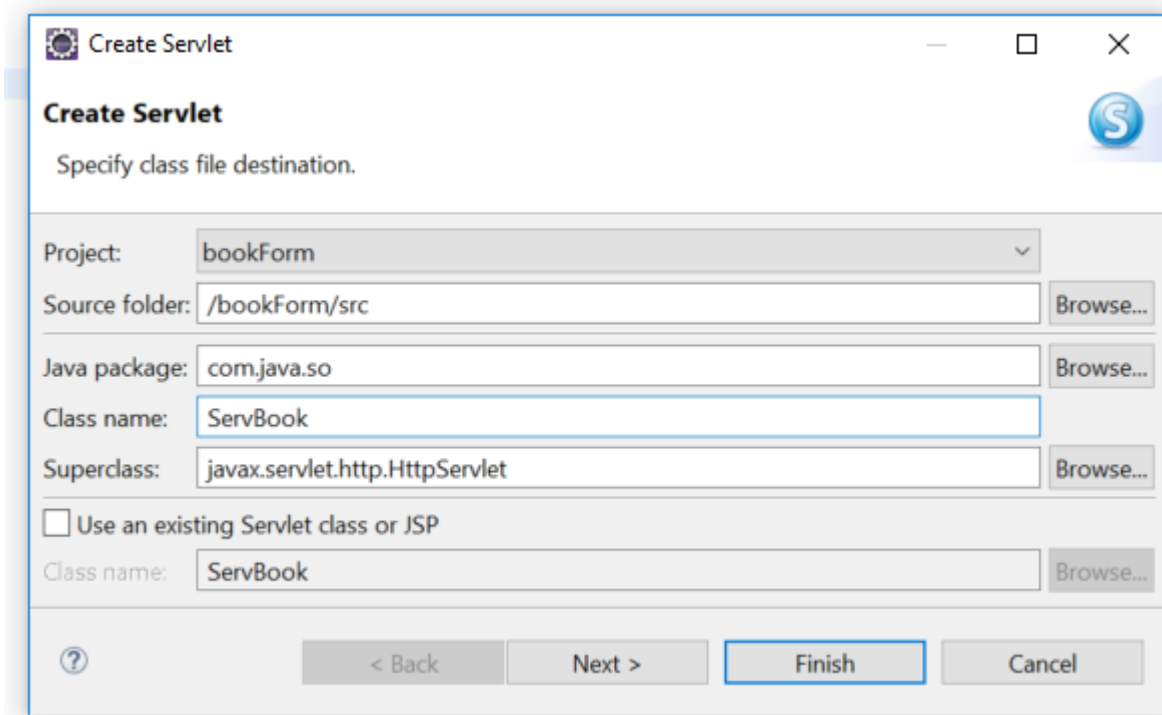




New Project first look

- ▼ bookForm
 - > Deployment Descriptor: bookForm
 - > JAX-WS Web Services
 - > Java Resources
 - > JavaScript Resources
 - build
 - ▼ WebContent
 - ▼ META-INF
 - MANIFEST.MF
 - ▼ WEB-INF
 - lib
 - web.xml

Create servlet



New Project with servlet

- bookForm
 - Deployment Descriptor: bookForm
 - JAX-WS Web Services
 - Java Resources
 - src
 - com.java.so
 - ServBook.java
 - Libraries
 - JavaScript Resources
 - build
 - WebContent
 - META-INF
 - MANIFEST.MF
 - WEB-INF
 - lib
 - web.xml
 - index.html

Configuration files

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"
  xsi:schemaLocation="http://java.sun.com/xml/ns/javaee
http://java.sun.com/xml/ns/javaee/web-app_2_5.xsd"
  id="WebApp_ID" version="2.5">
  <display-name>bookForm</display-name>
  <welcome-file-list>
    <welcome-file>index.html</welcome-file>
    <welcome-file>index.htm</welcome-file>
    <welcome-file>index.jsp</welcome-file>
    <welcome-file>default.html</welcome-file>
    <welcome-file>default.htm</welcome-file>
    <welcome-file>default.jsp</welcome-file>
  </welcome-file-list>

  <servlet>
    <description></description>
    <display-name>ServBook</display-name>
    <servlet-name>ServBook</servlet-name>
    <servlet-class>com.java.so.ServBook</servlet-class>
    <init-param>
      <param-name>inicio</param-name>
      <param-value>10</param-value>
    </init-param>
  </servlet>

  <servlet-mapping>
    <servlet-name>ServBook</servlet-name>
    <url-pattern>/ServBook</url-pattern>
  </servlet-mapping>

</web-app>
```

Web files

index.html

```
<!DOCTYPE html>
<html>
<head>
<meta charset="ISO-8859-1">
<title>Insert title here</title>
</head>

<body>
Hello Class, how are u? Loving persistence, entities and servlets?
Really?
</body>

</html>
```


Java files

servlet

```
package com.java.so;

import java.io.IOException;
import java.io.PrintWriter;

import javax.servlet.ServletConfig;
import javax.servlet.ServletException;
import javax.servlet.http.HttpServlet;
import javax.servlet.http.HttpServletRequest;
import javax.servlet.http.HttpServletResponse;

public class ServBook extends HttpServlet {
    private static final long serialVersionUID = 1L;

    // create static to share all petitions
    private int peticiones;

    public ServBook() {
        super();
    }

    @Override
    public void init(ServletConfig config) throws ServletException {

        // peticiones get this value
        peticiones =
Integer.parseInt(config.getInitParameter("inicio"));
    }

    protected void doGet(HttpServletRequest request,
HttpServletResponse response)
        throws ServletException, IOException {

        PrintWriter pw = response.getWriter();
        pw.println("We are printing from Servlet BookForm");
        pw.print("Request number: " + peticiones);
        peticiones++;
        pw.close();
    }
}
```

output

← → ↻ 🏠 ⓘ localhost:8080/bookForm/ServBook

We are printing from Servlet BookForm
Request number: 10

← → ↻ 🏠 ⓘ localhost:8080/bookForm/ServBook

We are printing from Servlet BookForm
Request number: 14

Sevlets Exemple2 BookForm

Project

- ▼ 📁 bookForm
 - > 📄 Deployment Descriptor: bookForm
 - > 🌐 JAX-WS Web Services
 - ▼ 📁 Java Resources
 - ▼ 📁 src
 - ▼ 📁 com.java.so
 - > 📄 ServBook.java
 - > 📁 Libraries
 - > 📁 JavaScript Resources
 - > 📁 build
 - ▼ 📁 WebContent
 - ▼ 📁 META-INF
 - 📄 MANIFEST.MF
 - ▼ 📁 WEB-INF
 - 📁 lib
 - 📄 web.xml
 - 📄 formBook.html
 - 📄 index.html

Web files

index.html

```
<!DOCTYPE html>
<html>
<head>
<meta charset="ISO-8859-1">
<title>Insert title here</title>
</head>

<body>
Hello Class, how are u? Loving persistence, entities and servlets?
Really?
</body>

</html>
```

formBook.html

```
<!DOCTYPE html>
<html>
<head>
<meta charset="ISO-8859-1">
<title>Insert title here</title>
</head>
<body>
    <form action="ServBook" method="post">
        Book:<br>
        <br>
        <br> Book Title:<br> <input type="text" name="bookTitle"
            value="Anna Karenina"><br> Author:<br> <input
            type="text" name="author" value="Tolstoi"><br>
ISBN:<br>
        <input type="text" name="isbn"
value="78-1-2345-6789-7|2-8765-4301-X"><br>
        <br>
        <br>
        <br> Language:<br> <input type="radio" name="lang"
            value="russian" checked> Original Russian<br> <input
            type="radio" name="lang" value="english"> English<br>
<input
            type="radio" name="lang" value="spanish"> Spanish <br>
<input
            type="radio" name="lang" value="catalan"> Catalan <br>
<br>
        <br>
        <br> Type:<br> <select name="typeBook" multiple>
            <option value="eBook">eBook</option>
            <option value="paper">Paper</option>

        </select> <input type="submit" value="Submit">
    </form>

</body>
</html>
```

Java files

servlet

```
package com.java.so;

import java.io.IOException;
import java.io.PrintWriter;

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

public class ServBook extends HttpServlet {
    private static final long serialVersionUID = 1L;

    protected void doPost(HttpServletRequest request,
        HttpServletResponse response)
        throws ServletException, IOException {

        PrintWriter pw = response.getWriter();
        pw.println(request.getParameter("bookTitle"));
        pw.println(request.getParameter("author"));
        pw.println(request.getParameter("isbn"));

        pw.println(request.getParameter("lang"));

        String[] list = request.getParameterValues("typeBook");

        for (int j = 0; j < list.length; j++) {
            pw.println(list[j]);
        }

        pw.close();
    }
}
```

Configuration files

web.xml

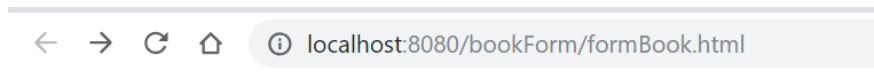
```
<?xml version="1.0" encoding="UTF-8"?>
<web-app xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
  xmlns="http://java.sun.com/xml/ns/javaee"
  xsi:schemaLocation="http://java.sun.com/xml/ns/javaee
http://java.sun.com/xml/ns/javaee/web-app_2_5.xsd"
  id="WebApp_ID" version="2.5">
  <display-name>bookForm</display-name>
  <welcome-file-list>
    <welcome-file>index.html</welcome-file>
    <welcome-file>index.htm</welcome-file>
    <welcome-file>index.jsp</welcome-file>
    <welcome-file>default.html</welcome-file>
    <welcome-file>default.htm</welcome-file>
    <welcome-file>default.jsp</welcome-file>
  </welcome-file-list>

  <servlet>
    <description></description>
    <display-name>ServBook</display-name>
    <servlet-name>ServBook</servlet-name>
    <servlet-class>com.java.so.ServBook</servlet-class>
  </servlet>

  <servlet-mapping>
    <servlet-name>ServBook</servlet-name>
    <url-pattern>/ServBook</url-pattern>
  </servlet-mapping>

</web-app>
```

output



Book:

Book Title:

Anna Karenina

Author:

Tolstoi

ISBN:

78-1-2345-6789-7|2-8765-4

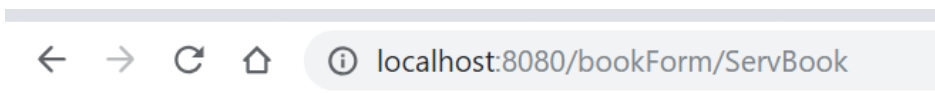
Language:

- Original Russian
- English
- Spanish
- Catalan

Type:

eBook
Paper

Submit



Anna Karenina

Tolstoi

78-1-2345-6789-7|2-8765-4301-X

russian

paper

Servlets Exemple3 StudentForm (NO HAY WEB.XML)

SetUp Project Dynamic Web Project

create servlets 3.1

New Dynamic Web Project

Dynamic Web Project

A project already exists with this name.

Project name:

Project location

Use default location

Location:

Target runtime

Dynamic web module version

Configuration

Hint: Get started quickly by selecting one of the pre-defined project configurations.

EAR membership

Add project to an EAR

















EAR project name:

Working sets

Add project to working sets

Working sets:

project

- ▼  studentForm
 - >  Deployment Descriptor: studentForm
 - >  JAX-WS Web Services
 - ▼  Java Resources
 - ▼  src
 - ▼  com.java.so
 - >  FormStudent.java
 - >  Libraries
 - >  JavaScript Resources
 - >  build
 - ▼  WebContent
 - ▼  META-INF
 -  MANIFEST.MF
 - ▼  WEB-INF
 -  lib
 -  index.html

Java files

servlet

```
package com.java.so;

import java.io.IOException;
import java.io.PrintWriter;

import javax.servlet.ServletException;
import javax.servlet.annotation.WebServlet;
import javax.servlet.http.HttpServlet;
import javax.servlet.http.HttpServletRequest;
import javax.servlet.http.HttpServletResponse;

@WebServlet("/FormStudent")
public class FormStudent extends HttpServlet {
    private static final long serialVersionUID = 1L;

    protected void doPost (HttpServletRequest request,
        HttpServletResponse response)
        throws ServletException, IOException {

        String email = request.getParameter("email");
        String username = request.getParameter("uname");
        String password = request.getParameter("pass");
        String gender = request.getParameter("gender");

        String[] course = request.getParameterValues("course");

        response.setContentType("text/html");

        PrintWriter pw = response.getWriter();
        pw.write("<h2> Following data received sucessfully.. <h2>
<br>");
        pw.write("<h3> Email: " + email + " </h3>");
        pw.write("<h4> User name: " + username + " </h4>");
        pw.write("<h4> Password: " + password + " </h4>");
        pw.write("<h4> Gender: " + gender + " </h4>");
        pw.write("<h5> Course: ");
```

```
    for (String c : course) {  
        pw.write(c + " ");  
    }  
    pw.write("</h5>");  
  
    pw.close();  
} }
```

Web files

index.html

```
<!DOCTYPE html>
<html>
<head>
<meta charset="UTF-8">
<title>Insert title here</title>
</head>
<body>

    Student FORM with SERVLET 3.1
    <br>
    <br>
    <form action="FormStudent" method="post">
        Email : <input type="text" name="email"> <br> User Name:
        <input type="text" name="uname"> <br> Password: <input
            type="password" name="pass"> <br> Gender : <input
            type="radio" name="gender" value="male" checked> Male
    <input
        type="radio" name="gender" value="female"> Female
    <input
        type="radio" name="gender" value="other"> <br> Course:
    <input
        type="checkbox" name="course" value="Java"> Java
    <input
        type="checkbox" name="course" value="C"> C <input
        type="checkbox" name="course" value="js"> JS <br>
    <input
        type="submit" value="Submit"> <input type="reset">
    </form>

</body>
</html>
```

output

← → ↻ ⓘ localhost:8080/bookFormServ/index.html

Student FORM with SERVELET 3.1

Email :

User Name:

Password:

Gender : Male Female

Course: Java C JS

← → ↻ ⓘ localhost:8080/studentForm/index.html

Student FORM with SERVELET 3.1

Email :

User Name:

Password:

Gender : Male Female

Course: Java C JS

← → ↻ ⓘ localhost:8080/studentForm/FormStudent

Following data received sucessfully..

Email: natalia@natalia.com

User name: natalia45

Password: natRocks1234;
















Gender: female

Course: Java

Sevlets Exemple4 Login **(NO HAY WEB.XML)**

SetUp Project Dynamic Web Project

project

- ▼  loginServlet
 - >  Deployment Descriptor: loginServlet
 - >  JAX-WS Web Services
 - ▼  Java Resources
 - ▼  src
 - ▼  com.java.so
 - >  Login.java
 - >  Libraries
 - >  JavaScript Resources
 - >  build
 - ▼  WebContent
 - >  META-INF
 - ▼  WEB-INF
 -  lib
 -  index.html

Java files

servlet

```
package com.java.so;

import java.io.IOException;
import java.io.PrintWriter;

import javax.servlet.annotation.WebServlet;
import javax.servlet.http.HttpServlet;
import javax.servlet.http.HttpServletRequest;
import javax.servlet.http.HttpServletResponse;

@WebServlet("/Login")
public class Login extends HttpServlet {
    private static final long serialVersionUID = 1L;
    protected void doPost (HttpServletRequest request,
        HttpServletResponse response) throws IOException {

        response.setContentType("text/html");

        // Post Parameters From The Request
        String param1 = request.getParameter("username");
        String param2 = request.getParameter("password");

        PrintWriter pw = response.getWriter();

        if(param1 == null || param2 == null) {
            // The Request Parameters Were Not Present In The Query
            String. Do Something Or Exception Handling !!
            pw.write("<html><body><div id='serlvetResponse'
style='text-align: center;'>");
        } else if ("".equals(param1) || "".equals(param2)) {
            // The Request Parameters Were Present In The Query String
            But Has No Value. Do Something Or Exception Handling !!
            pw.write("<html><body><div id='serlvetResponse'
style='text-align: center;'>");
        } else {
            System.out.println("Username?= " + param1 + ", Password?= "
+ param2);

            // Print The Response
```



```
        pw.write("<html><body><div id='serlvetResponse'
style='text-align: center;'>");

        // Authentication Logic & Building The Html Response Code
        if((param1.equalsIgnoreCase("Natalia")) &&
(param2.equals("nataliaRocks@1234"))) {
            pw.write("<h2>Servlet Application Login Example</h2>");
            pw.write("<p style='color: green; font-size:
large;'>Congratulations! <span style='text-transform: capitalize;'>" +
param1 + "</span>, Natalia, you are an authorised login, and welcomed
again!</p>");
        } else {
            pw.write("<p style='color: red; font-size: larger;'>You
are not an authorised user! Please check with administrator! Your are
not Natalia!</p>");
        }
        pw.write("</div></body></html>");
        pw.close();
    }
}
}
```

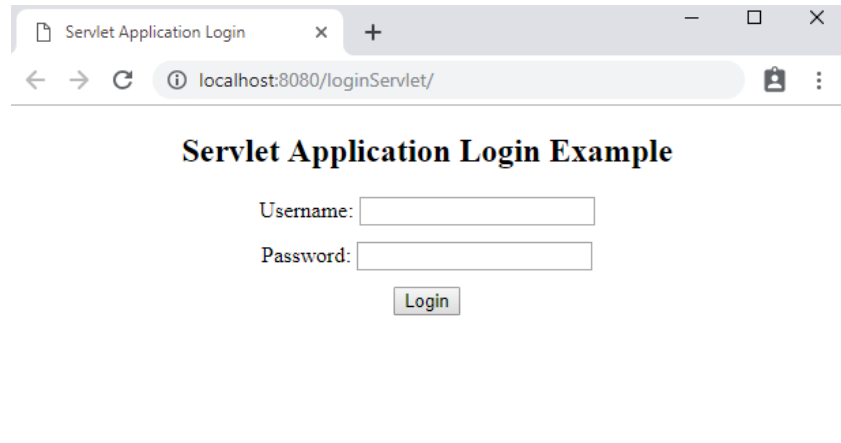
Web files

index.html

```
<!DOCTYPE html PUBLIC "-//W3C//DTD HTML 4.01 Transitional//EN"
"http://www.w3.org/TR/html4/loose.dtd">
<html>
  <head>
    <meta http-equiv="Content-Type" content="text/html;
charset=UTF-8">
    <title>Servlet Application Login</title>
    <style type="text/css">
      .paddingBtm {
        padding-bottom: 12px;
      }
    </style>
  </head>
  <body>
    <center>
      <h2>Servlet Application Login Example</h2>
      <form id="login" name="loginForm" method="post"
action="Login">
        <div id="usernameDiv" class="paddingBtm">
          <span id="user">Username: </span><input type="text"
name="username" />
        </div>
        <div id="passwordDiv" class="paddingBtm">
          <span id="pass">Password: </span><input
type="password" name="password" />
        </div>
        <div id="loginBtn">
          <input id="btn" type="submit" value="Login" />
        </div>
      </form>
    </center>
  </body>
</html>
```

Output

<http://localhost:8080/loginServlet/>

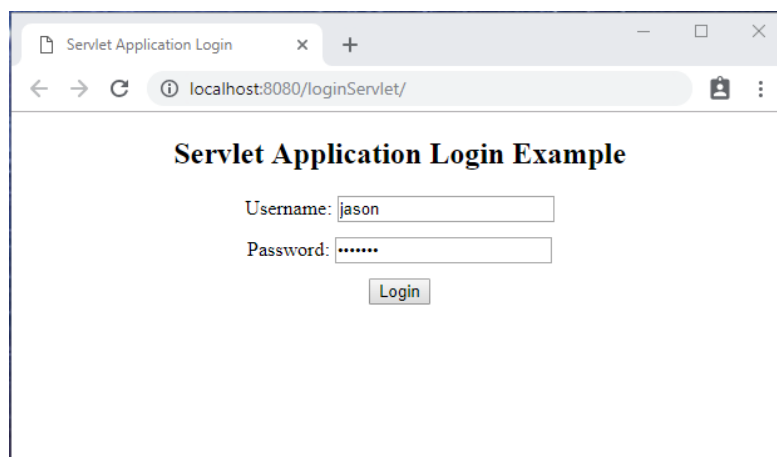


Servlet Application Login Example

Username:

Password:

Login

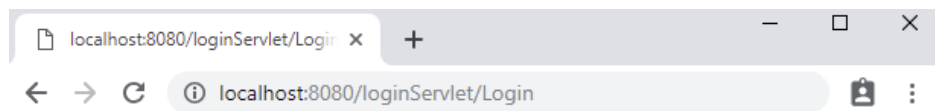


Servlet Application Login Example

Username:

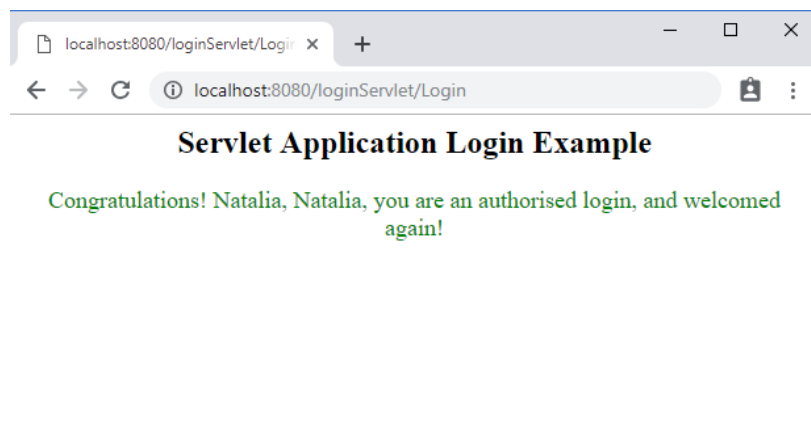
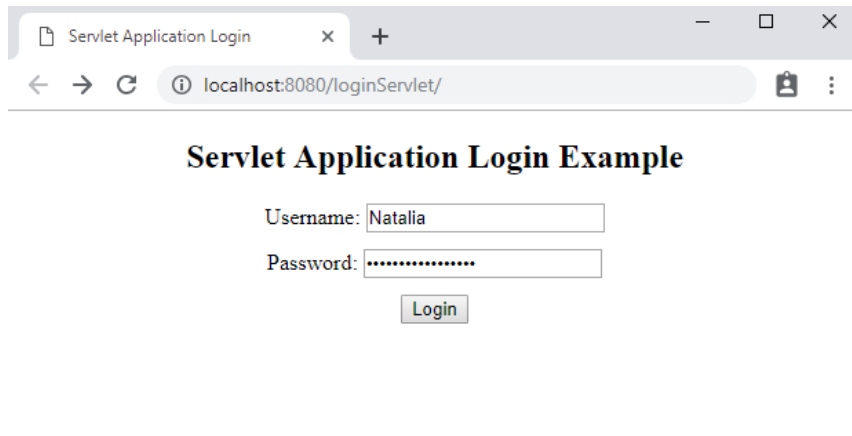
Password:

Login



localhost:8080/loginServlet/Login

You are not an authorised user! Please check with administrator! Your are not Natalia!



Exemple:

```
@WebServlet(  
    name = "HelloServlet",  
    urlPatterns = {"/hello", "/helloBoy"},  
    asyncSupported = false,  
    initParams = {  
        @WebInitParam(name = "name", value = "admin"),  
        @WebInitParam(name = "name1", value = "admin1"),  
        @WebInitParam(name = "name2", value = "admin2")  
    }  
)  
public class HelloServlet extends HttpServlet {  
  
    ....  
}
```