Annotations

The @WebServlet annotation is used to declare a servlet.

The annotated class must extend the javax.servlet.http.HttpServlet class.

**Syntax**

|  |  |  |
| --- | --- | --- |
|  | |  | | --- | | @WebServlet(      attribute1=value1,      attribute2=value2,     ...  )  public class TheServlet extends javax.servlet.http.HttpServlet {      // servlet code...  } | |

**Attributes**

|  |  |  |  |
| --- | --- | --- | --- |
| **Name** | **Type** | **Required** | **Description** |
| **value**  or  **urlPatterns** | *String[]* | Required | Specify one or more URL patterns of the servlet. Either of attribute can be used, but not both. |
| **name** | *String* | Optional | Name of the servlet |
| **displayName** | *String* | Optional | Display name of the servlet |
| **description** | *String* | Optional | Description of the servlet |
| **initParams** | *WebInitParam[]* | Optional | Specify one or more initialization parameters of the servlet. Each parameter is specified by [@WebInitParam](http://www.codejava.net/java-ee/servlet/webinitparam-annotation-examples) annotation type. |
| **loadOnStartup** | *int* | Optional | Specify load-on-startup order of the servlet. |

**Examples**

* **A servlet is annotated with only the URL pattern:**

|  |  |
| --- | --- |
|  | @WebServlet("/processForm")  public class MyServlet extends HttpServlet {      public void doGet(HttpServletRequest request, HttpServletResponse response)              throws IOException {          response.getWriter().println("Hello");      }  } |

* **A servlet is annotated with multiple URL patterns:**

|  |  |
| --- | --- |
|  | @WebServlet(urlPatterns = {"/sendFile", "/uploadFile"})  public class UploadServlet extends HttpServlet {      // implement servlet doPost() and doGet()...  } |

* **Declare a servlet with additional information:**

|  |  |
| --- | --- |
|  | @WebServlet(          name = "MyOwnServlet",          description = "This is my first annotated servlet",          urlPatterns = "/processServlet"  )  public class MyServlet extends HttpServlet {      // implement servlet doPost() and doGet()...  } |

* **Declare a servlet with some init parameters:**

|  |  |  |
| --- | --- | --- |
|  | @WebServlet(          urlPatterns = "/imageUpload",          initParams =          {              @WebInitParam(name = "saveDir", value = "D:/FileUpload"),              @WebInitParam(name = "allowedTypes", value = "jpg,jpeg,gif,png")          }  )  public class ImageUploadServlet extends HttpServlet {        public void doGet(HttpServletRequest request, HttpServletResponse response)              throws IOException {          String saveDir = getInitParameter("saveDir");          String fileTypes = getInitParameter("allowedTypes");            PrintWriter writer = response.getWriter();            writer.println("saveDir = " + saveDir);          writer.println("fileTypes = " + fileTypes);      }  } | |
|  | @WebServlet(          urlPatterns = "/myController",          loadOnStartup = 1,    )  public class StartupServlet extends HttpServlet {        public void init(ServletConfig config) {          System.out.println("My servlet has been initialized");      }        // implement servlet doPost() and doGet()...  } | |

Annotations for Filters

Examples

The following example registers a filter for the URL pattern /admin:

@WebFilter("/admin")

public class MyFilter implements Filter {

// implements Filter's methods here...

}

Apply a filter for all URLs:

@WebFilter("/\*")

public class MyFilter implements Filter {

// implements Filter's methods here...

}

Register a filter for a specific servlet:

@WebFilter(servletNames = "MyOwnServlet")

public class MyFilter implements Filter {

// implements Filter's methods here...

}

Register a filter for multiple servlets:

@WebFilter(servletNames = {"MyOwnServlet", "UploadServlet"})

public class MyFilter implements Filter {

// implements Filter's methods here...

}

Specify initialization parameters for the filter:

@WebFilter(

urlPatterns = "/uploadFilter",

initParams = @WebInitParam(name = "fileTypes", value = "doc;xls;zip;txt;jpg;png;gif")

)

public class UploadFilter implements Filter {

// implements Filter's methods here...

}

Listeners

@WebListener([optional description])

**import** java.io.IOException;

**import** java.io.PrintWriter;

**import** javax.servlet.ServletConfig;

**import** javax.servlet.ServletException;

**import** javax.servlet.annotation.WebInitParam;

**import** javax.servlet.annotation.WebServlet;

**import** javax.servlet.http.HttpServlet;

**import** javax.servlet.http.HttpServletRequest;

**import** javax.servlet.http.HttpServletResponse;

/\*\*

\* Servlet implementation class WelcomeServlet

\*/

@WebServlet(urlPatterns = "/welcome",

initParams = @WebInitParam(name = "myname", value = "myValue"),

loadOnStartup = 1)

**public** **class** WelcomeServlet **extends** HttpServlet {

**private** **static** **final** **long** ***serialVersionUID*** = 1L;

/\*\*

\* **@see** HttpServlet#HttpServlet()

\*/

**public** WelcomeServlet() {

**super**();

// **TODO** Auto-generated constructor stub

}

@Override

**public** **void** init(ServletConfig config) **throws** ServletException {

System.***out***.println("Init Method.....");

**super**.init(config);

}

/\*\*

\* **@see** HttpServlet#service(HttpServletRequest request, HttpServletResponse response)

\*/

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

PrintWriter pw = response.getWriter();

pw.println("<b> Welcome.....</b>");

ServletConfig config = getServletConfig();

System.***out***.println("Init Param value....." + config.getInitParameter("myname"));

}

}

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("/SimpleWebServlet")

public class WebServletDemo extends HttpServlet

{

private static final long serialVersionUID = 1L;

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

{

response.setContentType ("text/html");

PrintWriter out = response.getWriter ();

out.print ("<html><body>");

out.print ("<h3>Hello Servlet</h3>");

out.print ("</body></html>");

}

}

----------------------------------

import java.io.IOException;

import java.io.PrintWriter;

import javax.servlet.ServletException;

import javax.servlet.annotation.WebInitParam;

import javax.servlet.annotation.WebServlet;

import javax.servlet.http.HttpServlet;

import javax.servlet.http.HttpServletRequest;

import javax.servlet.http.HttpServletResponse;

@WebServlet(value = "/Simple", initParams = { @WebInitParam(name = "s1", value = "Hello "), @WebInitParam(name = "s2", value = " World!") })

public class InitParamsDemo extends HttpServlet

{

private static final long serialVersionUID = 1L;

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

{

response.setContentType ("text/html");

PrintWriter out = response.getWriter ();

out.print ("<html><body>");

out.print ("<h3>Hello Servlet</h3>");

out.println (getInitParameter ("s1"));

out.println (getInitParameter ("s2"));

out.print ("</body></html>");

}

}

---------------------------------------------------------------------------------------------------------------------------------

import java.io.IOException;

import javax.servlet.annotation.WebFilter;

import javax.servlet.annotation.WebInitParam;

import javax.servlet.\*;

import java.util.\*;

// Implements Filter class

@WebFilter(urlPatterns = { "/\*" }, initParams = {@WebInitParam(name = "test-param", value = "Initialization Paramter") })

public class LogFilter implements Filter

{

public void init (FilterConfig config) throws ServletException

{

// Get init parameter

String testParam = config.getInitParameter ("test-param");

// Print the init parameter

System.out.println ("Test Param: " + testParam);

}

public void doFilter (ServletRequest request, ServletResponse response, FilterChain chain) throws IOException, ServletException

{

// Log the current timestamp.

System.out.println ("Time " + new Date ().toString ());

// Pass request back down the filter chain

chain.doFilter (request, response);

}

public void destroy ()

{

/\*

\* Called before the Filter instance is removed from service by the web

\* container

\*/

}

}

-------------------------------------------------------------------------------------------------------------------------

import javax.servlet.annotation.WebListener;

import javax.servlet.http.HttpSessionAttributeListener;

import javax.servlet.http.HttpSessionBindingEvent;

import javax.servlet.http.HttpSessionEvent;

import javax.servlet.http.HttpSessionListener;

@WebListener()

public class WebListenerExample implements HttpSessionListener, HttpSessionAttributeListener

{

public void sessionCreated (HttpSessionEvent he)

{

System.out.println ("Session is created");

}

public void sessionDestroyed (HttpSessionEvent he)

{

System.out.println ("Session is destroyed");

}

public void attributeAdded (HttpSessionBindingEvent arg0)

{

System.out.println ("value is added");

}

public void attributeRemoved (HttpSessionBindingEvent arg0)

{

System.out.println ("value is removed");

}

public void attributeReplaced (HttpSessionBindingEvent arg0)

{

System.out.println ("value has been replaced");

}

}

import java.io.IOException;

import java.io.PrintWriter;

import javax.servlet.annotation.WebServlet;

import javax.servlet.ServletException;

import javax.servlet.http.HttpSession;

import javax.servlet.http.HttpServlet;

import javax.servlet.http.HttpServletRequest;

import javax.servlet.http.HttpServletResponse;

@WebServlet (name = "ListenerTester", urlPatterns = "/listener")

public class ListenerTester extends HttpServlet

{

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

{

response.setContentType ("text/html");

PrintWriter out = response.getWriter ();

HttpSession session = request.getSession ();

out.println("<h3>This is a simple example of @WebServlet with @WebListener</h3>");

session.setAttribute ("info", "My Info");

String s = (String) session.getAttribute ("info");

out.println ("<br>Attribute value that you have set = " + s);

session.setAttribute ("info", "Your Info");

String s1 = (String) session.getAttribute ("info");

out.println ("<br>Attribute value has replaced by = " + s1);

session.removeAttribute ("info");

session.invalidate ();

}

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

{

doGet (request, response);

}

}

web.xml

Here <listener></listener> entry is not required in web.xml file but in the older version, you were bound to map this entry into web.xml file.

------------------------------------------------------------------------------------------------------------------------------------------

