

SERVLET LISTENERS

Servlet Listeners:

Servlet listeners are interfaces that contain event handlers to handle events which are generated by web container.

List of servlet listeners:

- 1) ServletContextListener
- 2) ServletContextAttributeListener
- 3) HttpSessionListener
- 4) HttpSessionAttributeListener, .. etc.,

ServletContextListener:

It contains the following event handlers to handle ServletContextEvent which is generated by web container.

- 1) public default void contextInitialized(ServletContextEvent);
- 2) public default void contextDestroyed(ServletContextEvent);

ServletContextEvent is generated by web container whenever ServletContext is created & destroyed.

ServletContext is created by web container whenever web application is deployed.

ServletContext is destroyed by web container whenever web application is undeployed.

Example:

MyListener.java:

```
import java.sql.*;
import javax.servlet.*;

public class MyListener implements ServletContextListener
{
    Connection con;

    public void contextInitialized(ServletContextEvent sce)
    {
        try{
            Class.forName("oracle.jdbc.driver.OracleDriver");
            con=DriverManager.getConnection("jdbc:oracle:thin:@localhost:1521:xe","system","manager");

            ServletContext sc=sce.getServletContext();
            sc.setAttribute("oracle", con);
        }catch(Exception e)
        {
            System.err.println(e);
        }
    }

    public void contextDestroyed(ServletContextEvent sce)
    {
    }
```

```
try{
    con.close();
}catch(Exception e)
{
    System.err.println(e);
}
}
```

MyServlet.java:

```
import java.io.*;
import java.sql.*;
import javax.servlet.*;
import javax.servlet.http.*;

public class MyServlet extends HttpServlet
{
    public void doGet(HttpServletRequest hreq, HttpServletResponse hres)
    {
        try{
            ServletContext sc=getServletContext();
            Connection con=(Connection)sc.getAttribute("oracle");
            PrintWriter pw=hres.getWriter();
            pw.println("Connection Object Obtained Successfully");
        }
    }
}
```

```
}catch(Exception e)
{
    System.err.println(e);
}
}
}
```

web.xml:

```
<web-app>
<listener>
<listener-class>MyListener</listener-class>
</listener>
<servlet>
<servlet-name>demo</servlet-name>
<servlet-class>MyServlet</servlet-class>
</servlet>
<servlet-mapping>
<servlet-name>demo</servlet-name>
<url-pattern>/test</url-pattern>
</servlet-mapping>
</web-app>
```

ServletContextAttributeListener:

It contains the following event handlers to handle ServletContextAttributeEvent which is generated by web container.

- 1) public default void
attributeAdded(javax.servlet.ServletContextAttributeEvent);
- 2) public default void
attributeRemoved(javax.servlet.ServletContextAttributeEvent);
- 3) public default void
attributeReplaced(javax.servlet.ServletContextAttributeEvent);

ServletContextAttributeEvent is generated by web container whenever attribute is added to ServletContext, attribute is removed from ServletContext and attribute is replaced in a ServletContext.

HttpSessionListener:

It contains the following event handlers to handle HttpSessionEvent which is generated by the web container.

- 1) public default void
sessionCreated(javax.servlet.http.HttpSessionEvent);
- 2) public default void
sessionDestroyed(javax.servlet.http.HttpSessionEvent);

HttpSessionEvent is generated by web container whenever session is created & destroyed.

HttpSessionAttributeListener:

It contains the following event handlers to handle HttpSessionBindingEvent which is generated by web container.

- 1) public default void
attributeAdded(javax.servlet.http.HttpSessionBindingEvent);
- 2) public default void
attributeRemoved(javax.servlet.http.HttpSessionBindingEvent);
- 3) public default void
attributeReplaced(javax.servlet.http.HttpSessionBindingEvent);

HttpSessionBindingEvent is generated by web container whenever attribute is added to session, attribute is removed from session and attribute is replaced in a session.

By

Mr. Venkatesh Mansani

Naresh i Technologies