B.Sc. In Software Development. Year 4.
Semester I. Enterprise Development.
Working With Listeners.

LIMERICK INSTITUTE OF TECHNOLOGY
SCHOOL OF SCIENCE, ENGINEERING & I.T.
Department of Information Technology

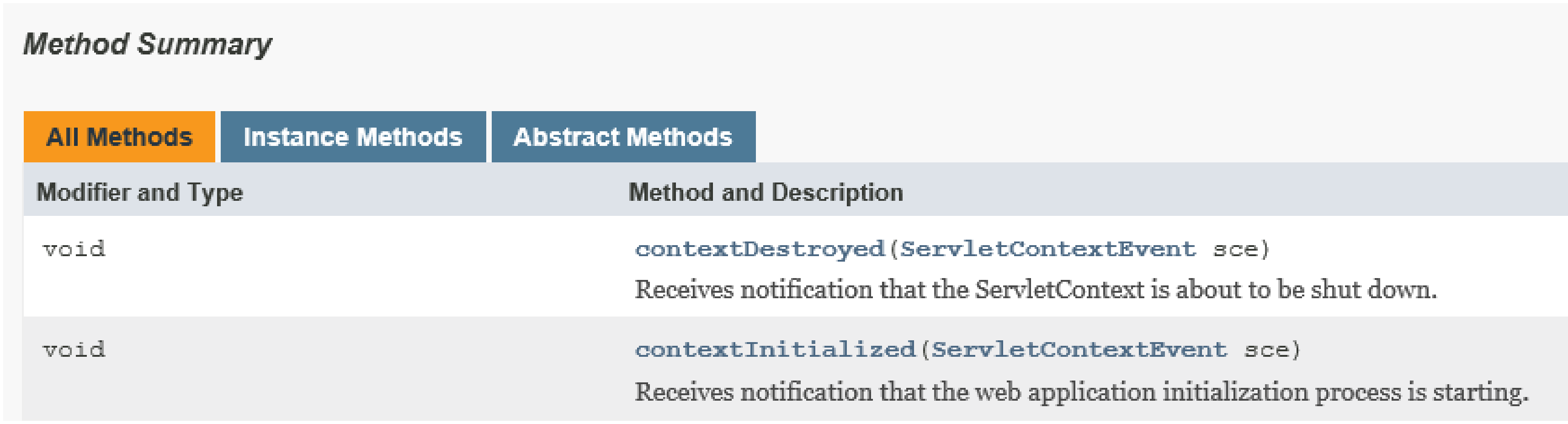
Introduction

- Introduced to the Servlet specification with version 2.3.
- They execute when an event occurs in a web application.
- 6 inbuilt types.
- Two examples covered today:
  1. ServletContextListener
  2. HttpSessionListener

Types of Listeners

ServletContextListener

Observes when a servlet context is created or about to be shut down.


Method Summary

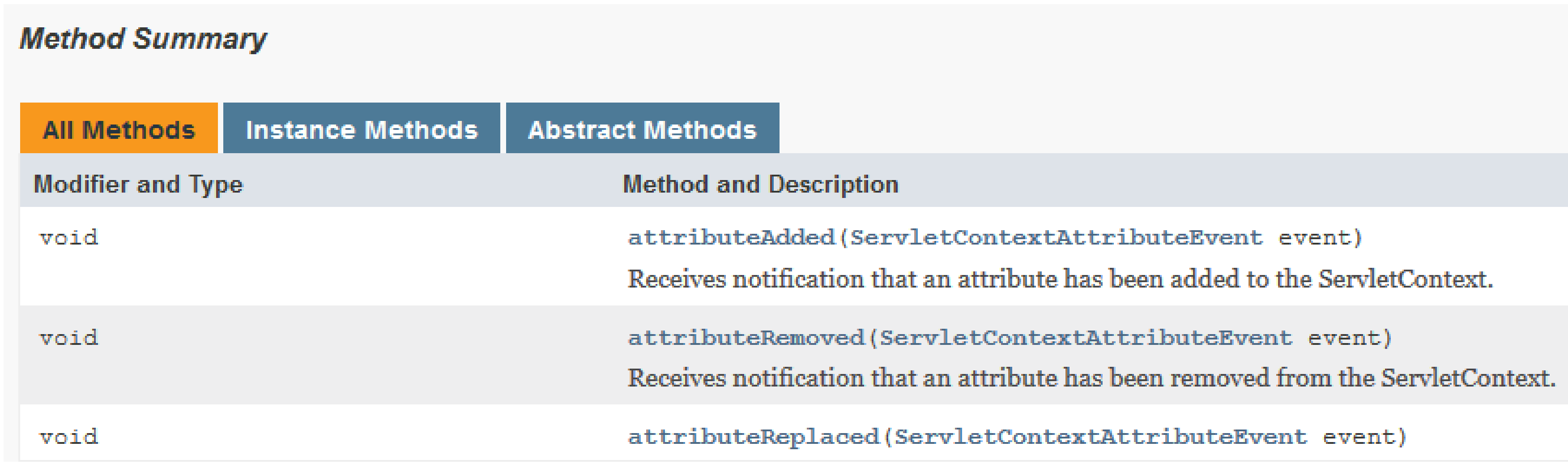
All Methods | Instance Methods | Abstract Methods

Modifier and Type	Method and Description
`void`	`contextDestroyed(ServletContextEvent sce)` Receives notification that the ServletContext is about to be shut down.
`void`	`contextInitialized(ServletContextEvent sce)` Receives notification that the web application initialization process is starting.

Types of Listeners

ServletContextAttributeListener

Observes the servlet context's attributes lifecycle.



Method Summary

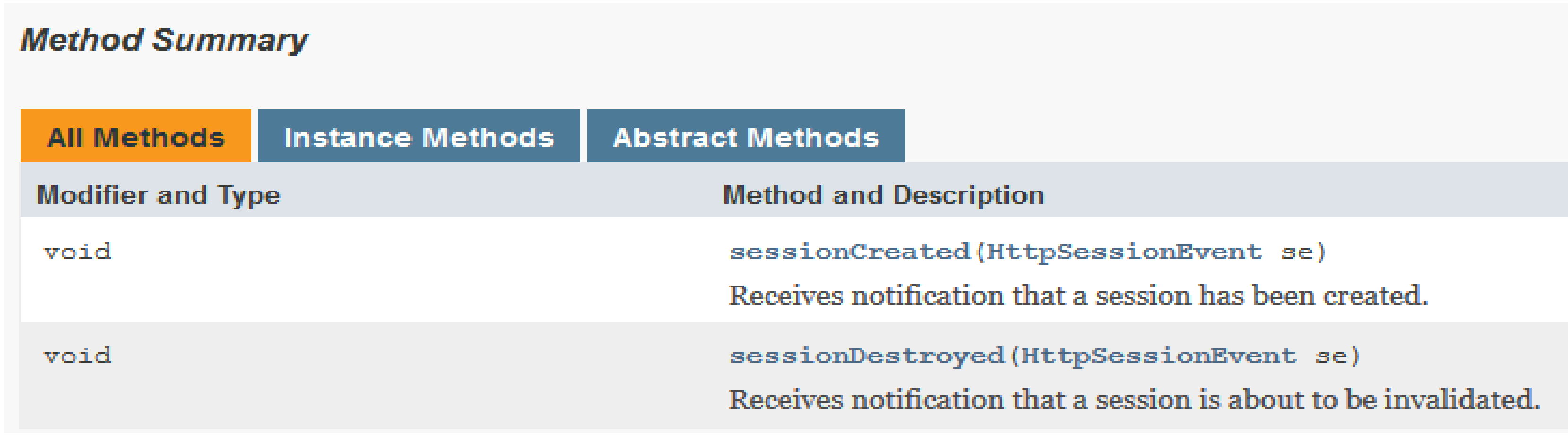
All Methods | Instance Methods | Abstract Methods

Modifier and Type	Method and Description
`void`	`attributeAdded(ServletContextAttributeEvent event)` Receives notification that an attribute has been added to the ServletContext.
`void`	`attributeRemoved(ServletContextAttributeEvent event)` Receives notification that an attribute has been removed from the ServletContext.
`void`	`attributeReplaced(ServletContextAttributeEvent event)`

Types of Listeners

HttpSessionListener

Observes when a HTTP session is created or about to be destroyed.



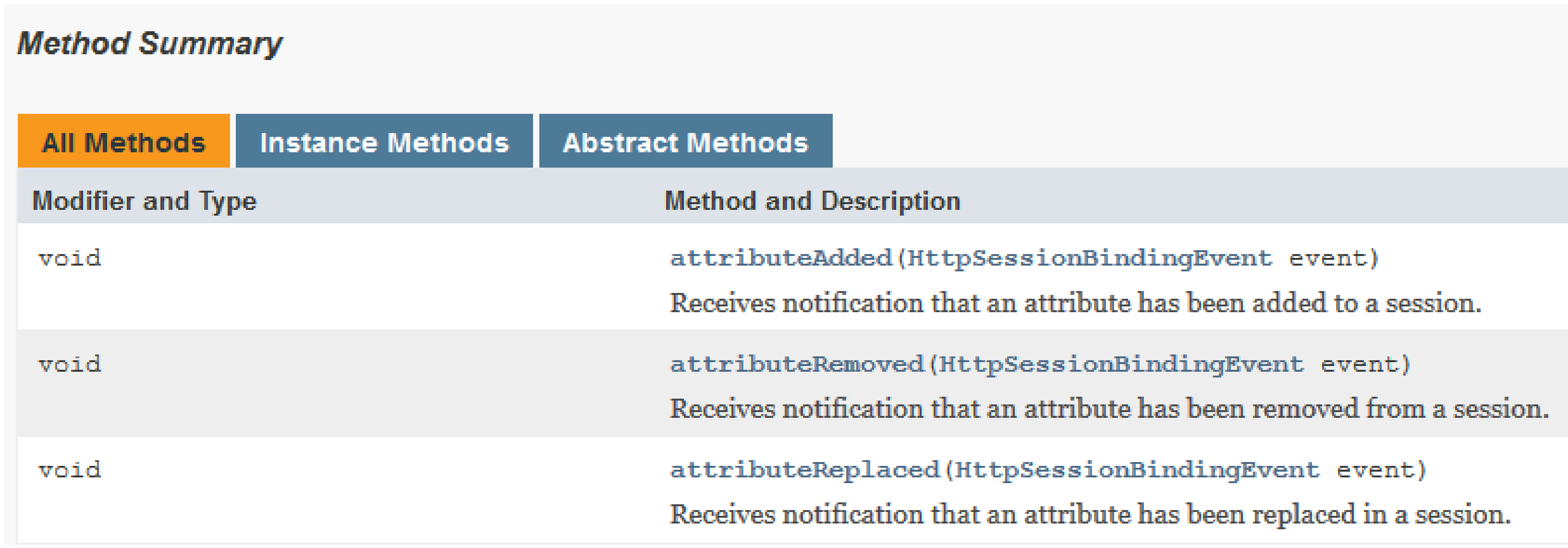
Method Summary

Modifier and Type	Method and Description
`void`	`sessionCreated(HttpSessionEvent se)` Receives notification that a session has been created.
`void`	`sessionDestroyed(HttpSessionEvent se)` Receives notification that a session is about to be invalidated.

Types of Listeners

HttpSessionAttributeListener

Observes events relating to HttpSession attribute changes.

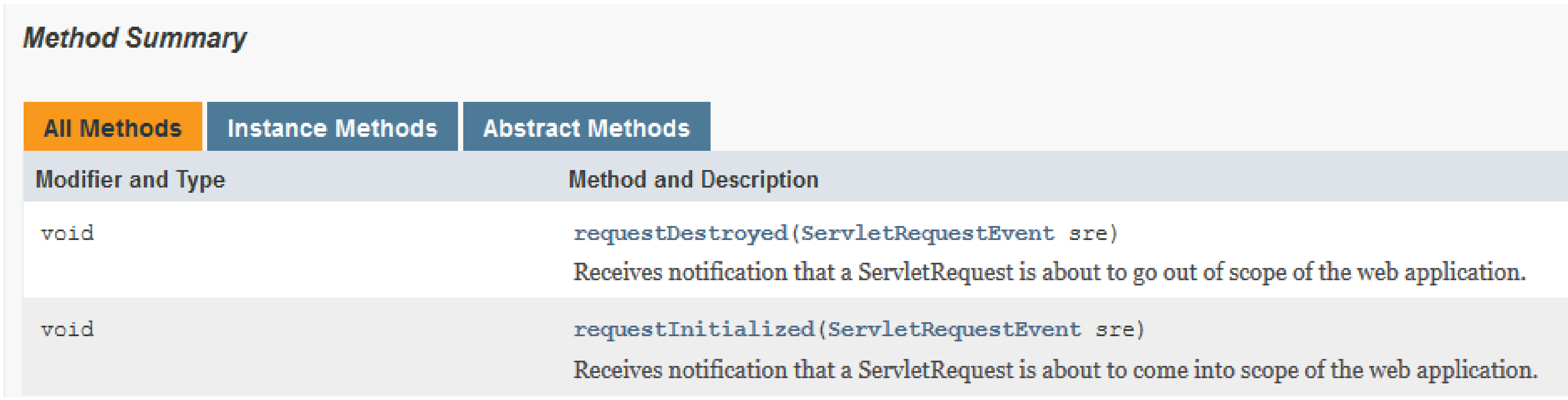

Method Summary

Modifier and Type	Method and Description
`void`	`attributeAdded(HttpSessionBindingEvent event)` Receives notification that an attribute has been added to a session.
`void`	`attributeRemoved(HttpSessionBindingEvent event)` Receives notification that an attribute has been removed from a session.
`void`	`attributeReplaced(HttpSessionBindingEvent event)` Receives notification that an attribute has been replaced in a session.

Types of Listeners

ServletRequestListener

Observes events relating to requests coming into and going out of scope of a web application.

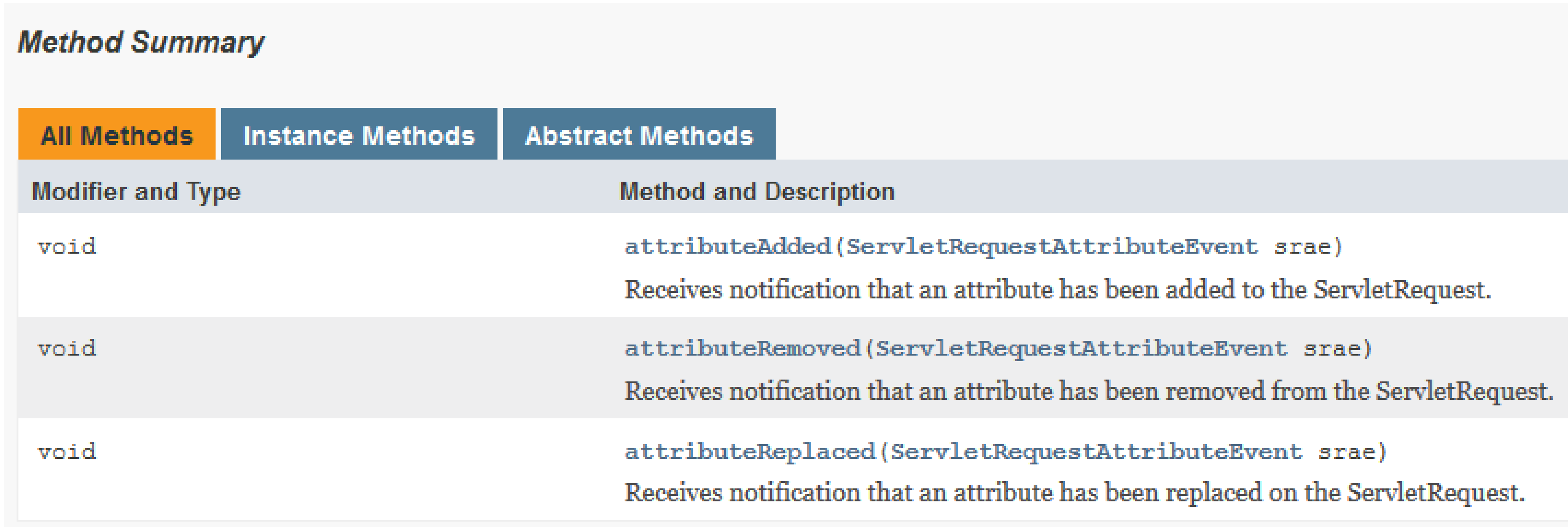

Method Summary

Modifier and Type	Method and Description
`void`	`requestDestroyed(ServletRequestEvent sre)` Receives notification that a ServletRequest is about to go out of scope of the web application.
`void`	`requestInitialized(ServletRequestEvent sre)` Receives notification that a ServletRequest is about to come into scope of the web application.

Types of Listeners

ServletRequestAttributeListener

Observes events that relate to ServletRequest attribute changes.



Method Summary

All Methods | Instance Methods | Abstract Methods

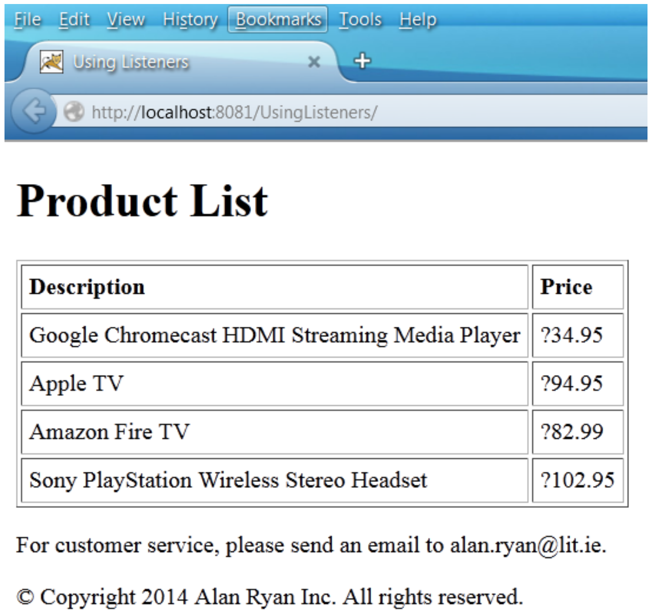
Modifier and Type	Method and Description
`void`	`attributeAdded(ServletRequestAttributeEvent srae)` Receives notification that an attribute has been added to the ServletRequest.
`void`	`attributeRemoved(ServletRequestAttributeEvent srae)` Receives notification that an attribute has been removed from the ServletRequest.
`void`	`attributeReplaced(ServletRequestAttributeEvent srae)` Receives notification that an attribute has been replaced on the ServletRequest.

Example 1: Using a ServletContextListener

- A *ServletContextListener* is used to determine when an application is started.
- Use its *contextInitialised* method to initialise one or more global variables when that event occurs.

Example 1: Using a ServletContextListener

- A simple example as it displays a list of products (which are contained in a file).
- Along with displaying a list of products, the application also displays a customer service email address and a copyright year.


File Edit View History Bookmarks Tools Help
Using Listeners
http://localhost:8081/UsingListeners/

Product List

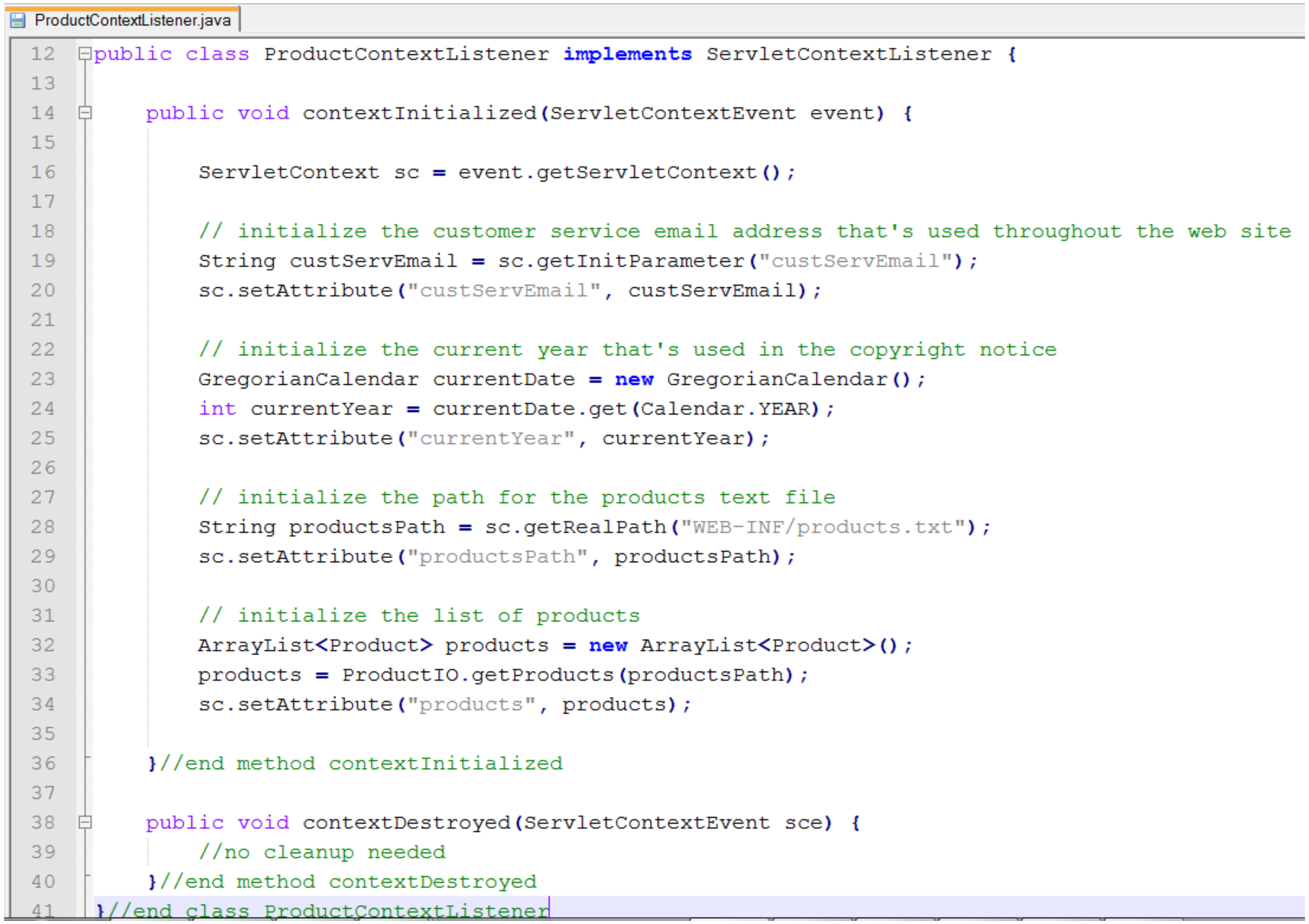
Description	Price
Google Chromecast HDMI Streaming Media Player	?34.95
Apple TV	?94.95
Amazon Fire TV	?82.99
Sony PlayStation Wireless Stereo Headset	?102.95

For customer service, please send an email to alan.ryan@lit.ie.

© Copyright 2014 Alan Ryan Inc. All rights reserved.

Example 1: Using a ServletContextListener

- The code for the listener is as follows:



```java
public class ProductContextListener implements ServletContextListener {

    public void contextInitialized(ServletContextEvent event) {

        ServletContext sc = event.getServletContext();

        // initialize the customer service email address that's used throughout the web site
        String custServEmail = sc.getInitParameter("custServEmail");
        sc.setAttribute("custServEmail", custServEmail);

        // initialize the current year that's used in the copyright notice
        GregorianCalendar currentDate = new GregorianCalendar();
        int currentYear = currentDate.get(Calendar.YEAR);
        sc.setAttribute("currentYear", currentYear);

        // initialize the path for the products text file
        String productsPath = sc.getRealPath("WEB-INF/products.txt");
        sc.setAttribute("productsPath", productsPath);

        // initialize the list of products
        ArrayList<Product> products = new ArrayList<Product>();
        products = ProductIO.getProducts(productsPath);
        sc.setAttribute("products", products);

    }//end method contextInitialized

    public void contextDestroyed(ServletContextEvent sce) {
        //no cleanup needed
    }//end method contextDestroyed
}//end class ProductContextListener
```

An aside – what is a ServletContext object

- The *ServletContext* is an object that contains meta information about your web application.
- You can access it via the *HttpRequest* object, like this:

```
ServletContext context = request.getSession().getServletContext();
```

- Just like in the session object you can store attributes in the servlet context like this:

```
context.setAttribute("someValue", "aValue");
```

- You can access the attributes again like this:

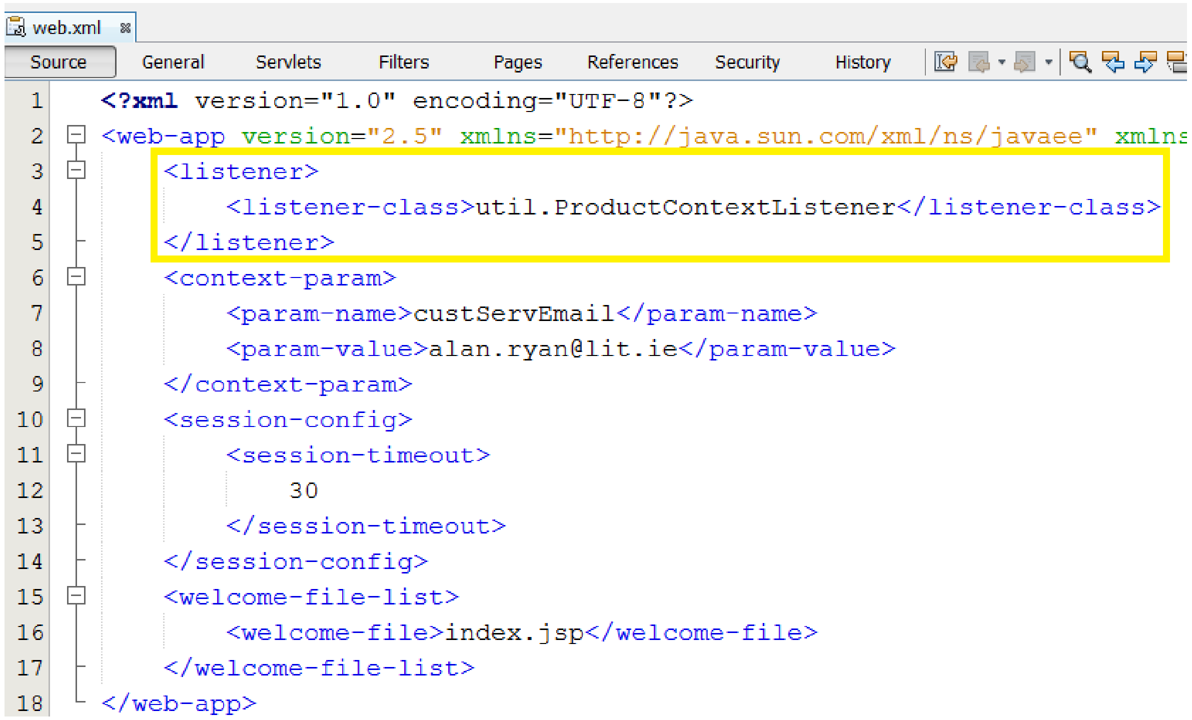
```
Object attribute = context.getAttribute("someValue");
```

An aside – what is a ServletContext object

- The attributes stored in the *ServletContext* are available to all servlets in your application, and between requests and sessions.
- That means, that the attributes are available to all clients.
    - Session attributes are just available to a single user.
- The *ServletContext* attributes are still stored in the memory of the servlet container.

How to register a listener

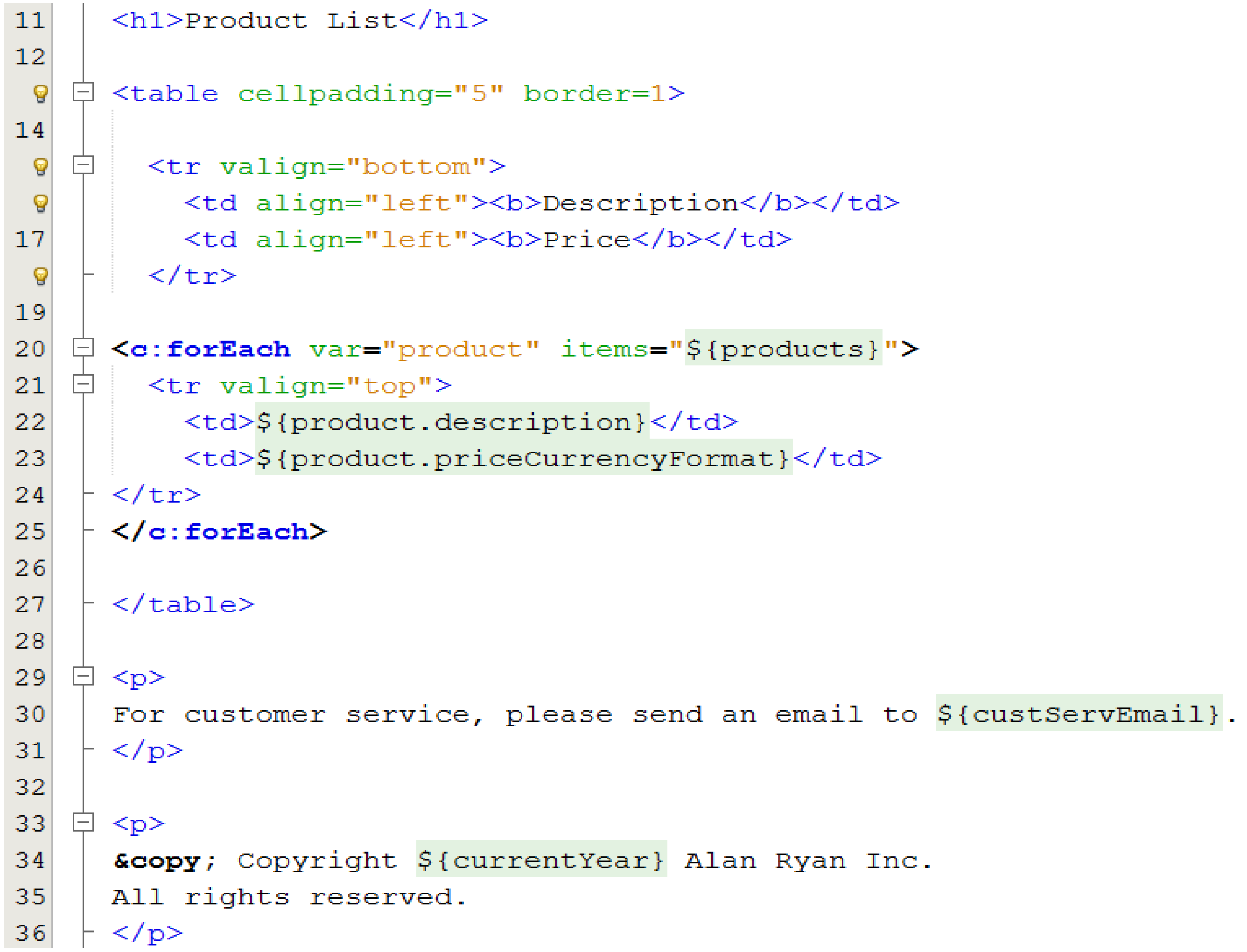
- After you code the listener you must register the listener with the web application.
- To do that you must add a listener element to the applications web.xml file.



```
<?xml version="1.0" encoding="UTF-8"?>
<web-app version="2.5" xmlns="http://java.sun.com/xml/ns/javaee" xmlns
    <listener>
        <listener-class>util.ProductContextListener</listener-class>
    </listener>
    <context-param>
        <param-name>custServEmail</param-name>
        <param-value>alan.ryan@lit.ie</param-value>
    </context-param>
    <session-config>
        <session-timeout>
            30
        </session-timeout>
    </session-config>
    <welcome-file-list>
        <welcome-file>index.jsp</welcome-file>
    </welcome-file-list>
</web-app>
```

How to code a JSP that uses listeners

- The (partial) code listing below shows a JSP that uses the attributes set by the listener.



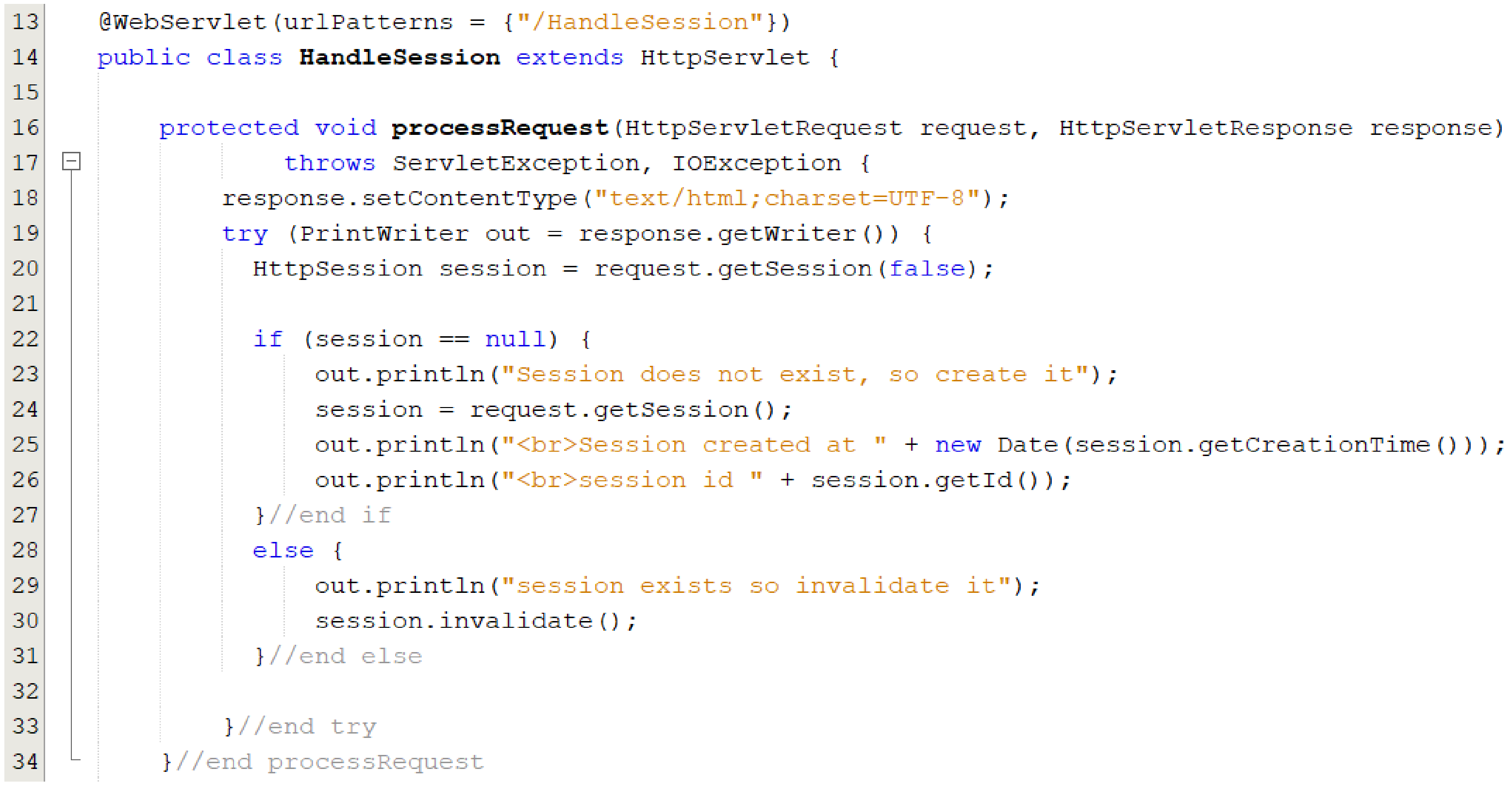
```
<h1>Product List</h1>

<table cellpadding="5" border=1>

  <tr valign="bottom">
    <td align="left"><b>Description</b></td>
    <td align="left"><b>Price</b></td>
  </tr>

<c:forEach var="product" items="${products}">
  <tr valign="top">
    <td>${product.description}</td>
    <td>${product.priceCurrencyFormat}</td>
</tr>
</c:forEach>

</table>

<p>
For customer service, please send an email to ${custServEmail}.
</p>

<p>
&copy; Copyright ${currentYear} Alan Ryan Inc.
All rights reserved.
</p>
```

Example 2: Counting Sessions

- Example to count the number of active sessions in a web application.
    - Every time a session is created -> increment a counter.
    - Every time a session is destroyed ->decrement a counter.

Example 2: Counting Sessions

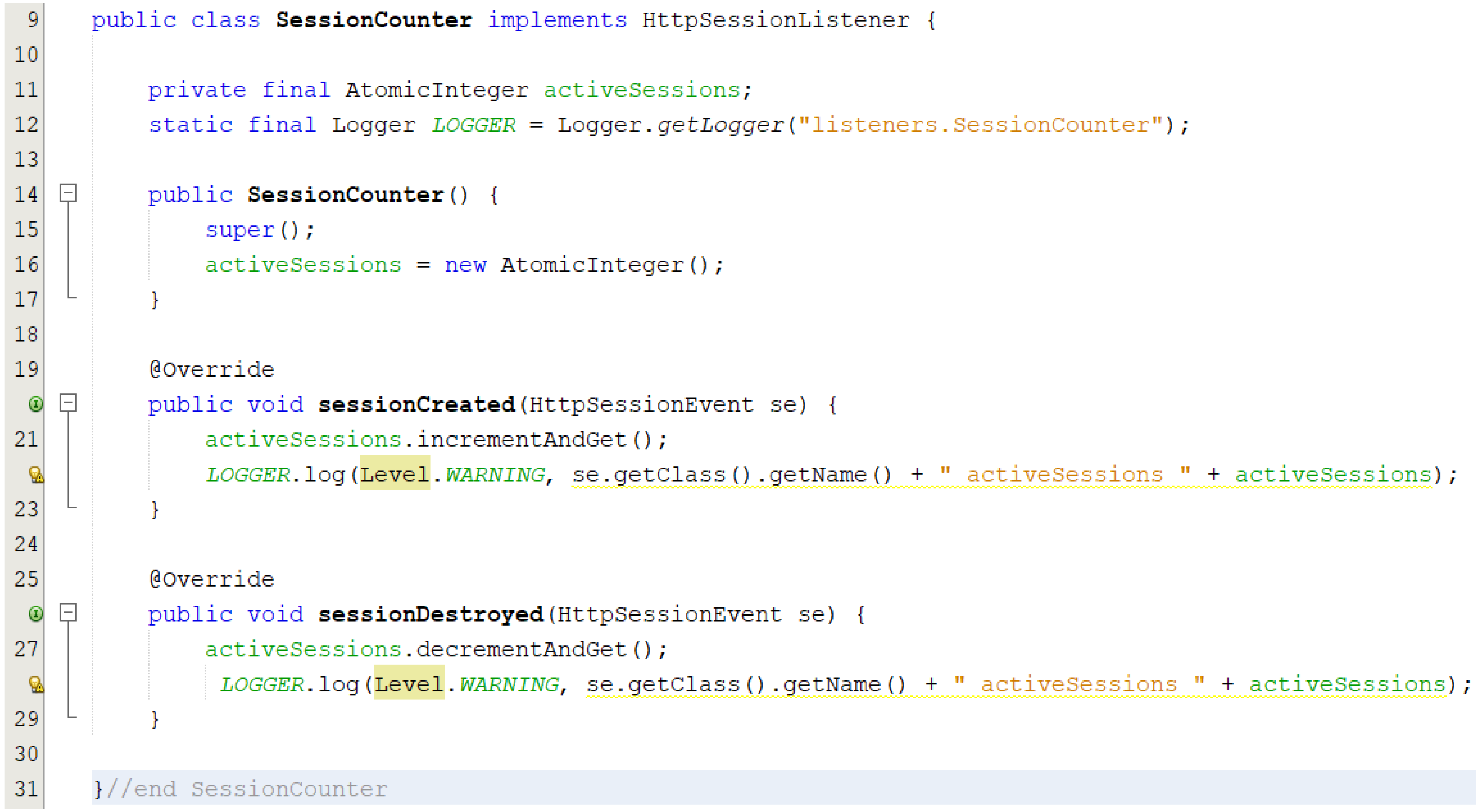
Step One – Create a servlet to create/destroy a session as appropriate.



```java
@WebServlet(urlPatterns = {"/HandleSession"})
public class HandleSession extends HttpServlet {

    protected void processRequest(HttpServletRequest request, HttpServletResponse response)
            throws ServletException, IOException {
        response.setContentType("text/html;charset=UTF-8");
        try (PrintWriter out = response.getWriter()) {
          HttpSession session = request.getSession(false);

          if (session == null) {
              out.println("Session does not exist, so create it");
              session = request.getSession();
              out.println("<br>Session created at " + new Date(session.getCreationTime()));
              out.println("<br>session id " + session.getId());
          }//end if
          else {
              out.println("session exists so invalidate it");
              session.invalidate();
          }//end else

        }//end try
    }//end processRequest
```

Example 2: Counting Sessions

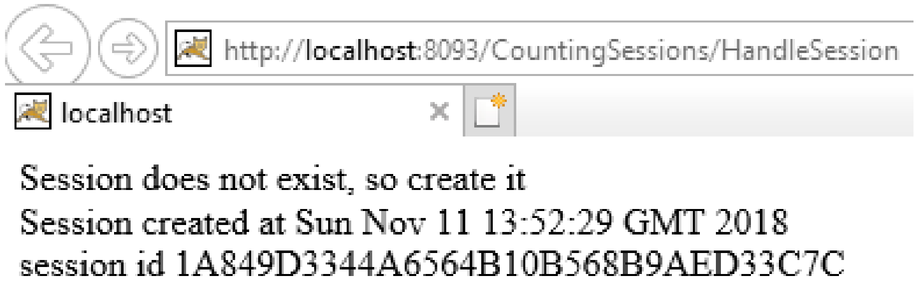
Step Two – Create a listener (listener class registered in web.xml).



```java
public class SessionCounter implements HttpSessionListener {

    private final AtomicInteger activeSessions;
    static final Logger LOGGER = Logger.getLogger("listeners.SessionCounter");

    public SessionCounter() {
        super();
        activeSessions = new AtomicInteger();
    }

    @Override
    public void sessionCreated(HttpSessionEvent se) {
        activeSessions.incrementAndGet();
        LOGGER.log(Level.WARNING, se.getClass().getName() + " activeSessions " + activeSessions);
    }

    @Override
    public void sessionDestroyed(HttpSessionEvent se) {
        activeSessions.decrementAndGet();
         LOGGER.log(Level.WARNING, se.getClass().getName() + " activeSessions " + activeSessions);
    }

}//end SessionCounter
```

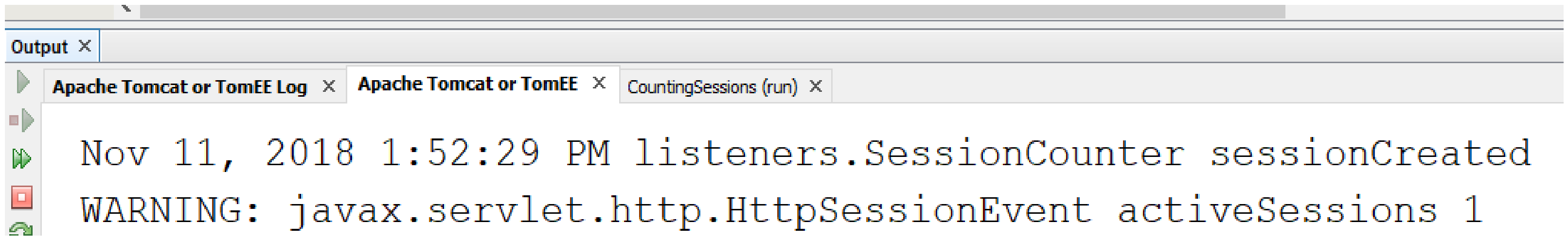
Example 2: Counting Sessions

Step Three – Test the app with a number of different browsers:



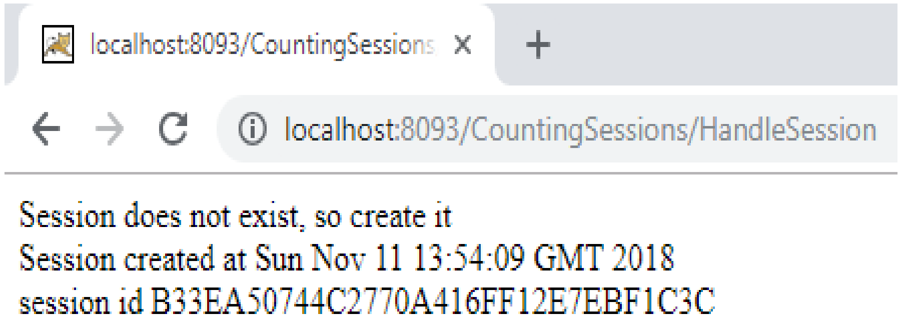
Session does not exist, so create it
Session created at Sun Nov 11 13:52:29 GMT 2018
session id 1A849D3344A6564B10B568B9AED33C7C

Internet Explorer



```
Nov 11, 2018 1:52:29 PM listeners.SessionCounter sessionCreated
WARNING: javax.servlet.http.HttpSessionEvent activeSessions 1
```

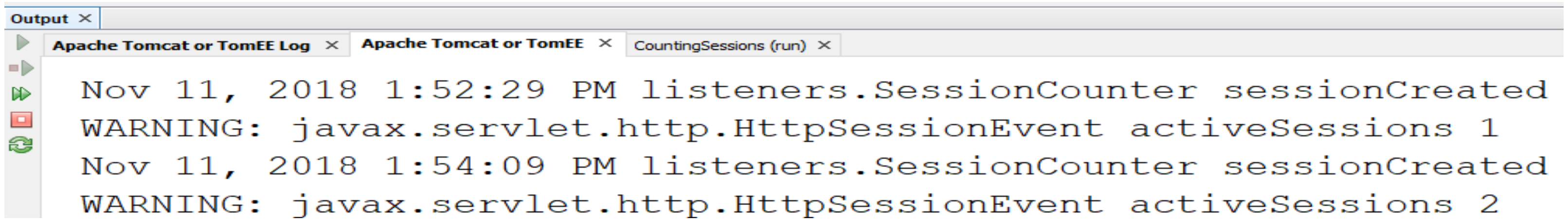
Example 2: Counting Sessions



localhost:8093/CountingSessions
localhost:8093/CountingSessions/HandleSession

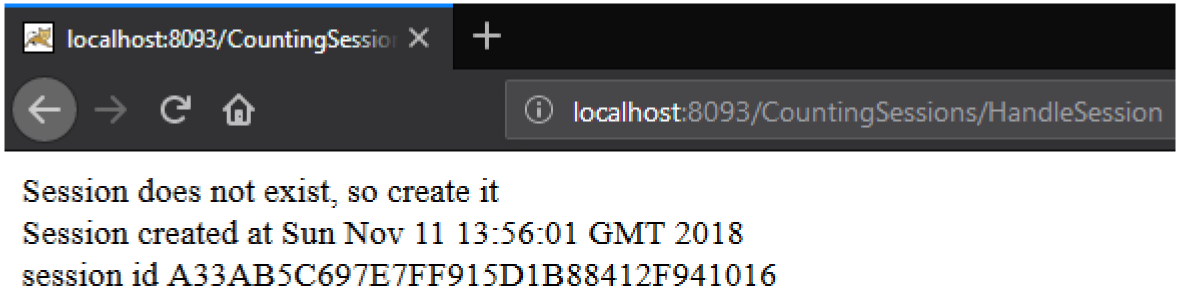
Session does not exist, so create it
Session created at Sun Nov 11 13:54:09 GMT 2018
session id B33EA50744C2770A416FF12E7EBF1C3C

Chrome



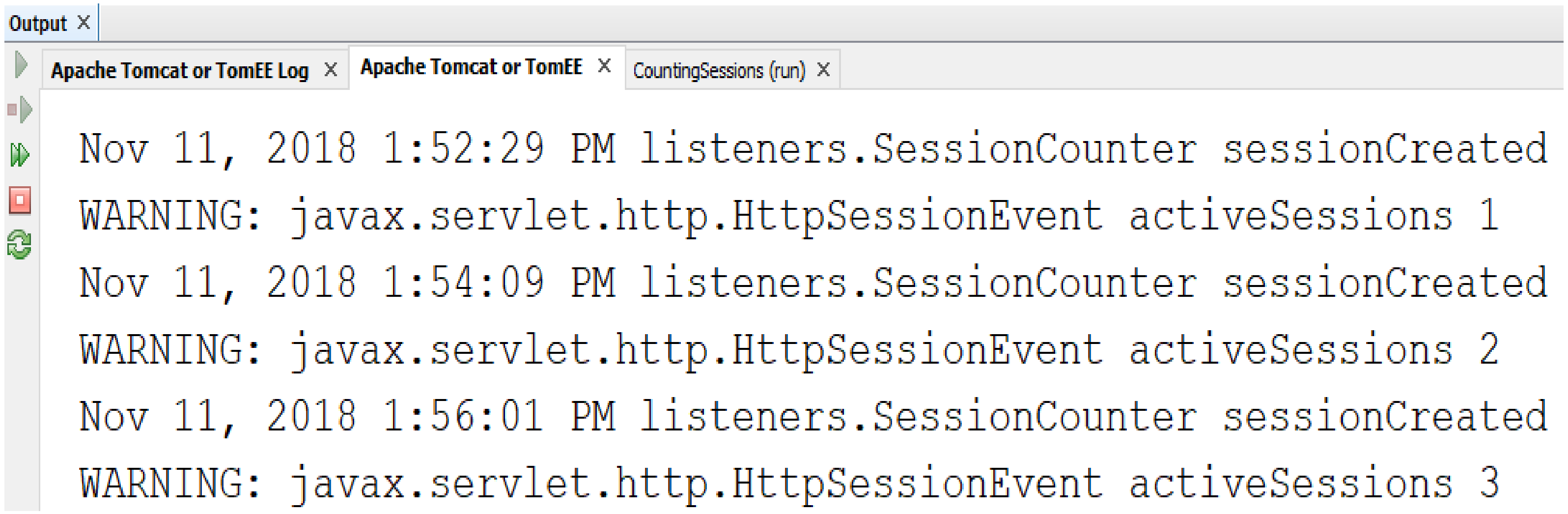
```
Nov 11, 2018 1:52:29 PM listeners.SessionCounter sessionCreated
WARNING: javax.servlet.http.HttpSessionEvent activeSessions 1
Nov 11, 2018 1:54:09 PM listeners.SessionCounter sessionCreated
WARNING: javax.servlet.http.HttpSessionEvent activeSessions 2
```

Example 2: Counting Sessions


localhost:8093/CountingSessio
localhost:8093/CountingSessions/HandleSession

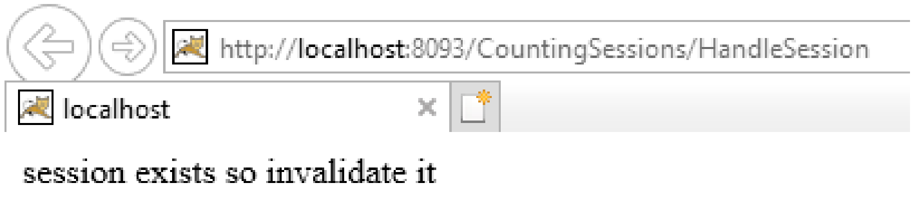
Session does not exist, so create it
Session created at Sun Nov 11 13:56:01 GMT 2018
session id A33AB5C697E7FF915D1B88412F941016

Firefox



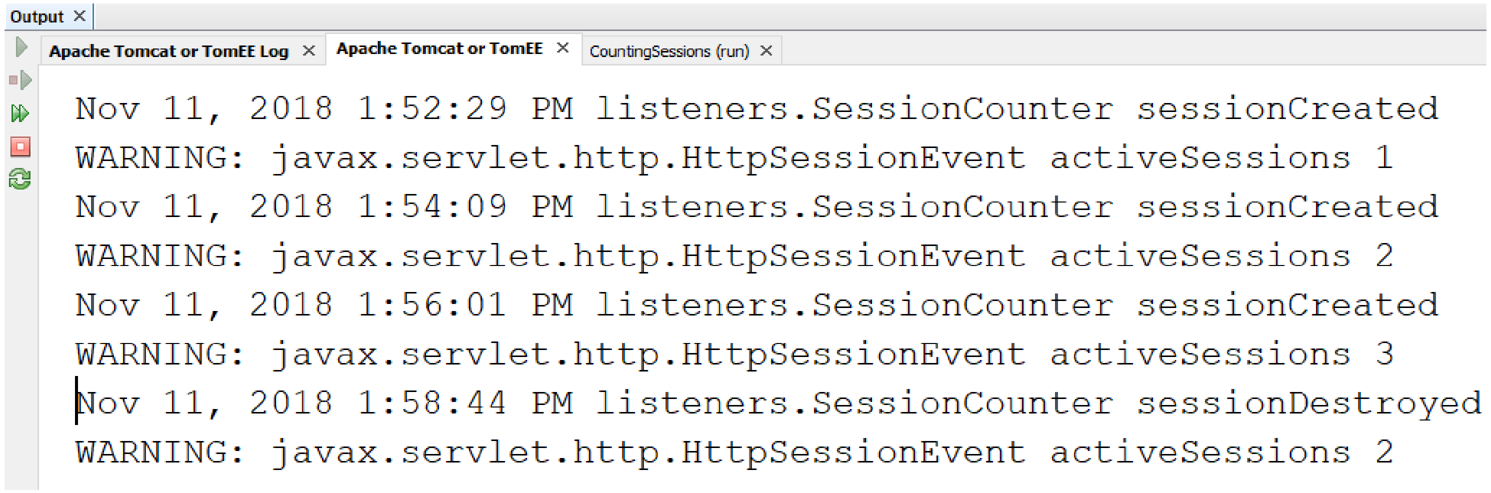
```
Nov 11, 2018 1:52:29 PM listeners.SessionCounter sessionCreated
WARNING: javax.servlet.http.HttpSessionEvent activeSessions 1
Nov 11, 2018 1:54:09 PM listeners.SessionCounter sessionCreated
WARNING: javax.servlet.http.HttpSessionEvent activeSessions 2
Nov 11, 2018 1:56:01 PM listeners.SessionCounter sessionCreated
WARNING: javax.servlet.http.HttpSessionEvent activeSessions 3
```

Example 2: Counting Sessions



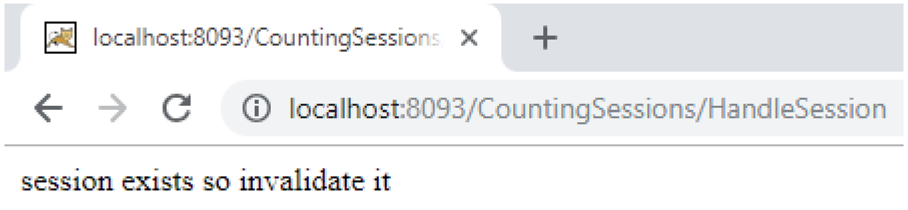
session exists so invalidate it

Internet Explorer



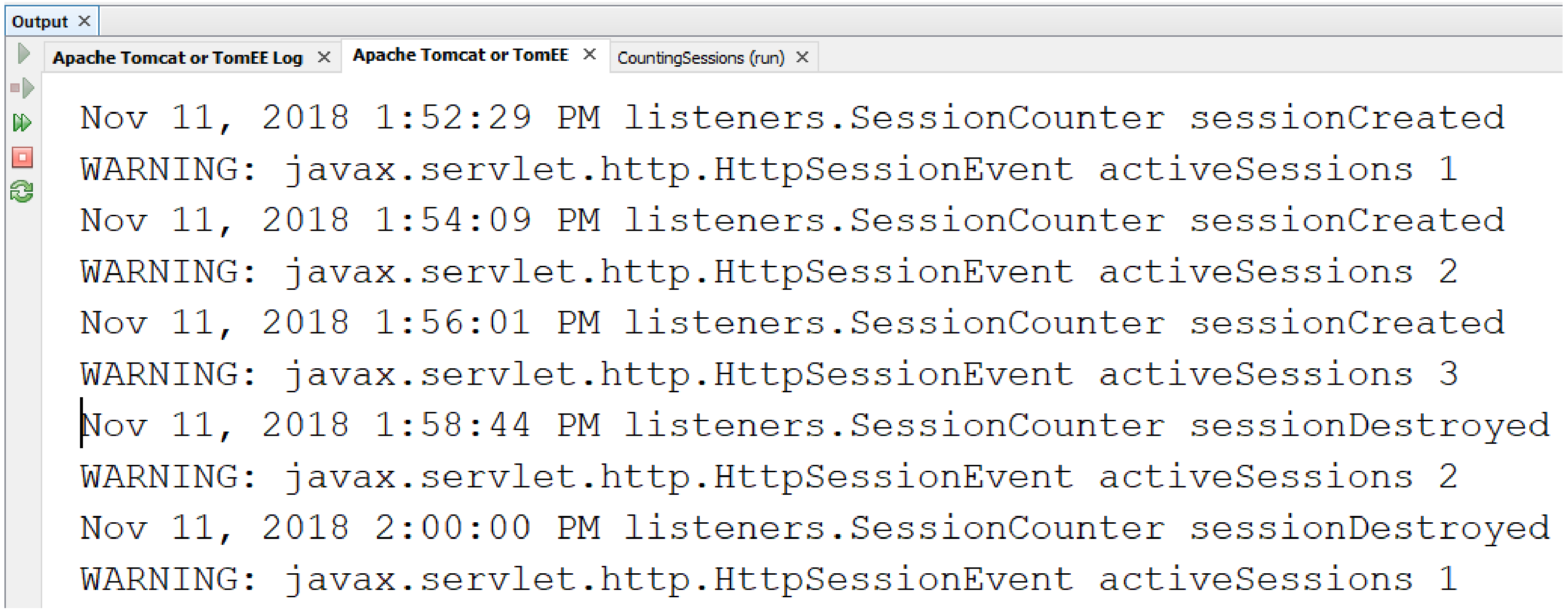
```
Nov 11, 2018 1:52:29 PM listeners.SessionCounter sessionCreated
WARNING: javax.servlet.http.HttpSessionEvent activeSessions 1
Nov 11, 2018 1:54:09 PM listeners.SessionCounter sessionCreated
WARNING: javax.servlet.http.HttpSessionEvent activeSessions 2
Nov 11, 2018 1:56:01 PM listeners.SessionCounter sessionCreated
WARNING: javax.servlet.http.HttpSessionEvent activeSessions 3
Nov 11, 2018 1:58:44 PM listeners.SessionCounter sessionDestroyed
WARNING: javax.servlet.http.HttpSessionEvent activeSessions 2
```

Example 2: Counting Sessions


localhost:8093/CountingSessions
localhost:8093/CountingSessions/HandleSession

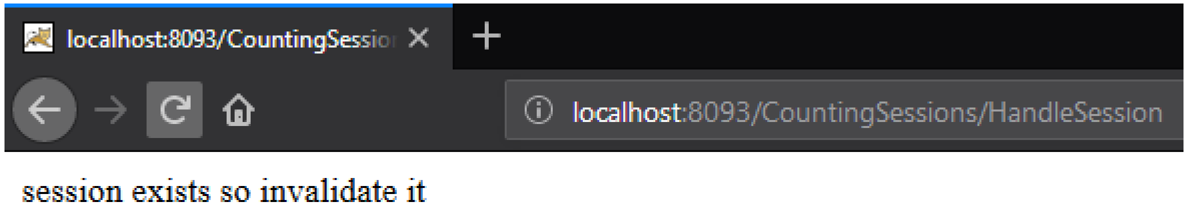
session exists so invalidate it

Chrome



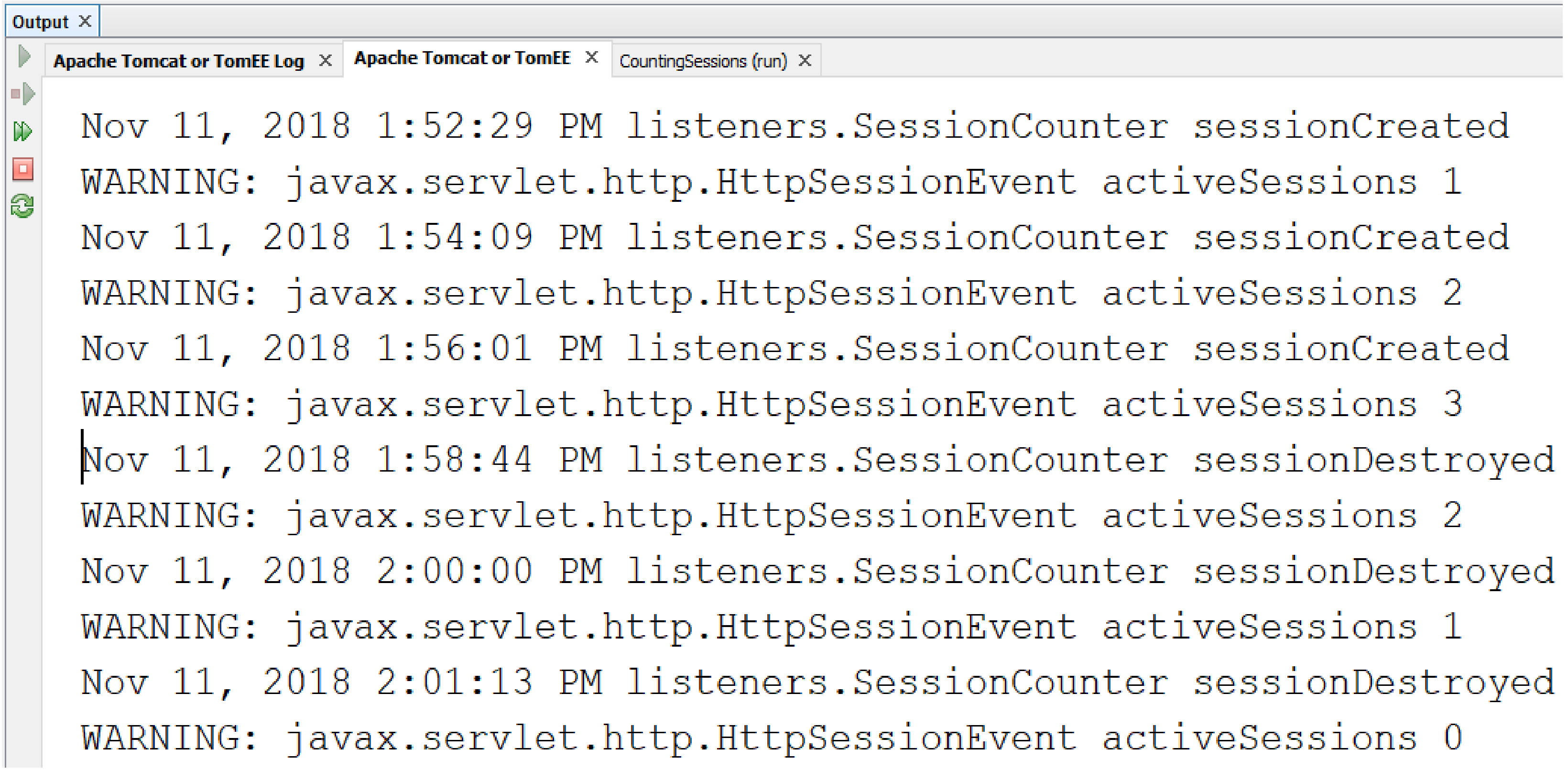
```
Nov 11, 2018 1:52:29 PM listeners.SessionCounter sessionCreated
WARNING: javax.servlet.http.HttpSessionEvent activeSessions 1
Nov 11, 2018 1:54:09 PM listeners.SessionCounter sessionCreated
WARNING: javax.servlet.http.HttpSessionEvent activeSessions 2
Nov 11, 2018 1:56:01 PM listeners.SessionCounter sessionCreated
WARNING: javax.servlet.http.HttpSessionEvent activeSessions 3
Nov 11, 2018 1:58:44 PM listeners.SessionCounter sessionDestroyed
WARNING: javax.servlet.http.HttpSessionEvent activeSessions 2
Nov 11, 2018 2:00:00 PM listeners.SessionCounter sessionDestroyed
WARNING: javax.servlet.http.HttpSessionEvent activeSessions 1
```

Example 2: Counting Sessions


localhost:8093/CountingSessio
localhost:8093/CountingSessions/HandleSession

session exists so invalidate it

Firefox



```
Nov 11, 2018 1:52:29 PM listeners.SessionCounter sessionCreated
WARNING: javax.servlet.http.HttpSessionEvent activeSessions 1
Nov 11, 2018 1:54:09 PM listeners.SessionCounter sessionCreated
WARNING: javax.servlet.http.HttpSessionEvent activeSessions 2
Nov 11, 2018 1:56:01 PM listeners.SessionCounter sessionCreated
WARNING: javax.servlet.http.HttpSessionEvent activeSessions 3
Nov 11, 2018 1:58:44 PM listeners.SessionCounter sessionDestroyed
WARNING: javax.servlet.http.HttpSessionEvent activeSessions 2
Nov 11, 2018 2:00:00 PM listeners.SessionCounter sessionDestroyed
WARNING: javax.servlet.http.HttpSessionEvent activeSessions 1
Nov 11, 2018 2:01:13 PM listeners.SessionCounter sessionDestroyed
WARNING: javax.servlet.http.HttpSessionEvent activeSessions 0
```

References

Murach, J., (2014) *MurachsJava Servlets JSP*, 3rd edn. Mike Murach and Associates, Inc.

https://www.baeldung.com/httpsessionlistener_with_metrics