



TAGLIBS: UTILITY COMPONENTS

Copyright ©2017 Liferay, Inc.

All Rights Reserved.

No material may be reproduced electronically or in print,
distributed, copied, sold, resold, or otherwise exploited
for any commercial purpose without express written
consent of Liferay, Inc.

USING LIFERAY-UTIL: TAGLIBS

- ❖ The `<liferay-util>` taglib is used to pull in other resources into an application or theme.
- ❖ It can also be used to dictate which resources need to be inserted at the bottom or top of the HTML source.
- ❖ A list of the available `<liferay-util>` taglibs can be found at <https://docs.liferay.com/portal/7.0/taglibs/util-taglib/>
- ❖ To use the taglib library, you'll need to make sure the following line is in your JSP:

```
<%@ taglib prefix="liferay-util" uri="http://liferay.com/tld/util" %>
```
- ❖ Each taglib has a list of attributes that can be passed to the tag.
- ❖ Since each of the `<liferay-util>` taglibs is unique, we'll go over each one briefly.

USING LIFERAY-UTIL: BODY-BOTTOM

- ❖ The content placed between the opening and closing of this tag will be moved to the bottom of the body tag.
- ❖ This tag allows you to insert your own HTML markup at the bottom of the body tag for the page.
- ❖ The attribute `outputKey` is the reference key for this content.

EXAMPLE OF LIFERAY-UTIL: BODY-BOTTOM

- ❖ Here is an example of using `<liferay-util:body-bottom>`:

```
<liferay-util:body-bottom outputKey="productMenu">
  <div class="lfr-product-menu-panel sidenav-fixed sidenav-menu-slider" id="
  <%= portletNamespace %>sidenavSliderId">
    <div class="product-menu sidebar sidenav-menu">
      <liferay-portlet:runtime portletName="<%=
      ProductNavigationProductMenuPortletKeys.
      PRODUCT_NAVIGATION_PRODUCT_MENU %>" />
    </div>
  </div>
</liferay-util:body-bottom>
```

USING LIFERAY-UTIL: BODY-TOP

- ❖ The content placed between the opening and closing of this tag will be moved to the top of the body tag.
- ❖ This tag allows you to insert your own HTML markup at the top of the body tag for the page.
- ❖ The attribute `outputKey` is the reference key for this content.
- ❖ Here is an example of using `<liferay-util:body-top>`:

```
<%@ taglib uri="http://liferay.com/tld/util" prefix="liferay-util" %>

<liferay-util:body-top outputKey="topContent">
    <div>
        <h1>I'm at the top of the page!</h1>
    </div>
</liferay-util:body-top>
```

USING LIFERAY-UTIL: BUFFER

- ❖ The content placed between the opening and closing of this tag is saved to the value of the `var` attribute.
- ❖ This allows you to build a piece of markup that can be reused in a JSP. Here is an example of using `<liferay-util:buffer>`:

```
<%@ taglib uri="http://liferay.com/tld/util" prefix="liferay-util" %>
<liferay-util:buffer var="myBuffer">
    <small class="text-capitalize text-muted">
        This is my buffer content
    </small>
</liferay-util:buffer>
<div class="container">
    <h1>Welcome!</h1>
    <%= myBuffer %>
</div>
<div class="container">
    <h1>A Wonderful Title!</h1>
    <%= myBuffer %>
</div>
```

USING LIFERAY-UTIL: DYNAMIC-INCLUDE

- ❖ This taglib allows you to register some content with the `DynamicIncludeRegistry`.
- ❖ Read more about the OSGi Service Registry here: <http://docs.spring.io/osgi/docs/current/reference/html/service-registry.html>
- ❖ The Service Registry makes it easier for modules in the registry to use the content included from the taglib.
- ❖ Here is an example:

```
<%@ taglib uri="http://liferay.com/tld/util" prefix="liferay-util" %>

<liferay-util:dynamic-include key="/path/to/jsp#pre" />

<div>
    <p>And here we have our content</p>
</div>

<liferay-util:dynamic-include key="/path/to/jsp#post" />
```

USING LIFERAY-UTIL: GET-URL

- ❖ This tag scraps the URL provided by the `url` attribute.
- ❖ If a value is provided for the `var` attribute, the content from the screen-cap is scoped to that variable.
- ❖ Otherwise, the content will be displayed where the taglib is used.

EXAMPLE OF LIFERAY-UTIL: GET-URL

❖ Here is an example:

```
<%@ taglib uri="http://liferay.com/tld/util" prefix="liferay-util" %>

<liferay-util:get-url url="https://www.google.com/" />

<!-- Using var attribute -->

<liferay-util:get-url url="https://www.google.com/" var="google" />

<div>
    <h2>We stole <a href="https://www.google.com/">Google</a>, here it is.</h2>

    <div class="google">
        <%= google %>
    </div>
</div>
```

USING LIFERAY-UTIL: HTML-BOTTOM

- ❖ The content placed between the opening and closing of this tag will be moved to the bottom of the `html` tag.
- ❖ This tag allows you to insert your own HTML markup at the bottom of the `body` tag for the page.
- ❖ The attribute `outputKey` is the reference key for this content.

EXAMPLE OF LIFERAY-UTIL: HTML-BOTTOM

- Here is an example of using `<liferay-util:html-bottom>`. Many times, the content passed to this taglib will be JavaScript:

```
<%@ taglib uri="http://liferay.com/tld/util" prefix="liferay-util" %>

<liferay-util:html-bottom outputKey="taglib_alert_user">
    <au:script use="liferay-alert">
        new Liferay.Alert(
            {
                closeable: true,
                message: 'Just saying hello from the &lt;liferay-util:html-bot
                type: 'success'
            }
        ).render(#wrapper);
    </au:script>
</liferay-util:html-bottom>
```

USING LIFERAY-UTIL:HTML-TOP

- ❖ The content placed between the opening and closing of this tag will be moved to the head tag.
- ❖ This tag allows you to insert your own HTML markup at the top of the body tag for the page. The attribute outputKey is the reference key for this content.
- ❖ Here is an example of using `<liferay-util:html-top>`:

```
<%@ taglib uri="http://liferay.com/tld/util" prefix="liferay-util" %>

<liferay-util:html-top>
    <link data-senna-track="permanent" href="/path/to/style.css"
        rel="stylesheet" type="text/css" />
</liferay-util:html-top>
```

USING LIFERAY-UTIL: INCLUDE

- ❖ This tag can be used to include other JSP files in an application.
- ❖ It can help for readability in an application and a separation of concerns for JSP files.
- ❖ The page attribute is required and the value is the path to the JSP or JSPF to be included. The `servletContext` attribute refers to the request context that the included JSP should use.
- ❖ By passing `<%= application %>` to this attribute, the included JSP can use the same request object and other objects that might be set in the prior JSP.
- ❖ Here is an example of using `<liferay-util:include>`:

```
<%@ taglib uri="http://liferay.com/tld/util" prefix="liferay-util" %>
```

```
<liferay-util:include page="/path/to/view.jsp" servletContext="  
<%= application %>" />
```

USING LIFERAY-UTIL: PARAM

- ❖ This tag can be used to add a parameter value to a url.
- ❖ It is useful when used in tandem with `<liferay-util:include>` for accessing new parameter values in another JSP.
- ❖ Here is an example of using `<liferay-util:param>`:

```
<%@ taglib uri="http://liferay.com/tld/util" prefix="liferay-util" %>
```

```
<liferay-util:include page="/path/to/answer.jsp" servletContext="
```

```
<%= application %>">
```

```
    <liferay-util:param name="answer" value="42" />
```

```
</liferay-util:include>
```

- ❖ In `answer.jsp`:

```
<%@ page import="com.liferay.portal.kernel.util.ParamUtil" %>
```

```
<div>
```

```
    <p>The answer to life the universe and everything is <%= ParamUtil.getStri  
    (request, "answer") %></p>
```

```
</div>
```

USING LIFERAY-UTIL: WHITESPACE-REMOVER

- ❖ This taglib is used for removing all whitespace from whatever is included between the opening and closing of the tag.
- ❖ Here is an example of using `<liferay-util:whitespace-remover>`:

```
<%@ taglib uri="http://liferay.com/tld/util" prefix="liferay-util" %>

<liferay-util:whitespace-remover>
  <div class="nput-container">
    <label for="myInput">
      Is the &lt;liferay-util:whitespace-remover&gt; taglib fantastic!
    </label>

    <input class="input" id="myInput" name="myInput" type="checkbox">
  </div>
</liferay-util:whitespace-remover>
```

Notes: