

AbstractHstResponseState

org.hippoecm.hst.core.component.AbstractHstResponseState

Coverage Dependencies Duplications LCOM4 Sources **Violations**

0 4 29 1 0

Filter: Expand:

```
48     return count;
49     }
50 }
51
52     protected boolean isActionResponse;
53     ↑ Visibility Modifier : Variable 'isActionResponse' must be private and have accessor methods.
54     protected boolean isRenderResponse;
55     ↑ Visibility Modifier : Variable 'isRenderResponse' must be private and have accessor methods.
56     protected boolean isResourceResponse;
57     ↑ Visibility Modifier : Variable 'isResourceResponse' must be private and have accessor methods.
58     protected boolean isMimeResponse;
59     ↑ Visibility Modifier : Variable 'isMimeResponse' must be private and have accessor methods.
60     protected boolean isStateAwareResponse;
61     ↑ Visibility Modifier : Variable 'isStateAwareResponse' must be private and have accessor methods.
62     protected Locale defaultLocale;
63     ↑ Visibility Modifier : Variable 'defaultLocale' must be private and have accessor methods.
64     protected boolean flushed;
65     ↑ Visibility Modifier : Variable 'flushed' must be private and have accessor methods.
66     protected ByteArrayOutputStream byteOutputBuffer;
67     ↑ Visibility Modifier : Variable 'byteOutputBuffer' must be private and have accessor methods.
68     protected CharArrayWriterBuffer charOutputBuffer;
69     ↑ Visibility Modifier : Variable 'charOutputBuffer' must be private and have accessor methods.
70     protected ServletOutputStream outputStream;
71     ↑ Visibility Modifier : Variable 'outputStream' must be private and have accessor methods.
72     protected PrintWriter printWriter;
73     ↑ Visibility Modifier : Variable 'printWriter' must be private and have accessor methods.
74     protected Map<String, List<String>> headers;
75     ↑ Visibility Modifier : Variable 'headers' must be private and have accessor methods.
76     protected List<Cookie> cookies;
77     ↑ Visibility Modifier : Variable 'cookies' must be private and have accessor methods.
78     protected List<KeyValue<String, Element>> headElements;
79     ↑ Visibility Modifier : Variable 'headElements' must be private and have accessor methods.
80     protected boolean committed;
81     ↑ Visibility Modifier : Variable 'committed' must be private and have accessor methods.
82     protected boolean hasStatus;
83     ↑ Visibility Modifier : Variable 'hasStatus' must be private and have accessor methods.
84     protected boolean hasError;
85     ↑ Visibility Modifier : Variable 'hasError' must be private and have accessor methods.
86     protected Locale locale;
87     ↑ Visibility Modifier : Variable 'locale' must be private and have accessor methods.
88     protected boolean setContentTypeAfterEncoding;
89     ↑ Visibility Modifier : Variable 'setContentTypeAfterEncoding' must be private and have accessor methods.
90     protected boolean closed;
91     ↑ Visibility Modifier : Variable 'closed' must be private and have accessor methods.
92     protected String characterEncoding;
93     ↑ Visibility Modifier : Variable 'characterEncoding' must be private and have accessor methods.
94     protected int contentLength = -1;
95     ↑ Visibility Modifier : Variable 'contentLength' must be private and have accessor methods.
96     protected String contentType;
97     ↑ Visibility Modifier : Variable 'contentType' must be private and have accessor methods.
98     protected int errorCode;
```

	<p>⬆ Visibility Modifier : Variable 'errorCode' must be private and have accessor methods.</p>
78	<pre>protected String errorMessage;</pre> <p>⬆ Visibility Modifier : Variable 'errorMessage' must be private and have accessor methods.</p>
79	<pre>protected int statusCode;</pre> <p>⬆ Visibility Modifier : Variable 'statusCode' must be private and have accessor methods.</p>
80	
81	<pre>private Object response;</pre>
82	
83	<pre>private String redirectLocation;</pre>
84	
85	<pre>private String forwardPathInfo;</pre>
86	
87	<pre>public AbstractHstResponseState(Object request, Object response) {</pre> <p>⬆ Unused formal parameter : Avoid unused constructor parameters such as 'request'.</p>
88	<pre> this.response = response;</pre>
89	<pre> this.defaultLocale = null;</pre>
90	<pre>}</pre>
91	
92	<pre>protected List<String> getHeaderList(String name, boolean create) {</pre>
225	<pre> failIfCommitted();</pre>
226	<pre> closed = true;</pre>
227	<pre> committed = true;</pre>
228	
229	<pre> if (isMimeResponse) {</pre> <p>⬆ Empty If Stmt : Avoid empty if statements</p>
	<p>⬆ Dodgy - Useless control flow : Useless control flow in org.hippocm.hst.core.component.AbstractHstResponseState.sendRedirect(String)</p>
230	<pre> }</pre>
231	<pre> this.redirectLocation = redirectLocation;</pre>
232	<pre> }</pre>
233	<pre>}</pre>
234	
507	<pre>public void setContentType(String type) {</pre>
508	<pre> if (isMimeResponse && !committed) {</pre>
509	<pre> contentType = type;</pre>
510	<pre> setContentTypeAfterEncoding = false;</pre>
511	<pre> if (printWriter == null) {</pre> <p>⬆ Empty If Stmt : Avoid empty if statements</p>
	<p>⬆ Dodgy - Useless control flow : Useless control flow in org.hippocm.hst.core.component.AbstractHstResponseState.setContentType(String)</p>
512	<pre> // TODO: parse possible encoding for better return value from getCharacterEncoding()</pre>
513	<pre> }</pre>
514	<pre> }</pre>
515	<pre>}</pre>
516	
604	<pre> statusCode = 0;</pre>
605	<pre> redirectLocation = null;</pre>
606	<pre>}</pre>
607	
608	<pre>public void flush() throws IOException {</pre> <p>⬆ Cyclomatic Complexity : Cyclomatic Complexity is 30 (max allowed is 10).</p> <p>⬆ Ncss Method Count : The method flush() has an NCSS line count of 61</p>
609	<pre> if (flushed) {</pre>
610	<pre> //throw new IllegalStateException("Already flushed");</pre>
611	<pre> // Just ignore...</pre>
612	<pre> return;</pre>
613	<pre> }</pre>
694	<pre> outputStream.close();</pre>
695	<pre> outputStream = null;</pre>
696	<pre> byteOutputBuffer = null;</pre>

```
697     } else if (printWriter != null) {
698         if (!closed) {
699             printWriter.flush();
700             if (charOutputBuffer.getCount() > 0) {
701                 getResponseWriter().write(charOutputBuffer.getBuffer(), 0, charOutputBuffer.getCount());
702             }
703             printWriter.close();

```

⚡ **Collapsible If Statements** : These nested if statements could be combined