

Nest.php

```
1. <?php
2.
3. namespace dokuwiki\Parsing\Handler;
4.
5. /**
6.  * Generic call writer class to handle nesting of rendering
7.  * instructions
8.  * within a render instruction. Also see nest() method of renderer
9.  * base class
10. */
11. class Nest extends AbstractRewriter
12. {
13.     protected $closingInstruction;
14.
15.     /**
16.      * @inheritDoc
17.      *
18.      * @param CallWriterInterface $CallWriter      the parser's
19.      * current call writer, i.e. the one above us in the chain
20.      * @param string             $close           closing instruction
21.      * name, this is required to properly terminate the
22.      * syntax mode if the
23.      * document ends without a closing pattern
24.      */
25.     public function __construct(CallWriterInterface $CallWriter,
26.     $close = "nest_close")
27.     {
28.         parent::__construct($CallWriter);
29.         $this->closingInstruction = $close;
30.     }
31.
32.     /**
33.      * @inheritDoc */
34.     public function writeCall($call)
35.     {
36.         $this->calls[] = $call;
37.     }
38.
39.     /**
40.      * @inheritDoc */
41.     public function finalise()
42.     {
```

```
43.         $last_call = end($this->calls);
44.         $this->writeCall(array($this->closingInstruction, array(),
45.             $last_call[2]));
46.     }
47.     $this->process();
48.     $this->callWriter->finalise();
49.     unset($this->callWriter);
50. }
51. /**
52.  * @inheritDoc
53.  */
54. public function process()
55. {
56.     // merge consecutive cdata
57.     $unmerged_calls = $this->calls;
58.     $this->calls = array();
59.
60.     foreach ($unmerged_calls as $call) $this->addCall($call);
61.
62.     $first_call = reset($this->calls);
63.     $this->callWriter->writeCall(array("nest",
64.         array($this->calls), $first_call[2]));
65. }
66. /**
67.  * @param array $call
68.  */
69. protected function addCall($call)
70. {
71.     $key = count($this->calls);
72.     if ($key and ($call[0] == 'cdata') and
73.         ($this->calls[$key-1][0] == 'cdata')) {
74.         $this->calls[$key-1][1][0] .= $call[1][0];
75.     } elseif ($call[0] == 'eol') {
76.         // do nothing (eol shouldn't be allowed, to counter
77.         // preformatted fix in #1652 & #1699)
78.     } else {
79.         $this->calls[] = $call;
80.     }
81. }
82. }
```

From:
<https://wwoss.ru/> - **worldwide open-source software**



Permanent link:
<https://wwoss.ru/doku.php?id=wiki:xref:dokumiki:inc:parsing:handler:nest.php>

Last update: **2025/01/16 22:01**