

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 = $close;
30.     }
31.
32.     /** @inheritdoc */
33.     public function writeCall($call)
34.     {
35.         $this->calls[] = $call;
36.     }
37.
38.     /** @inheritdoc */
39.     public function writeCalls($calls)
40.     {
41.         $this->calls = array_merge($this->calls, $calls);
42.     }
43.
44.     /** @inheritdoc */
45.     public function finalise()
46.     {
47.         // ...
48.     }
49. }
```

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

From:

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

Permanent link:

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

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

