

JLiteralNull.java

```
1  // Copyright 2013 Bill Campbell, Swami Iyer and Bahar Akbal-Delibas
2
3  package jminusminus;
4
5  import static jminusminus.CLConstants.*;
6
7  /**
8   * The AST node for the null literal.
9   */
10
11 class JLiteralNull extends JExpression {
12
13     /**
14      * Construct an AST node for the null literal given its line number.
15      *
16      * @param line
17      *         line in which the literal occurs in the source file.
18      */
19
20     public JLiteralNull(int line) {
21         super(line);
22     }
23
24     /**
25      * Analyzing the null literal is trivial.
26      *
27      * @param context
28      *         context in which names are resolved (ignored here).
29      * @return the analyzed (and possibly rewritten) AST subtree.
30      */
31
32     public JExpression analyze(Context context) {
33         type = Type.NULLTYPE;
34         return this;
35     }
36
37     /**
38      * Generating code for a null literal means generating code to push it onto
39      * the stack.
40      *
41      * @param output
42      *         the code emitter (basically an abstraction for producing the
43      *         .class file).
44      */
45
46     public void codegen(CLEmitter output) {
47         output.addNoArgInstruction(ACONST_NULL);
48     }
49
50     /**
51      * @inheritDoc
52      */
53
54     public void writeToStdOut(PrettyPrinter p) {
55         p.printf("<JLiteralNull line=\"%d\" type=\"%s\"/>\n", line(),
56                 ((type == null) ? "" : type.toString()));
57     }
58
59 }
60
```

Assignment Project Exam Help

<https://powcoder.com>

Add WeChat powcoder