

JavaScript is disabled on your browser.

- [Prev Class](#)
- [Next Class](#)

- [Frames](#)
- [No Frames](#)

- [All Classes](#)

- [Summary:](#)
- [Nested |](#)
- [Field |](#)
- [Constr |](#)
- [Method](#)

- [Detail:](#)
- [Field |](#)
- [Constr |](#)
- [Method](#)

jminusminus

Class JLiteralNull

- [java.lang.Object](#)
- [jminusminus.AST](#)
- [jminusminus.JStatement](#)
- [jminusminus.JExpression](#)
- [jminusminus.JLiteralNull](#)

.

Add WeChat powcoder

```
class JLiteralNull
extends JExpression
```

The AST node for the null literal.

- **Field Summary**
- **Fields inherited from class jminusminus.JExpression**
[isStatementExpression](#), [type](#)
- **Fields inherited from class jminusminus.JAST**
[compilationUnit](#), [line](#)
- **Constructor Summary**

Constructors

Constructor and Description

```
JLiteralNull(int line)
```

Construct an AST node for the null literal given its line number.

- **Method Summary**

Methods	
Modifier and Type	Method and Description
JExpression	analyze (Context context) Analyzing the null literal is trivial.
void	codegen (CLEmitter output) Generating code for a null literal means generating code to push it onto the stack.
void	writeToStdOut (PrettyPrinter p) Write the information pertaining to this AST to STDOUT.

- **Methods inherited from class jminusminus.JExpression**

codegen, isStatementExpression, type

- **Methods inherited from class jminusminus.JAST**

line, partialCodegen

- **Methods inherited from class java.lang.Object**

clone, equals, finalize, getClass, hashCode, notify, notifyAll, toString, wait, wait, wait

- **Constructor Detail**

- **JLiteralNull**

public JLiteralNull(int line)

Construct an AST node for the null literal given its line number.

Parameters:

line - line in which the literal occurs in the source file.

- **Method Detail**

- **analyze**

public JExpression analyze(Context context)

Analyzing the null literal is trivial.

Specified by:

analyze in class JExpression

Parameters:

context - context in which names are resolved (ignored here).

Returns:

the analyzed (and possibly rewritten) AST subtree.

- **codegen**

public void codegen(CLEmitter output)

Generating code for a null literal means generating code to push it onto the stack.

Specified by:

codegen in class JAST

Assignment Project Exam Help

<https://powcoder.com>

Add WeChat powcoder

Parameters:

output - the code emitter (basically an abstraction for producing the .class file).

- **writeToStdOut**

```
public void writeToStdOut(PrettyPrinter p)
```

Description copied from class: JAST

Write the information pertaining to this AST to STDOUT.

Specified by:

`writeToStdOut` in class `JAST`

Parameters:

p - for pretty printing with indentation.

- [Prev Class](#)
- [Next Class](#)

- [Frames](#)
- [No Frames](#)

- [All Classes](#)

- [Summary:](#)
- [Nested |](#)
- [Field |](#)
- [Constr |](#)
- [Method](#)

- [Detail:](#)
- [Field |](#)
- [Constr |](#)
- [Method](#)

Assignment Project Exam Help

<https://powcoder.com>

Add WeChat powcoder