

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 JFieldDeclaration

- [java.lang.Object](#)
 - [jminusminus.AST](#)
 - [jminusminus.JFieldDeclaration](#)

- [All Implemented Interfaces:](#)
[JMember](#)

Assignment Project Exam Help
<https://powcoder.com>

Add WeChat powcoder

```
class JFieldDeclaration
extends JAST
implements JMember
```

The AST node for a field declaration.

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

Constructors

Constructor and Description

```
JFieldDeclaration(int line, ArrayList<String> mods,
ArrayList<JVariableDeclarator> decls)
```

Construct an AST node for a field declaration given the line number, modifiers, and the variable declarators.

- **Method Summary**

Methods

Modifier and Type	Method and Description
<code>JFieldDeclaration</code>	<code>analyze</code> (<code>Context</code> context) Analysis of field declaration involves rewriting initializations (if any) as assignment statements.
<code>void</code>	<code>codegen</code> (<code>CLEmitter</code> output) Code generation for field declaration involves generate field the header.
<code>void</code>	<code>codegenInitializations</code> (<code>CLEmitter</code> output) Generate code for any field initializations (now rewritten as assignment statements).
<code>ArrayList<String></code>	<code>mods</code> () Return the list of modifiers.
<code>void</code>	<code>preAnalyze</code> (<code>Context</code> context, <code>CLEmitter</code> partial) Declare fields in the parent's (partial) class.
<code>void</code>	<code>writeToStdOut</code> (<code>PrettyPrinter</code> p) Write the information pertaining to this AST to STDOUT.

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

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

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

- **Constructor Detail**

- **`JFieldDeclaration`**

```
public JFieldDeclaration(int line,
                        ArrayList<String> mods,
                        ArrayList<JVariableDeclarator> decls)
```

Construct an AST node for a field declaration given the line number, modifiers, and the variable declarators.

Parameters:

`line` - line in which the variable declaration occurs in the source file.
`mods` - field modifiers.
`decls` - variable declarators.

- **Method Detail**

- **`mods`**

```
public ArrayList<String> mods()
```

Return the list of modifiers.

Returns:

list of modifiers.

- **`preAnalyze`**

```
public void preAnalyze(Context context,
                      CLEmitter partial)
```

Assignment Project Exam Help

<https://powcoder.com>

Add WeChat powcoder

Declare fields in the parent's (partial) class.

Specified by:

`preAnalyze` in interface `JMember`

Parameters:

`context` - the parent (class) context.

`partial` - the code emitter (basically an abstraction for producing the partial class).

- **analyze**

```
public JFieldDeclaration analyze(Context context)
```

Analysis of field declaration involves rewriting initializations (if any) as assignment statements.

Specified by:

`analyze` in class `JAST`

Parameters:

`context` - context in which names are resolved.

Returns:

the analyzed `JFieldDeclaration` subtree.

- **codegenInitializations**

```
public void codegenInitializations(CLEmitter output)
```

Generate code for any field initializations (now rewritten as assignment statements).

Parameters:

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

- **codegen**

```
public void codegen(CLEmitter output)
```

Code generation for field declaration involves generate field the header.

Specified by:

`codegen` in class `JAST`

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.

Assignment Project Exam Help

<https://powcoder.com>

Add WeChat powcoder

- **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