

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 JConstructorDeclaration

- [java.lang.Object](#)
  - [jminusminus.AST](#)
    - [jminusminus.JMethodDeclaration](#)
      - [jminusminus.JConstructorDeclaration](#)
- All Implemented Interfaces:  
[JMember](#)

<https://powcoder.com>

Add WeChat powcoder

```
class JConstructorDeclaration
extends JMethodDeclaration
implements JMember
```

The AST node for a constructor declaration. A constructor looks very much like a method.

- **Field Summary**

Fields	
Modifier and Type	Field and Description
(package private) <a href="#">JClassDeclaration</a>	<b>definingClass</b> Defining class

- **Fields inherited from class [jminusminus.JMethodDeclaration](#)**  
[body](#), [context](#), [descriptor](#), [isAbstract](#), [isPrivate](#), [isStatic](#), [mods](#), [name](#), [params](#), [returnType](#)
- **Fields inherited from class [jminusminus.JAST](#)**  
[compilationUnit](#), [line](#)

- **Constructor Summary**

Constructors  
**Constructor and Description**

```
JConstructorDeclaration(int line, ArrayList<String> mods,
String name, ArrayList<JFormalParameter> params,
JBlock body)
```

Construct an AST node for a constructor declaration given the line number, modifiers, constructor name, formal parameters, and the constructor body.

- **Method Summary**

Methods

**Modifier and  
Type**

**Method and Description**

JAST	<pre><b>analyze</b>(Context context)</pre> <p>Analysis for a constructor declaration is very much like that for a method declaration.</p>
------	---

void	<pre><b>codegen</b>(CLEmitter output)</pre> <p>Generate code for the constructor declaration.</p>
------	---

void	<pre><b>partialCodegen</b>(Context context, CLEmitter partial)</pre> <p>Add this constructor declaration to the partial class.</p>
------	--

void	<pre><b>preAnalyze</b>(Context context, CLEmitter partial)</pre> <p>Declare this constructor in the parent (class) context.</p>
------	---

void	<pre><b>writeToStdOut</b>(PrettyPrinter p)</pre> <p>Write the information pertaining to this AST to STDOUT.</p>
------	---

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

line

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

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

- **Field Detail**

- **definingClass**

JClassDeclaration definingClass

Defining class

- **Constructor Detail**

- **JConstructorDeclaration**

```
publicJConstructorDeclaration(intline,
                             ArrayList<String>mods,
                             Stringname,
                             ArrayList<JFormalParameter>params,
                             JBlockbody)
```

Assignment Project Exam Help

<https://powcoder.com>

Add WeChat powcoder

Construct an AST node for a constructor declaration given the line number, modifiers, constructor name, formal parameters, and the constructor body.

**Parameters:**

line - line in which the constructor declaration occurs in the source file.  
mods - modifiers.  
name - constructor name.  
params - the formal parameters.  
body - constructor body.

- **Method Detail**

- **preAnalyze**

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

Declare this constructor in the parent (class) context.

**Specified by:**

preAnalyze in interface JMember

**Overrides:**

preAnalyze in class JMethodDeclaration

**Parameters:**

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

- **analyze**

```
public AST analyze(Context context)
```

Analysis for a constructor declaration is very much like that for a method declaration.

**Overrides:**

analyze in class JMethodDeclaration

**Parameters:**

context - context in which names are resolved.

**Returns:**

the analyzed (and possibly rewritten) AST subtree.

- **partialCodegen**

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

Add this constructor declaration to the partial class.

**Overrides:**

partialCodegen in class JMethodDeclaration

**Parameters:**

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

- **codegen**

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

Generate code for the constructor declaration.

**Overrides:**

codegen in class JMethodDeclaration

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.

**Overrides:**

`writeToStdOut` in class `JMethodDeclaration`

**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