| | [**Overview**](http://docs.google.com/overview-summary.html) | [**Package**](http://docs.google.com/package-summary.html) | **Class** | [**Use**](http://docs.google.com/class-use/ScriptEngineFactory.html) | [**Tree**](http://docs.google.com/package-tree.html) | [**Deprecated**](http://docs.google.com/deprecated-list.html) | [**Index**](http://docs.google.com/index-files/index-1.html) | [**Help**](http://docs.google.com/help-doc.html) | | --- | --- | --- | --- | --- | --- | --- | --- | | | ***Java™ Platform***  ***Standard Ed. 6*** |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| [**PREV CLASS**](http://docs.google.com/javax/script/ScriptEngine.html)   [**NEXT CLASS**](http://docs.google.com/javax/script/ScriptEngineManager.html) | [**FRAMES**](http://docs.google.com/index.html?javax/script/ScriptEngineFactory.html)    [**NO FRAMES**](http://docs.google.com/ScriptEngineFactory.html)     [**All Classes**](http://docs.google.com/allclasses-noframe.html) |
| SUMMARY: NESTED | FIELD | CONSTR | [METHOD](#3znysh7) | DETAIL: FIELD | CONSTR | [METHOD](#2et92p0) |

## **javax.script**

Interface ScriptEngineFactory

public interface **ScriptEngineFactory**

ScriptEngineFactory is used to describe and instantiate ScriptEngines.

Each class implementing ScriptEngine has a corresponding factory that exposes metadata describing the engine class.

The ScriptEngineManager uses the service provider mechanism described in the *Jar File Specification* to obtain instances of all ScriptEngineFactories available in the current ClassLoader.

**Since:** 1.6

| **Method Summary** | |
| --- | --- |
| [String](http://docs.google.com/java/lang/String.html) | [**getEngineName**](http://docs.google.com/javax/script/ScriptEngineFactory.html#getEngineName())()            Returns the full name of the ScriptEngine. |
| [String](http://docs.google.com/java/lang/String.html) | [**getEngineVersion**](http://docs.google.com/javax/script/ScriptEngineFactory.html#getEngineVersion())()            Returns the version of the ScriptEngine. |
| [List](http://docs.google.com/java/util/List.html)<[String](http://docs.google.com/java/lang/String.html)> | [**getExtensions**](http://docs.google.com/javax/script/ScriptEngineFactory.html#getExtensions())()            Returns an immutable list of filename extensions, which generally identify scripts written in the language supported by this ScriptEngine. |
| [String](http://docs.google.com/java/lang/String.html) | [**getLanguageName**](http://docs.google.com/javax/script/ScriptEngineFactory.html#getLanguageName())()            Returns the name of the scripting langauge supported by this ScriptEngine. |
| [String](http://docs.google.com/java/lang/String.html) | [**getLanguageVersion**](http://docs.google.com/javax/script/ScriptEngineFactory.html#getLanguageVersion())()            Returns the version of the scripting language supported by this ScriptEngine. |
| [String](http://docs.google.com/java/lang/String.html) | [**getMethodCallSyntax**](http://docs.google.com/javax/script/ScriptEngineFactory.html#getMethodCallSyntax(java.lang.String,%20java.lang.String,%20java.lang.String...))([String](http://docs.google.com/java/lang/String.html) obj, [String](http://docs.google.com/java/lang/String.html) m, [String](http://docs.google.com/java/lang/String.html)... args)            Returns a String which can be used to invoke a method of a Java object using the syntax of the supported scripting language. |
| [List](http://docs.google.com/java/util/List.html)<[String](http://docs.google.com/java/lang/String.html)> | [**getMimeTypes**](http://docs.google.com/javax/script/ScriptEngineFactory.html#getMimeTypes())()            Returns an immutable list of mimetypes, associated with scripts that can be executed by the engine. |
| [List](http://docs.google.com/java/util/List.html)<[String](http://docs.google.com/java/lang/String.html)> | [**getNames**](http://docs.google.com/javax/script/ScriptEngineFactory.html#getNames())()            Returns an immutable list of short names for the ScriptEngine, which may be used to identify the ScriptEngine by the ScriptEngineManager. |
| [String](http://docs.google.com/java/lang/String.html) | [**getOutputStatement**](http://docs.google.com/javax/script/ScriptEngineFactory.html#getOutputStatement(java.lang.String))([String](http://docs.google.com/java/lang/String.html) toDisplay)            Returns a String that can be used as a statement to display the specified String using the syntax of the supported scripting language. |
| [Object](http://docs.google.com/java/lang/Object.html) | [**getParameter**](http://docs.google.com/javax/script/ScriptEngineFactory.html#getParameter(java.lang.String))([String](http://docs.google.com/java/lang/String.html) key)            Returns the value of an attribute whose meaning may be implementation-specific. |
| [String](http://docs.google.com/java/lang/String.html) | [**getProgram**](http://docs.google.com/javax/script/ScriptEngineFactory.html#getProgram(java.lang.String...))([String](http://docs.google.com/java/lang/String.html)... statements)            Returns A valid scripting language executable progam with given statements. |
| [ScriptEngine](http://docs.google.com/javax/script/ScriptEngine.html) | [**getScriptEngine**](http://docs.google.com/javax/script/ScriptEngineFactory.html#getScriptEngine())()            Returns an instance of the ScriptEngine associated with this ScriptEngineFactory. |

| **Method Detail** |
| --- |

### getEngineName

[String](http://docs.google.com/java/lang/String.html) **getEngineName**()

Returns the full name of the ScriptEngine. For instance an implementation based on the Mozilla Rhino Javascript engine might return *Rhino Mozilla Javascript Engine*.

**Returns:**The name of the engine implementation.

### getEngineVersion

[String](http://docs.google.com/java/lang/String.html) **getEngineVersion**()

Returns the version of the ScriptEngine.

**Returns:**The ScriptEngine implementation version.

### getExtensions

[List](http://docs.google.com/java/util/List.html)<[String](http://docs.google.com/java/lang/String.html)> **getExtensions**()

Returns an immutable list of filename extensions, which generally identify scripts written in the language supported by this ScriptEngine. The array is used by the ScriptEngineManager to implement its getEngineByExtension method.

**Returns:**The list of extensions.

### getMimeTypes

[List](http://docs.google.com/java/util/List.html)<[String](http://docs.google.com/java/lang/String.html)> **getMimeTypes**()

Returns an immutable list of mimetypes, associated with scripts that can be executed by the engine. The list is used by the ScriptEngineManager class to implement its getEngineByMimetype method.

**Returns:**The list of mime types.

### getNames

[List](http://docs.google.com/java/util/List.html)<[String](http://docs.google.com/java/lang/String.html)> **getNames**()

Returns an immutable list of short names for the ScriptEngine, which may be used to identify the ScriptEngine by the ScriptEngineManager. For instance, an implementation based on the Mozilla Rhino Javascript engine might return list containing {"javascript", "rhino"}.

### getLanguageName

[String](http://docs.google.com/java/lang/String.html) **getLanguageName**()

Returns the name of the scripting langauge supported by this ScriptEngine.

**Returns:**The name of the supported language.

### getLanguageVersion

[String](http://docs.google.com/java/lang/String.html) **getLanguageVersion**()

Returns the version of the scripting language supported by this ScriptEngine.

**Returns:**The version of the supported language.

### getParameter

[Object](http://docs.google.com/java/lang/Object.html) **getParameter**([String](http://docs.google.com/java/lang/String.html) key)

Returns the value of an attribute whose meaning may be implementation-specific. Keys for which the value is defined in all implementations are:

* ScriptEngine.ENGINE
* ScriptEngine.ENGINE\_VERSION
* ScriptEngine.NAME
* ScriptEngine.LANGUAGE
* ScriptEngine.LANGUAGE\_VERSION

The values for these keys are the Strings returned by getEngineName, getEngineVersion, getName, getLanguageName and getLanguageVersion respectively.

A reserved key, **THREADING**, whose value describes the behavior of the engine with respect to concurrent execution of scripts and maintenance of state is also defined. These values for the **THREADING** key are:

null - The engine implementation is not thread safe, and cannot be used to execute scripts concurrently on multiple threads."MULTITHREADED" - The engine implementation is internally thread-safe and scripts may execute concurrently although effects of script execution on one thread may be visible to scripts on other threads."THREAD-ISOLATED" - The implementation satisfies the requirements of "MULTITHREADED", and also, the engine maintains independent values for symbols in scripts executing on different threads."STATELESS" - The implementation satisfies the requirements of "THREAD-ISOLATED". In addition, script executions do not alter the mappings in the Bindings which is the engine scope of the ScriptEngine. In particular, the keys in the Bindings and their associated values are the same before and after the execution of the script.

Implementations may define implementation-specific keys.

**Parameters:**key - The name of the parameter **Returns:**The value for the given parameter. Returns null if no value is assigned to the key.

### getMethodCallSyntax

[String](http://docs.google.com/java/lang/String.html) **getMethodCallSyntax**([String](http://docs.google.com/java/lang/String.html) obj,  
 [String](http://docs.google.com/java/lang/String.html) m,  
 [String](http://docs.google.com/java/lang/String.html)... args)

Returns a String which can be used to invoke a method of a Java object using the syntax of the supported scripting language. For instance, an implementaton for a Javascript engine might be;

public String getMethodCallSyntax(String obj,  
 String m, String... args) {  
 String ret = obj;  
 ret += "." + m + "(";  
 for (int i = 0; i < args.length; i++) {  
 ret += args[i];  
 if (i == args.length - 1) {  
 ret += ")";  
 } else {  
 ret += ",";  
 }  
 }  
 return ret;  
 }

**Parameters:**obj - The name representing the object whose method is to be invoked. The name is the one used to create bindings using the put method of ScriptEngine, the put method of an ENGINE\_SCOPE Bindings,or the setAttribute method of ScriptContext. The identifier used in scripts may be a decorated form of the specified one.m - The name of the method to invoke.args - names of the arguments in the method call. **Returns:**The String used to invoke the method in the syntax of the scripting language.

### getOutputStatement

[String](http://docs.google.com/java/lang/String.html) **getOutputStatement**([String](http://docs.google.com/java/lang/String.html) toDisplay)

Returns a String that can be used as a statement to display the specified String using the syntax of the supported scripting language. For instance, the implementaton for a Perl engine might be;

public String getOutputStatement(String toDisplay) {  
 return "print(" + toDisplay + ")";  
 }

**Parameters:**toDisplay - The String to be displayed by the returned statement. **Returns:**The string used to display the String in the syntax of the scripting language.

### getProgram

[String](http://docs.google.com/java/lang/String.html) **getProgram**([String](http://docs.google.com/java/lang/String.html)... statements)

Returns A valid scripting language executable progam with given statements. For instance an implementation for a PHP engine might be:

public String getProgram(String... statements) {  
 $retval = "<?\n";  
 int len = statements.length;  
 for (int i = 0; i < len; i++) {  
 $retval += statements[i] + ";\n";  
 }  
 $retval += "?>";  
  
 }

**Parameters:**statements - The statements to be executed. May be return values of calls to the getMethodCallSyntax and getOutputStatement methods. **Returns:**The Program

### getScriptEngine

[ScriptEngine](http://docs.google.com/javax/script/ScriptEngine.html) **getScriptEngine**()

Returns an instance of the ScriptEngine associated with this ScriptEngineFactory. A new ScriptEngine is generally returned, but implementations may pool, share or reuse engines.

**Returns:**A new ScriptEngine instance.

| | [**Overview**](http://docs.google.com/overview-summary.html) | [**Package**](http://docs.google.com/package-summary.html) | **Class** | [**Use**](http://docs.google.com/class-use/ScriptEngineFactory.html) | [**Tree**](http://docs.google.com/package-tree.html) | [**Deprecated**](http://docs.google.com/deprecated-list.html) | [**Index**](http://docs.google.com/index-files/index-1.html) | [**Help**](http://docs.google.com/help-doc.html) | | --- | --- | --- | --- | --- | --- | --- | --- | | | ***Java™ Platform***  ***Standard Ed. 6*** |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| [**PREV CLASS**](http://docs.google.com/javax/script/ScriptEngine.html)   [**NEXT CLASS**](http://docs.google.com/javax/script/ScriptEngineManager.html) | [**FRAMES**](http://docs.google.com/index.html?javax/script/ScriptEngineFactory.html)    [**NO FRAMES**](http://docs.google.com/ScriptEngineFactory.html)     [**All Classes**](http://docs.google.com/allclasses-noframe.html) |
| SUMMARY: NESTED | FIELD | CONSTR | [METHOD](#3znysh7) | DETAIL: FIELD | CONSTR | [METHOD](#2et92p0) |

[Submit a bug or feature](http://bugs.sun.com/services/bugreport/index.jsp)

For further API reference and developer documentation, see [Java SE Developer Documentation](http://docs.google.com/webnotes/devdocs-vs-specs.html). That documentation contains more detailed, developer-targeted descriptions, with conceptual overviews, definitions of terms, workarounds, and working code examples.

Copyright 2006 Sun Microsystems, Inc. All rights reserved. Use is subject to [license terms](http://docs.google.com/legal/license.html). Also see the [documentation redistribution policy](http://java.sun.com/docs/redist.html).