

JAVA.LANG.RUNTIME CLASS

http://www.tutorialspoint.com/java/lang/java_lang_runtime.htm

Copyright © tutorialspoint.com

Introduction

The **java.lang.Runtime** class allows the application to interface with the environment in which the application is running.

Class declaration

Following is the declaration for **java.lang.Runtime** class:

```
public class Runtime
    extends Object
```

Class methods

S.N.	Method & Description
1	<u>void addShutdownHookThreadhook</u> This method registers a new virtual-machine shutdown hook.
2	<u>int availableProcessors</u> This method returns the number of processors available to the Java virtual machine.
3	<u>Process execStringcommand</u> This method executes the specified string command in a separate process.
4	<u>Process execString[]cmdarray</u> This method executes the specified command and arguments in a separate process.
5	<u>Process execString[]cmdarray, String[]envp</u> This method executes the specified command and arguments in a separate process with the specified environment.
6	<u>Process execString[]cmdarray, String[]envp, Filedir</u> This method executes the specified command and arguments in a separate process with the specified environment and working directory.
7	<u>Process execStringcommand, String[]envp</u> This method executes the specified string command in a separate process with the specified environment.

8

[Process execStringcommand, String\[\]envp, Filedir](#)

This method executes the specified string command in a separate process with the specified environment and working directory.

9

[void exitintstatus](#)

This method terminates the currently running Java virtual machine by initiating its shutdown sequence.

10

[long freeMemory](#)

This method returns the amount of free memory in the Java Virtual Machine.

11

[void gc](#)

This method runs the garbage collector.

12

[static Runtime getRuntime](#)

This method returns the runtime object associated with the current Java application.

13

[void haltintstatus](#)

This method forcibly terminates the currently running Java virtual machine.

14

[void loadStringfilename](#)

This method loads the specified filename as a dynamic library.

15

[void loadLibraryStringlibname](#)

This method loads the dynamic library with the specified library name.

16

[long maxMemory](#)

This method returns the maximum amount of memory that the Java virtual machine will attempt to use.

17

[boolean removeShutdownHookThreadhook](#)

This method de-registers a previously-registered virtual-machine shutdown hook.

18

[void runFinalization](#)

This method runs the finalization methods of any objects pending finalization.

19

[long totalMemory](#)

This method returns the total amount of memory in the Java virtual machine.

20

[`void traceInstructions`](#)*`boolean`*

This method enables/disables tracing of instructions.

21

[`void traceMethodCalls`](#)*`boolean`*

This method enables/disables tracing of method calls.

Methods inherited

This class inherits methods from the following classes:

• [`java.lang.Object`](#)

Loading [MathJax]/jax/output/HTML-CSS/jax.js