

**NAME**

`jrunscript` – run a command–line script shell that supports interactive and batch modes

**SYNOPSIS****Note:**

This tool is **experimental** and unsupported.

**jrunscript** [*options*] [*arguments*]

*options* This represents the **jrunscript** command–line options that can be used. See **Options for the jrunscript Command**.

*arguments*

Arguments, when used, follow immediately after options or the command name. See **Arguments**.

**DESCRIPTION**

The **jrunscript** command is a language–independent command–line script shell. The **jrunscript** command supports both an interactive (read–eval–print) mode and a batch (**-f** option) mode of script execution. By default, JavaScript is the language used, but the **-l** option can be used to specify a different language. By using Java to scripting language communication, the **jrunscript** command supports an exploratory programming style.

If JavaScript is used, then before it evaluates a user defined script, the **jrunscript** command initializes certain built–in functions and objects, which are documented in the API Specification for **jrunscript** JavaScript built–in functions.

**OPTIONS FOR THE JRUNSCRIPT COMMAND**

**-cp** *path* or **-classpath** *path*

Indicates where any class files are that the script needs to access.

**-Dname=***value*

Sets a Java system property.

**-Jflag** Passes *flag* directly to the Java Virtual Machine where the **jrunscript** command is running.

**-l** *language*

Uses the specified scripting language. By default, JavaScript is used. To use other scripting languages, you must specify the corresponding script engine's JAR file with the **-cp** or **-classpath** option.

**-e** *script*

Evaluates the specified script. This option can be used to run one–line scripts that are specified completely on the command line.

**-encoding** *encoding*

Specifies the character encoding used to read script files.

**-f** *script–file*

Evaluates the specified script file (batch mode).

**-f** – Enters interactive mode to read and evaluate a script from standard input.

**-help** or **-?**

Displays a help message and exits.

**-q** Lists all script engines available and exits.

**ARGUMENTS**

If arguments are present and if no **-e** or **-f** option is used, then the first argument is the script file and the rest of the arguments, if any, are passed as script arguments. If arguments and the **-e** or the **-f** option are used, then all arguments are passed as script arguments. If arguments **-e** and **-f** are missing, then the interactive mode is used.

**EXAMPLE OF EXECUTING INLINE SCRIPTS**

```
jrunscript -e "print('hello world')"  
jrunscript -e "cat('http://www.example.com')"
```

**EXAMPLE OF USING SPECIFIED LANGUAGE AND EVALUATE THE SCRIPT FILE**

```
jrunscript -l js -f test.js
```

**EXAMPLE OF INTERACTIVE MODE**

```
jrunscript  
js> print('Hello World\n');  
Hello World  
js> 34 + 55  
89.0  
js> t = new java.lang.Thread(function() { print('Hello World\n'); })  
Thread[Thread-0,5,main]  
js> t.start()  
js> Hello World  
  
js>
```

**RUN SCRIPT FILE WITH SCRIPT ARGUMENTS**

In this example, the `test.js` file is the script file. The `arg1`, `arg2`, and `arg3` arguments are passed to the script. The script can access these arguments with an arguments array.

```
jrunscript test.js arg1 arg2 arg3
```