

I love **JSLint**.

It's a tool, that every language should have. It defines its own coding style, no doubt. But I like that style as its clean, crisp and small at a time.

<http://crockfordfacts.com/> | Douglas Crockford], the inventor of JSLint and living „Java Script machine“, has put up this wonderful little site, <http://www.jslint.com>], around the Java-Script-Function of JSLint. It's nice, but a little inconvenient when you're dealing with a „write - correct - write - correct“-Situation.

Douglas had a special Rhino ((Mozilla's Java-implementation of JavaScript)) version available at one time, but somehow the old page is not available anymore. Rhino, however, is nicely integrated into (at least my current) Linux distribution and runnable from the command line.

But it's incompatible with the way, JSLint operates (being solely a JavaScript-function).

I wrapped up this small script to make JSLint accessible from the command line using rhino:

```
load("fulljslint.js");
if(JSLINT(readFile(arguments[0])) === false) {
for (i = 0; i < JSLINT.errors.length; i = i + 1) { error = JSLINT.errors[i]; print (error.line + ", " +
error.character + ": " + error.reason); } }
```

Put it into the source directory of JSLint, which you can find at its

<https://github.com/douglascrockford/JS�int> | Github repository] and you're all set. Run it using

```
rhino rhinojslint.js
```