Node.js Global Objects

Node.js global objects are global in nature and available in all modules. You don't need to include these objects in your application; rather they can be used directly. These objects are modules, functions, strings and object etc. Some of these objects aren't actually in the global scope but in the module scope.

A list of Node.js global objects are given below:

* \_\_dirname
* \_\_filename
* Console
* Process
* Buffer
* setImmediate(callback[, arg][, ...])
* setInterval(callback, delay[, arg][, ...])
* setTimeout(callback, delay[, arg][, ...])
* clearImmediate(immediateObject)
* clearInterval(intervalObject)
* clearTimeout(timeoutObject)

Node.js \_\_dirname

It is a string. It specifies the name of the directory that currently contains the code.

*File: global-example1.js*

53.4M

979

HTML Tutorial

1. console.log(\_\_dirname);

Open Node.js command prompt and run the following code:

1. node global-example1.js





Node.js \_\_filename

It specifies the filename of the code being executed. This is the resolved absolute path of this code file. The value inside a module is the path to that module file.

*File: global-example2.js*

1. console.log(\_\_filename);

Open Node.js command prompt and run the following code:

1. node global-example2.js



