Node.js Process

Node.js provides the facility to get process information such as process id, architecture, platform, version, release, uptime, upu usage etc. It can also be used to kill process, set uid, set groups, unmask etc.

The process is a global object, an instance of EventEmitter, can be accessed from anywhere.

Node.js Process Properties

A list of commonly used Node.js process properties are given below.

|  |  |
| --- | --- |
| **Property** | **Description** |
| arch | returns process architecture: 'arm', 'ia32', or 'x64' |
| args | returns commands line arguments as an array |
| env | returns user environment |
| pid | returns process id of the process |
| platform | returns platform of the process: 'darwin', 'freebsd', 'linux', 'sunos' or 'win32' |
| release | returns the metadata for the current node release |
| version | returns the node version |
| versions | returns the node version and its dependencies |

## Node.js Process Properties Example

Let's see the simple process **example to print architecture, pid, platform and version** of the process.

1. console.log(`Process Architecture: ${process.arch}`);
2. console.log(`Process PID: ${process.pid}`);
3. console.log(`Process Platform: ${process.platform}`);
4. console.log(`Process Version: ${process.version}`);

Let's see another process **example to print command line arguments**. Here node is considered as the first argument, filename is considered as the second argument and actual command line arguments are considered as third, fourth, fifth and so on.

*File: process\_example2.js*

1. process.argv.forEach((value, index, array) => {
2. console.log(`${index}: ${value}`);
3. });

Node.js Process Functions

A list of commonly used Node.js process functions are given below.

|  |  |
| --- | --- |
| **Function** | **Description** |
| cwd() | returns path of current working directory |
| hrtime() | returns the current high-resolution real time in a [seconds, nanoseconds] array |
| memoryUsage() | returns an object having information of memory usage. |
| process.kill(pid[, signal]) | is used to kill the given pid. |
| uptime() | returns the Node.js process uptime in seconds. |

Node.js Process Functions Example

Let's see the process **example to print current working directory and uptime** of the process.

*File: process\_example3.js*

1. console.log(`Current directory: ${process.cwd()}`);
2. console.log(`Uptime: ${process.uptime()}`);