# NODE JS

>> npm install -D @types/node

## OS

var os = require('os')

console.log(os.platform()); *// Win32*

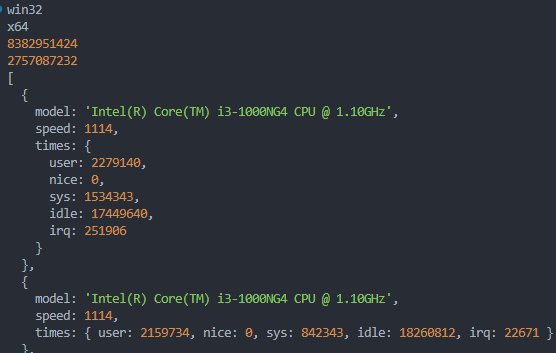
console.log(os.arch()); *// x64 - Architecture 32 Bit or 64 Bit (x64)*

console.log(os.totalmem());

console.log(os.freemem());

console.log(os.cpus());

**Output**



## File System

### WRITE

#### SYNC

fs.writeFileSync('sample.txt', 'Node is awesome..3.')

console.log('Created File')

console.log('End')

#### ASYNC

fs.writeFile('sample.txt', 'Node is awesome..3.', (*err*) => {

  if (*err*) throw *err*;

  console.log('Created File')

})

#### APPEND

fs.appendFile('sample.txt', '\nMore data appended 2.', (*err*) => {

  if (*err*) throw *err*;

  console.log('Appended to File')

})

#### RENAME

fs.rename('sample.txt', 'demo.txt', (*err*) => {

  if (*err*) throw *err*;

  console.log('Renamed File')

})

### READ

fs.readFile('demo.txt', (*err*, *data*) => {

  if (*err*) throw *err*;

  console.log(*data*)

})

**OUTPUT**

*/// <Buffer 4e 6f 64 65 20 69 73 20 61 77 65 73 6f 6d 65 2e 2e 33 2e 4d 6f 72 74 61 ... 35 more bytes>*

Encoded string, so decode use **data.toString()** or specify the encoding argument as below,

fs.readFile('demo.txt', 'utf8', (*err*, *data*) => {

  if (*err*) throw *err*;

  console.log(*data*)

})

### UNLINK (DELETE)

fs.unlink('demo.txt', (*err*) => {

  if (*err*) throw *err*;

  console.log('File Deleted')

})

## HTTP Request

var http = require('http')

var server = http.createServer((*req*, *res*) => {

*res*.write("<h1>NODY HODDY</h1>")

*res*.end("Welcome to node")

}).listen(5000)

### JSON response using FS

var fs = require('fs')

var server = http.createServer((*req*, *res*) => {

  fs.readFile('users.json', 'utf-8', (*err*, *data*) => {

*res*.write(*data*);

*res*.end()

  })

})

server.listen(5000)

### JSON response

Need to set the response header content-type to “json”

var rawJsonData = require('./users.json');

var server = http.createServer((*req*, *res*) => {

*res*.writeHead(200, {

    "content-type": "application/json"

  })

*res*.end(JSON.stringify(rawJsonData))

})

server.listen(5000)

## NODE MON

To avoid manual restart Node Server

>> npm i nodemon -g

**To start -** nodemon <file-name>

nodemon http.js

## Scripts Block in Package.json

Scripts block in package.json is used to configure commands to run our application or testsBottom of Form

. It is a key/value pair of strings.

**Key** is alias name and **Value** is command to be executed when executing the alias name.

### Predefined Keys

test, start, stop, restart

To execute the **Predefined keys**, we can simply execute – “**npm start**” , “npm stop”

For other **Custom keys** like “*dev*”, we need to execute with run – “**npm run dev**”

# Express

>> npm install express

>> npm install -D @types/express