Learn Node.js By Building Real-World Apps

* *Andrew Mead*

# Section-3 > Node.js Module System

--------------------------------------------------------------------------------------------------------------

## 3.1 Import node.js core modules

* The modules we use to work need to be imported using require function.
* FILE SYSTEM: module allows to access the files on OS we're working on.
* 1st method - fs.writeFileSync and fs.writeFile:
  + It takes two arguments: file name and contents to insert in it.

Note:

1. creates a file and write some data into it (it there is no file in the working directory)
2. overwrites content to a file (if the given file name already exists in the working directory)

--------------------------------------------------------------------------------------------------------------

## 3.2 Import and Export created files

* To load in a file that we create in the same working directory we use same require function
* To import a variable or any object

Note: everything in JS is treated as an object

we have to use...

1. "module.exports = <variable / object name>" to export a variable / object from a file.
2. require statement in the file where we import the certain variable or an object

--------------------------------------------------------------------------------------------------------------

## 3.3 import NPM modules

* when we install nodejs into our local machine... npm also comes with it
* NPM was already installed on our machine... this gives us the access to differ NPM modules
* to use these NPM modules...

1. Initialize NPM in our project:

* run single cmd... <npm init -y>
* it initializes the NPM in our project and creates configuration file to manage all of the dependencies
* after initialization it adds "package.json" into working directory (we use this to manage all the installed packages)

2. Install the necessary modules (which we use inside our project)

* run the cmd... <npm i "package-name"@"version"> ("i" for install)
* when we install... package-lock.json and node\_modules have been added into directory
* we load the package inside working file using... "require" function.

## 3.5 Global NPM modules and nodemon

intro:

1. so far we worked with locally installed packages because the packages are listed in the package.json file.

-> these get loaded into our working file using "require()" function.

2. when we install a package globally we do not load it in directly into our working files.

-> We are going to install "NODEMON" which runs application whenever there is a change in our code.

-> To install a package globally use... <npm i "package-name"@version -g>