NodeJS

What is NodeJS?

Node.js is an open source, cross-platform runtime environment for developing server-side and networking applications. Node.js applications are written in JavaScript, and can be run within the Node.js runtime on OS X, Microsoft Windows, and Linux.

Node.js also provides a rich library of various JavaScript modules which simplifies the development of web applications using Node.js to a great extent.

Node.js = Runtime Environment + JavaScript Library

Features of NodeJS?

- Asynchronous and Event Driven
- Very Fast - Being built on Google Chrome's V8 JavaScript Engine, Node.js library is very fast in code execution
- Single Threaded but Highly Scalable
- No Buffering - Node.js applications never buffer any data. These applications simply output the data in chunks.
- License − Node.js is released under the MIT license.

Where to Use NodeJS?

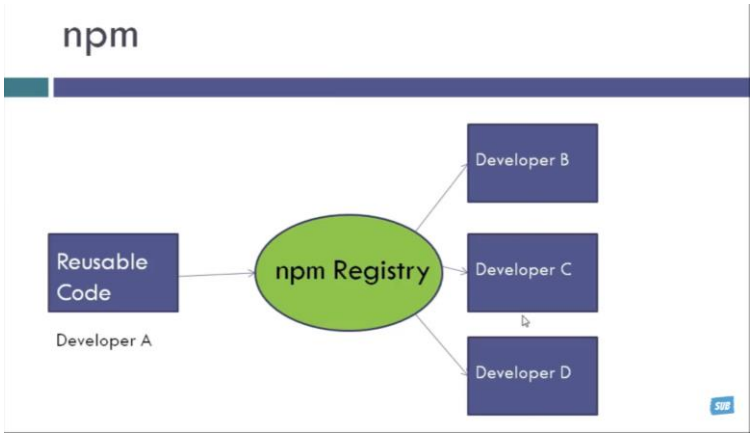
Following are the areas where Node.js is proving itself as a perfect technology partner.

- I/O bound Applications
- Data Streaming Applications
- Data Intensive Real-time Applications (DIRT)
- JSON APIs based Applications
- Single Page Applications

Where Not to Use NodeJS?

It is not advisable to use Node.js for CPU intensive applications.

Introduction to NPM



npm is the world's largest Software Registry. The registry contains over 800,000 code packages. Open-source developers use npm to share software. Many organizations also use npm to manage private development.

npm is free to use

we can download all npm public software packages without any registration or login.

npm includes a CLI (Command Line Client) that can be used to download and install software:

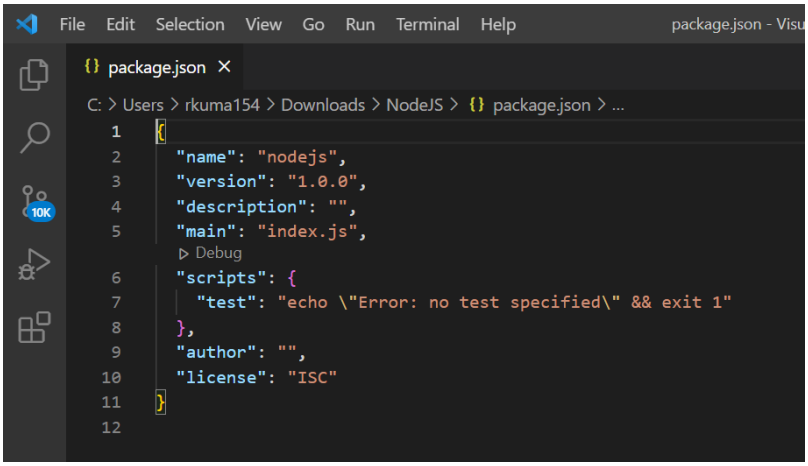
npm install <package>

npm is install with nodejs

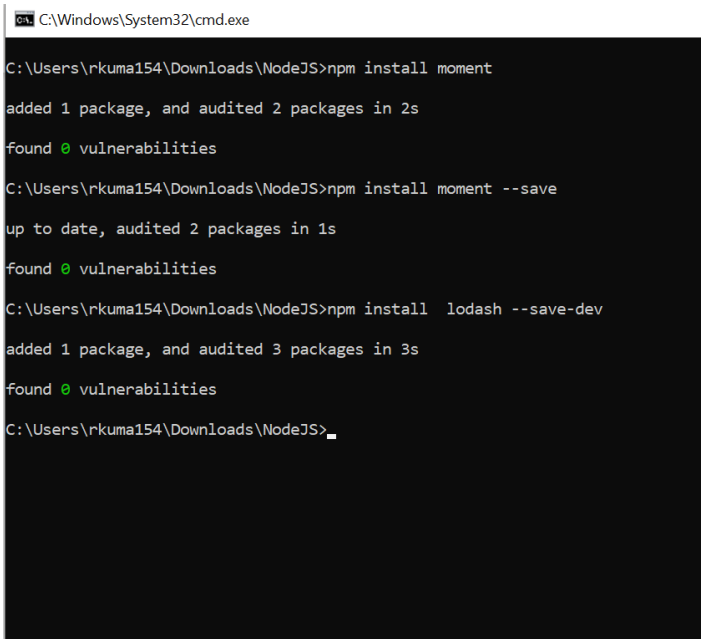
check the version of node and npm

all npm packages are defined in files called package.json

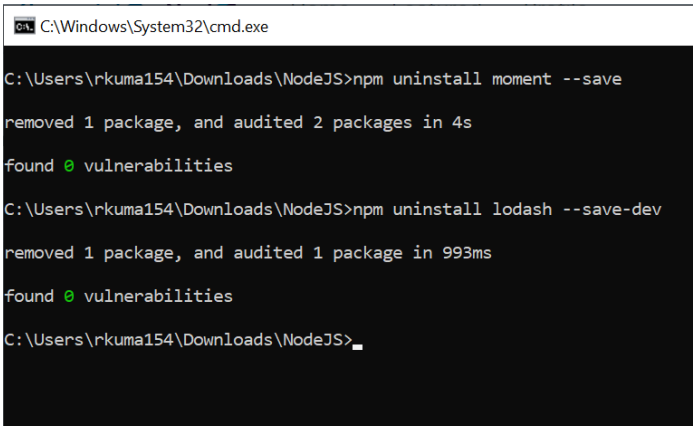
the content of package.json must be written in JSON like



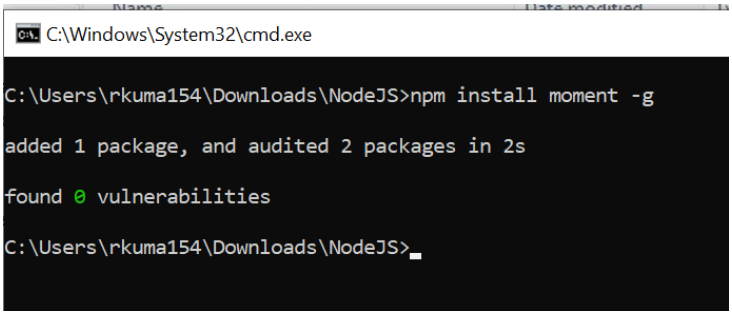
Npm install local packages - Local packages are installed in the directory where you run npm install <package-name> and they are put in the node_modules folder under this directory



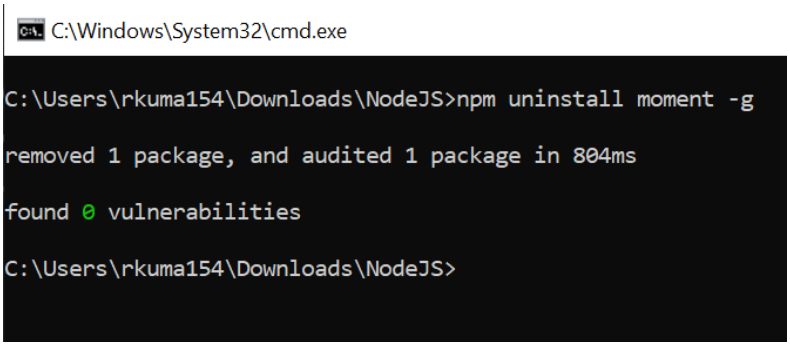
Npm uninstall local packages – for uninstall the local packages



Npm install global packages - global packages are all put in a single place in your system (exactly where depends on your setup), regardless of where you run npm install <package-name> -g



Npm uninstall global packages – for uninstall the global packages



Npm listing packages-

1. Npm list- it is going to display the packages in the tree like structure which also show the packages that are installed package depends upon
2. Npm list –depth 1- we can also restrict the depth of this tree using this command

3. If we specify depth 0 it was just going to list of all the packages and node dependencies for that particular packages
4. Npm list –global true –depth 0 - for display the list of global packages with depth 0

C:\Windows\System32\cmd.exe

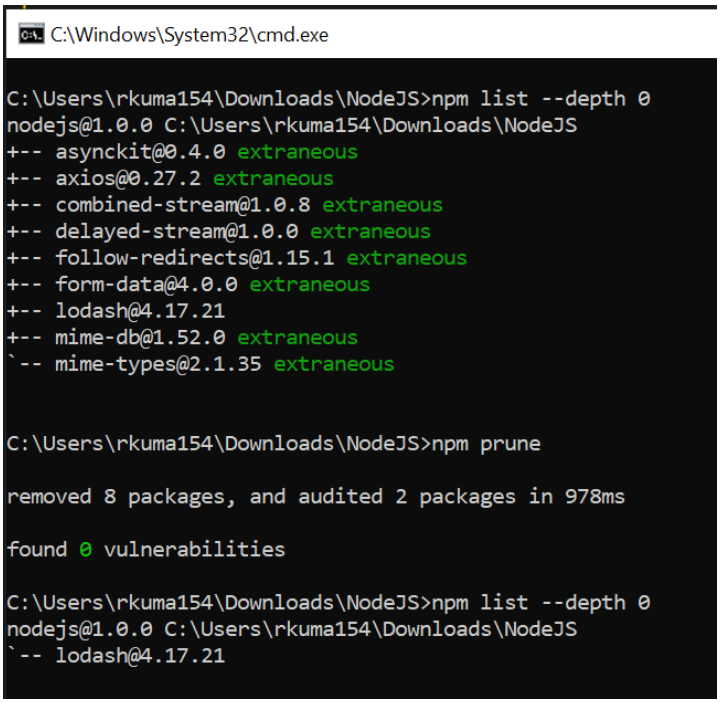
```
:\Users\rkuma154\Downloads\NodeJS>npm list
odejs@1.0.0 C:\Users\rkuma154\Downloads\NodeJS
-- lite-server@2.6.1
-- lodash@4.17.21


:\Users\rkuma154\Downloads\NodeJS>npm list --depth 1
odejs@1.0.0 C:\Users\rkuma154\Downloads\NodeJS
-- lite-server@2.6.1
 +-- browser-sync@2.27.10
 +-- connect-history-api-fallback@1.6.0
 +-- connect-logger@0.0.1
 +-- lodash@4.17.21 deduped
 `-- minimist@1.2.6
-- lodash@4.17.21


:\Users\rkuma154\Downloads\NodeJS>npm list  --depth 0
odejs@1.0.0 C:\Users\rkuma154\Downloads\NodeJS
-- lite-server@2.6.1
-- lodash@4.17.21


:\Users\rkuma154\Downloads\NodeJS>npm list --global true --depth 0
:\Users\rkuma154\AppData\Roaming\npm
-- (empty)


:\Users\rkuma154\Downloads\NodeJS>
```

Npm versioning – used for change the version of packages

C:\Windows\System32\cmd.exe

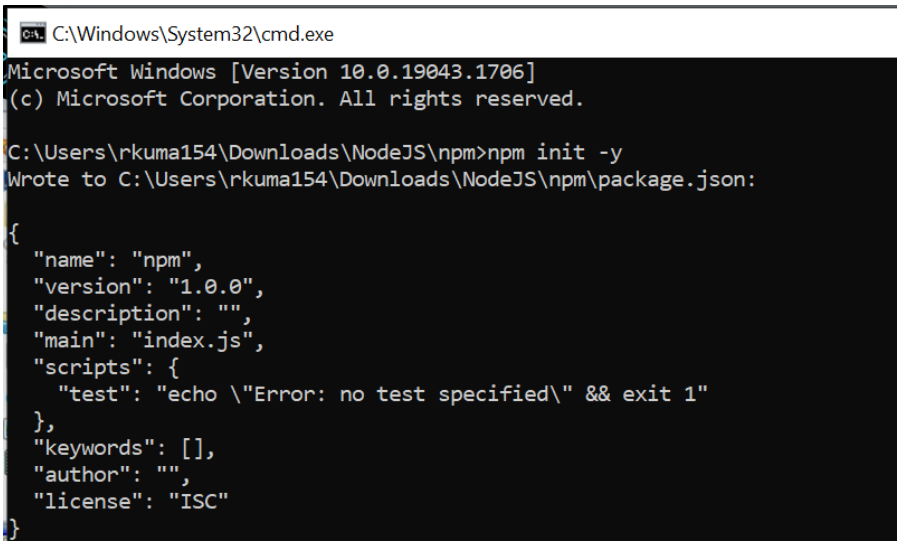
```
C:\Users\rkuma154\Downloads\NodeJS>npm install lodash --save

up to date, audited 174 packages in 1s

6 packages are looking for funding
  run `npm fund` for details

found 0 vulnerabilities

C:\Users\rkuma154\Downloads\NodeJS>npm install lodash@3.3.0 --save

added 3 packages, changed 1 package, and audited 177 packages in 10s

6 packages are looking for funding
  run `npm fund` for details

1 critical severity vulnerability

To address all issues (including breaking changes), run:
  npm audit fix --force

Run `npm audit` for details.

C:\Users\rkuma154\Downloads\NodeJS>npm install lodash@4.16

changed 1 package, and audited 177 packages in 5s

6 packages are looking for funding
  run `npm fund` for details

1 critical severity vulnerability

To address all issues, run:
  npm audit fix --force

Run `npm audit` for details.
```

Npm prune – this command is use to remove all the extraneous packages from our project

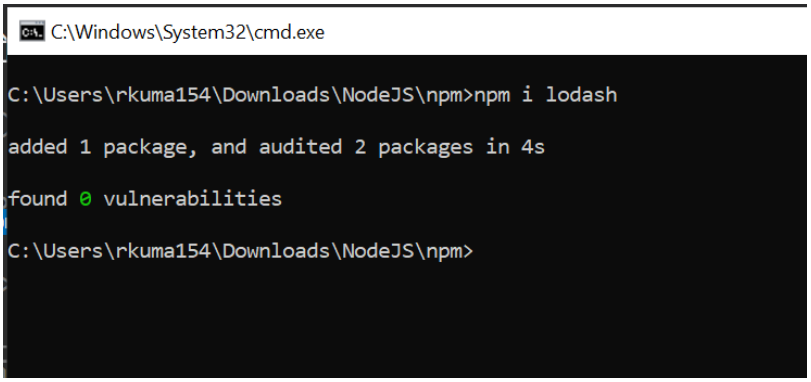


Npm shortcuts –

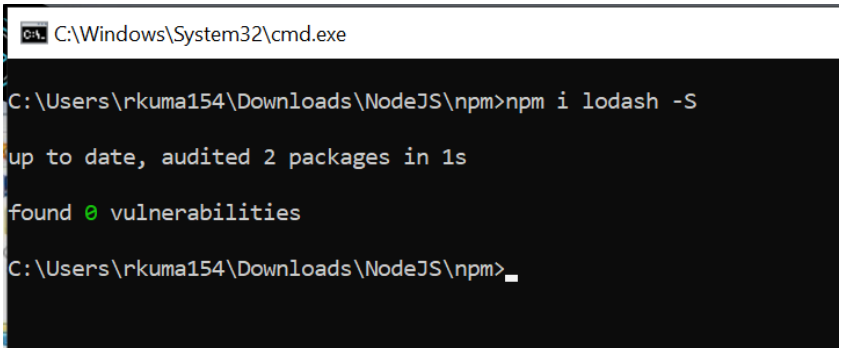
1. Creating a package.json file with default value



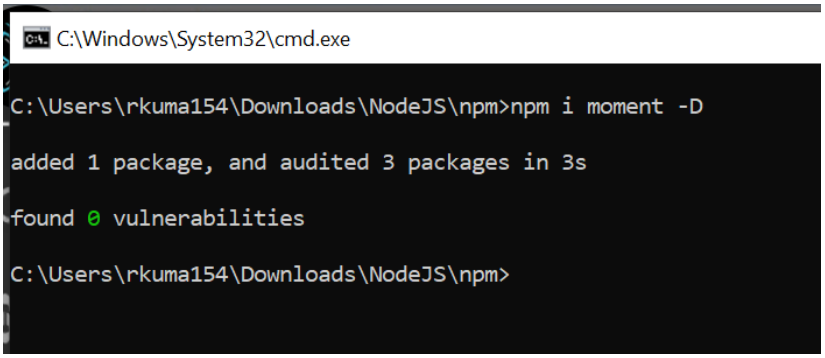
2. Install package locally



3. Save package in package.json
   Before – npm install lodash –save
   After- npm i lodash -S



4. Save a package as a dev dependency
   Before – npm install moment –save-dev
   After – npm I moment -D



5. Change
   --global to -g
   --version to -v
   --registry to -reg

- `-v : --version`
- `-h , -? , --help , -H : --usage`
- `-s , --silent : --loglevel silent`
- `-q , --quiet : --loglevel warn`
- `-d : --loglevel info`
- `-dd , --verbose : --loglevel verbose`
- `-ddd : --loglevel silly`
- `-g : --global`
- `-C : --prefix`
- `-l : --long`
- `-m : --message`
- `-p , --porcelain : --parseable`
- `-reg : --registry`
- `-f : --force`
- `-desc : --description`
- `-S : --save`
- `-D : --save-dev`
- `-O : --save-optional`
- `-B : --save-bundle`
- `-E : --save-exact`
- `-y : --yes`
- `-n : --yes false`
- `ll` and `la` commands: `ls --long`

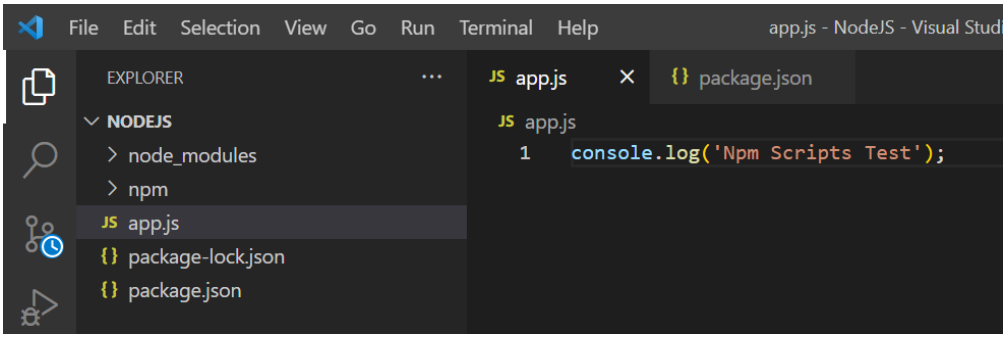
Npm Scripts :

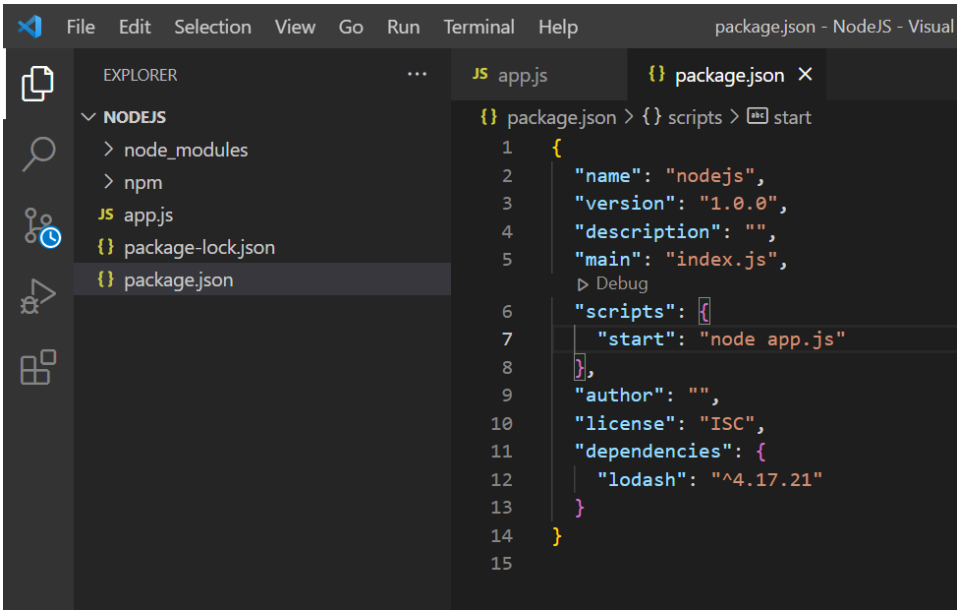
First we change the scripts like **node app.js** then create app.js file and enter some code inside this file

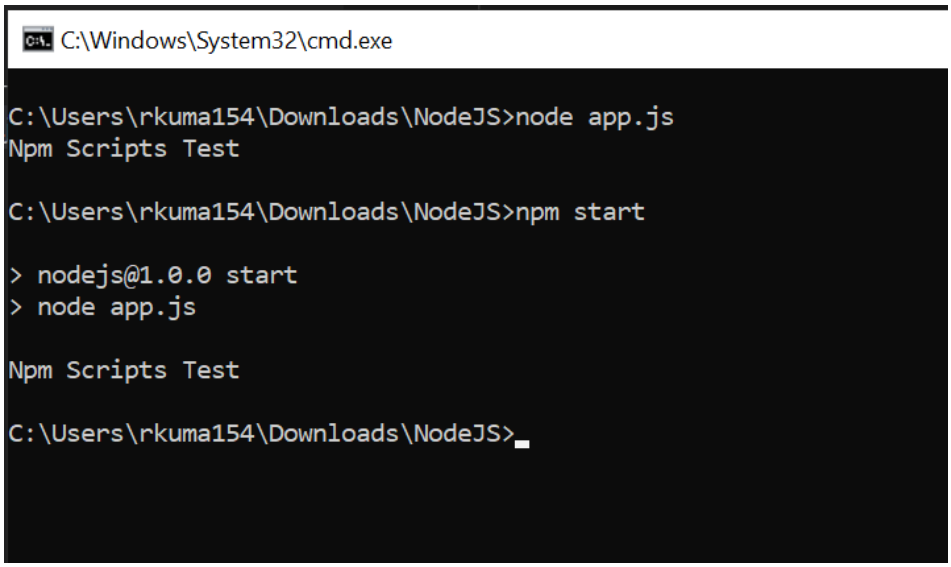
And then go to command prompt and run it by using commands like:

npm start

node app.js




File Edit Selection View Go Run Terminal Help
package.json - NodeJS - Visual
EXPLORER
NODEJS
node_modules
npm
app.js
package-lock.json
package.json
app.js
package.json
package.json > scripts > start
{
"name": "nodejs",
"version": "1.0.0",
"description": "",
"main": "index.js",
Debug
"scripts": {
"start": "node app.js"
},
"author": "",
"license": "ISC",
"dependencies": {
"lodash": "^4.17.21"
}
}


C:\Windows\System32\cmd.exe
C:\Users\rkuma154\Downloads\NodeJS>node app.js
Npm Scripts Test
C:\Users\rkuma154\Downloads\NodeJS>npm start
> nodejs@1.0.0 start
> node app.js
Npm Scripts Test
C:\Users\rkuma154\Downloads\NodeJS>