**Steps to create Node application**

**Step1:**

Create directory for your new application

D:\antony\nodespringsecuritywsclient

**Step2:**

D:\antony\nodespringsecuritywsclient >npm init

Use the npm init command to create a package.json file for your application.

{

"name": "springsecuritywsclient",

"version": "1.0.0",

"description": "Sample Node Application for create web services client",

"main": "index.js",

"scripts": {

"test": "echo \"Error: no test specified\" && exit 1"

},

"author": "Antony",

"license": "ISC"

}

What is package.json?

package.json is a plain JSON(Java Script Object Notation) text file which contains all metadata information about Node JS Project or application

Why that filename is “package”: because Node JS platform manages every feature as separate component. That component is also known as “Package” or “Module”.

Who use package.json file?

NPM (Node Package Manager) uses this package.json file information about Node JS Application information or Node JS Package details.

package.json file contains a number of different directives or elements. It uses these directives to tell NPM “How to handle the module or package”.

package.json file contains two mandatory directives;

1. name
2. version

**Optional Directives**

package.json file may contain many optional directives. We should use them only if our Node JS package really requires. We will discuss some important optional directives here.

1. **dependencies**: Every Node JS Project or Module may dependent on other Node JS or Third Party Modules or our own Custom Modules. We should provide all those dependencies by using this “dependencies” directive.

**Step3:**

D:\antony\nodespringsecuritywsclient >npm install express --save

Now install Express in the "NodeProject" directory and save it in the dependencies list of your package.json file

{

"name": "springsecuritywsclient",

"version": "1.0.0",

"description": "Sample Node Application for create web services client",

"main": "index.js",

"scripts": {

"test": "echo \"Error: no test specified\" && exit 1"

},

"author": "Antony",

"license": "ISC",

"dependencies": {

"express": "^4.16.4"

}

}

**Step4:**

Create a file named index.js in the root of the " nodespringsecuritywsclient " application directory (D:\antony\nodespringsecuritywsclient)

**Step5:**

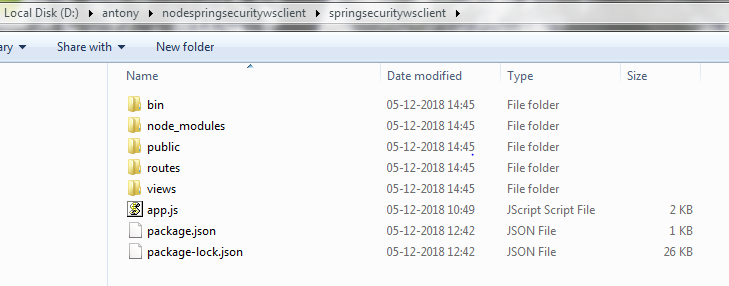
The Express Application Generator tool generates an Express application "skeleton". Install the generator using NPM as shown (the -g flag installs the tool globally so that you can call it from anywhere):

D:\antony\nodespringsecuritywsclient >npm install express-generator -g

To create an Express app named "springsecuritywsclient " with the default settings, navigate to where you want to create it and run the app as shown:

D:\antony\nodespringsecuritywsclient >express springsecuritywsclient

NPM will create the new Express app in a sub folder of your current location, displaying build progress on the console. On completion, the tool will display the commands you need to enter to install the Node dependencies and start the app.



**Step 6:**

Install all the dependencies for the springsecuritywsclient app using NPM as shown(node\_modules):

D:\antony\nodespringsecuritywsclient >cd springsecuritywsclient

D:\antony\nodespringsecuritywsclient\springsecuritywsclient >npm install

**Step7:**

Add your own web service js under” routes” folder

e.g:

NodeService.js

var express = require('express');

var router = express.Router();

var fs = require("fs");

/\* GET home page. \*/

router.get('/', function(req, res, next) {

res.render('index', { title: 'Express' });

});

router.get('/listUsers', function (req, res) {

fs.readFile(\_\_dirname + "/" + "users.json", 'utf8', function (err, data) {

console.log( data );

res.end( data );

});

})

module.exports = router;

And configure “NodeService.js” in “app.js” as follows

var serviceRouter = require('./routes/NodeService');

app.use('/ springsecuritywsclient', serviceRouter);

**Step 8:**

Start the server:

D:\antony\nodespringsecuritywsclient\springsecuritywsclient>SET DEBUG=springsecuritywsclient:\* & npm start

**Step 9:**

Web Service URL: http://localhost:3000/springsecuritywsclient/listUsers