This chapter includes creating and setting up a MEAN application. We are using NodeJS and ExpressJS together to create the project.
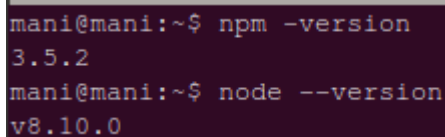
Prerequisites

Before we begin with creating a MEAN application, we need to install required prerequisites.

You can install latest version of Node.js by visiting the Node.js website at Node.js ⧉ (This is for Windows users). When you download Node.js, npm will get installed automatically on your system. Linux users can install the Node and npm by using this link ⧉ .

Check the version of Node and npm by using the below commands −

```
$ node --version
$ npm --version
```

The commands will display the versions as shown in the below image −

Creating Express Project

Create a project directory by using mkdir command as shown below −

```
$ mkdir mean-demo //this is name of repository
```

The above directory is the root of node application. Now, to create package.json file, run the below command −

```
$ cd webapp-demo
$ npm init
```

The init command will walk you through creating a package.json file −

This utility will walk you through creating a package.json file. It only covers the most common items, and tries to guess sensible defaults.

```
See `npm help json` for definitive documentation on these fields and exactly what th
Use `npm install --save` afterwards to install a package and save it as a dependency

Press ^C at any time to quit.
name: (mean-demo) mean_tutorial
version: (1.0.0)
description: this is basic tutorial example for MEAN stack
```

```
entry point: (index.js) server.js
test command: test
git repository:
keywords: MEAN,Mongo,Express,Angular,Nodejs
author: Manisha
license: (ISC)
About to write to /home/mani/work/rnd/mean-demo/package.json:


{
    "name": "mean_tutorial",
    "version": "1.0.0",
    "description": "this is basic tutorial example for MEAN stack",
    "main": "server.js",
    "scripts": {
        "test": "test"
    },
    "keywords": [
        "MEAN",
        "Mongo",
        "Express",
        "Angular",
        "Nodejs"
    ],
    "author": "Manisha",
    "license": "ISC"
}
```

```
Is this ok? (yes) yes
```

Click yes and a folder structure as below will be generated −

```
-mean-demo
    -package.json
```

The *package.json* file will have the following info −

```
{
    "name": "mean_tutorial",
    "version": "1.0.0",
    "description": "this is basic tutorial example for MEAN stack",
    "main": "server.js",
    "scripts": {
        "test": "test"
    },
    "keywords": [
        "MEAN",
        "Mongo",
        "Express",
        "Angular",
        "Nodejs"
```

```
    ],
    "author": "Manisha",
    "license": "ISC"
}
```

Now to configure the Express project into current folder and install configuration options for the framework, use the below command −

```
npm install express --save
```

Go to your project directory and open package.json file, you will see the below information −

```
{
    "name": "mean_tutorial",
    "version": "1.0.0",
    "description": "this is basic tutorial example for MEAN stack",
    "main": "server.js",
    "scripts": {
        "test": "test"
    },
    "keywords": [
        "MEAN",
        "Mongo",
        "Express",
        "Angular",
        "Nodejs"
    ],
    "author": "Manisha",
    "license": "ISC",
    "dependencies": {
        "express": "^4.17.1"
    }
}
```

Here you can see express dependency is added to the file. Now, the project structure is as below −

```
-mean-demo
    --node_modules created by npm install
    --package.json tells npm which packages we need
    --server.js set up our node application
```

Running Application

Navigate to your newly created project directory and create a server.js file with below contents.
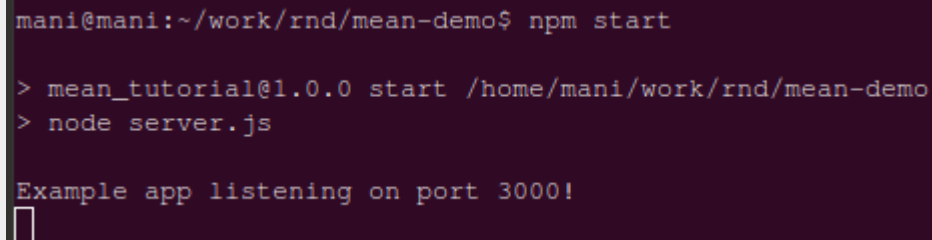
```
// modules =================================================
const express = require('express');
const app = express();
// set our port
const port = 3000;
app.get('/', (req, res) ⇒ res.send('Welcome to Tutorialspoint!'));
```

```
// startup our app at http://localhost:3000
app.listen(port, () ⇒ console.log(`Example app listening on port ${port}!`));
```

Next, run the application with the below command −

```
$ npm start
```

You will get a confirmation as shown in the image below −

```
mani@mani:~/work/rnd/mean-demo$ npm start

> mean_tutorial@1.0.0 start /home/mani/work/rnd/mean-demo
> node server.js

Example app listening on port 3000!
```

It informs that Express application is running. Open any browser and access the application using **http://localhost:3000**. You will see Welcome to Tutorialspoint! text as shown below −

```
←  →  C    ⓘ localhost:3000

Welcome to Tutorialspoint!
```