Table of Contents

[Change Log 2](#_Toc514956813)

[What is MEAN 2](#_Toc514956814)

[MongoDB 2](#_Toc514956815)

[Express 2](#_Toc514956816)

[Angular 3](#_Toc514956817)

[NodeJS 3](#_Toc514956818)

[RESTFul API 3](#_Toc514956819)

[Install Node JS (Windows) 3](#_Toc514956820)

[Install MongoDB 3](#_Toc514956821)

[Setting-up Mongo service. 4](#_Toc514956822)

[npm init (Project Initiation) 5](#_Toc514956823)

[Load this project into eclipse 6](#_Toc514956824)

[Install dependencies 6](#_Toc514956825)

[Build app.js 7](#_Toc514956826)

[Start application 7](#_Toc514956827)

[Application in browser 7](#_Toc514956828)

[Install npm monitor 7](#_Toc514956829)

[Monitoring server 8](#_Toc514956830)

[Building Routs 8](#_Toc514956831)

[Changes in app.js 8](#_Toc514956832)

[Update in Router.js 9](#_Toc514956833)

[API output in browser 9](#_Toc514956834)

[Establish Mongo DB connection 9](#_Toc514956835)

[Create a model and export 9](#_Toc514956836)

[Update to App.js 9](#_Toc514956837)

[Connection status 10](#_Toc514956838)

[Test ADD/Delete APIs with POSTMAN Chrome plug-in 10](#_Toc514956839)

[Building Angular Client project 11](#_Toc514956840)

[Install angular client 11](#_Toc514956841)

[Create angular project client using ng 11](#_Toc514956842)

[Run Client Project 12](#_Toc514956843)

[Do changes in app and verify 12](#_Toc514956844)

[Create component 13](#_Toc514956845)

[Create service 13](#_Toc514956846)

## Change Log

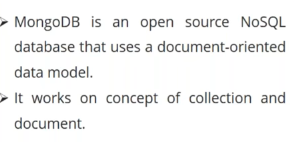
|  |  |
| --- | --- |
| Date | Details |
| 05/23/208 | * MEAN Basics * Node Installation * Mongo Installation * setting up mongo service * Creating new MEAN project. * Run simple app server script and validate.   Page 1 to 7 |
| 05/24/2018 | * Install ng monitor to auto deploy server. * Create a REST server project * Write services for CURD * Validate developed rest service with post-man plug-in * Create angular client project * Create service and modules to consume rest service.   Page 7 to 13 |

## What is MEAN

Mean is a full stack JS solution that helps you to build fast robust and maintainable production web application using MongoDB Express AngularJS, and Node JS.

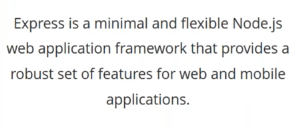


### MongoDB

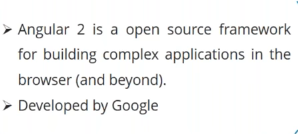


Collection : group of documents  
Documents: set of key value pair

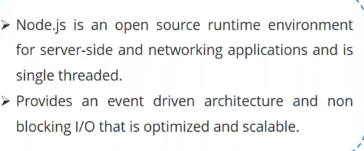
### Express



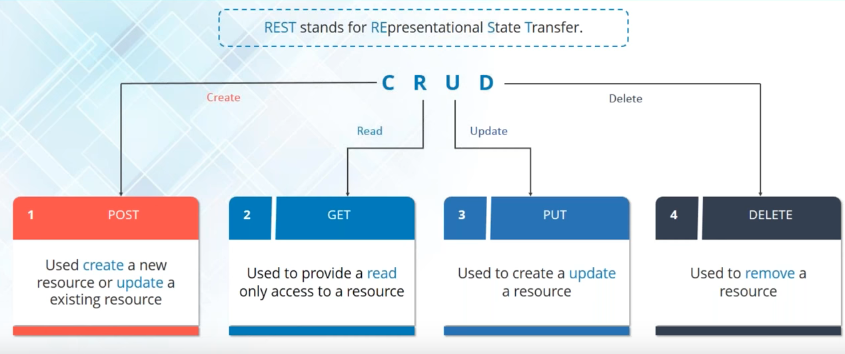
### Angular



### NodeJS

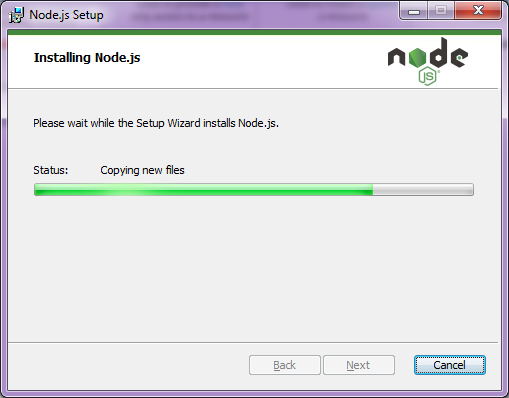


## RESTFul API



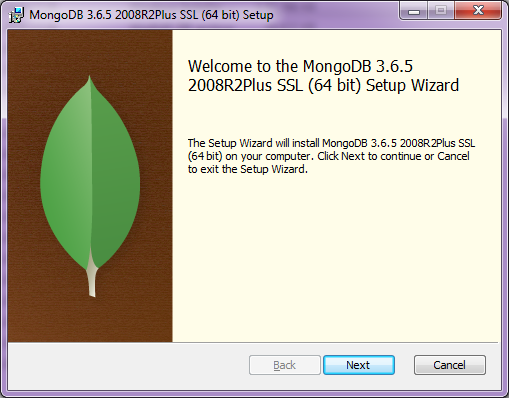
## Install Node JS (Windows)

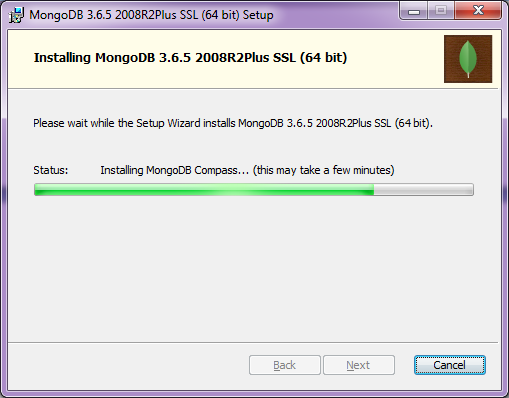
Download NodeJs exe from Nodejs.org and run exe.



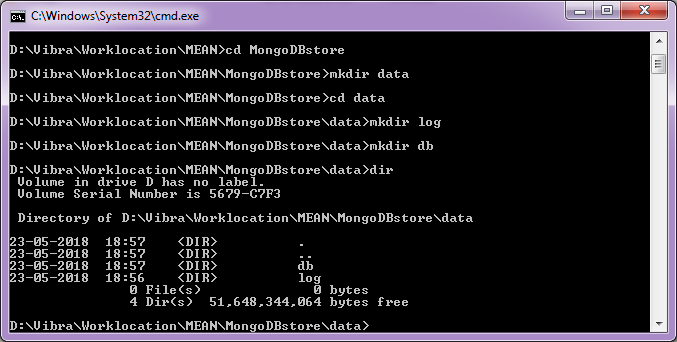
## Install MongoDB

Go to www.mongodb.com and download exe and install it.





### Setting-up Mongo service.



execute below commands. you can change the location/path based on your installation.

create mongod.cfg with below content.

systemLog:  
 destination: file  
 path: D:\Vibra\Worklocation\MEAN\MongoDBstore\data\log\mongod.log  
storage:  
 dbPath: D:\Vibra\Worklocation\MEAN\MongoDBstore\data\db

Run below commands in commandline with admin privillage.

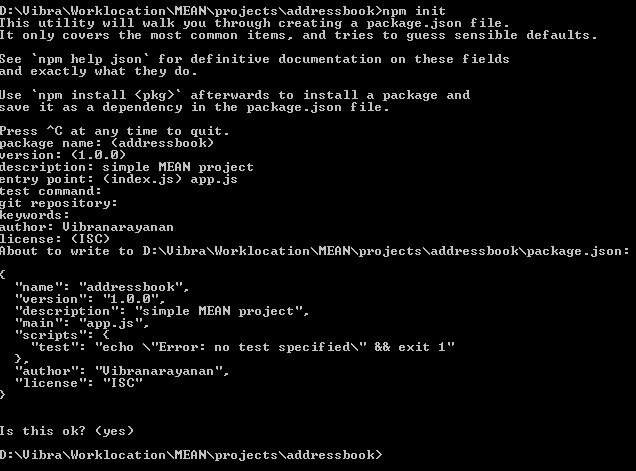
#"D:\Vibra\Worklocation\MEAN\mongo\bin\mongod.exe" --config "D:\Vibra\Worklocation\MEAN\mongo\mongod.cfg" --install

#sc.exe create MongoDB binPath= "\"D:\Vibra\Worklocation\MEAN\mongo\bin\mongod.exe\" --service --config=\"D:\Vibra\Worklocation\MEAN\mongo\mongod.cfg\"" DisplayName= "MongoDB" start= "auto"

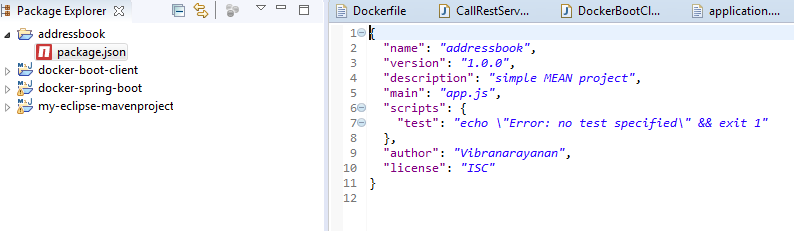
Result  
D:\Vibra\Worklocation\MEAN\MongoDBstore>net start MongoDB  
The MongoDB service is starting..  
The MongoDB service was started successfully.  
D:\Vibra\Worklocation\MEAN\MongoDBstore>

## npm init (Project Initiation)

Create a project folder and initiate project with npm command

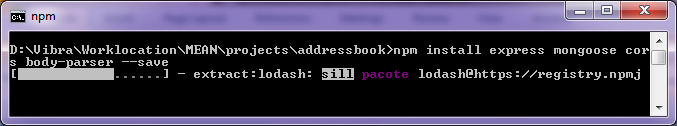


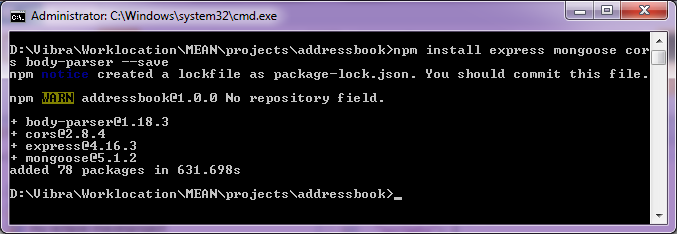
### Load this project into eclipse



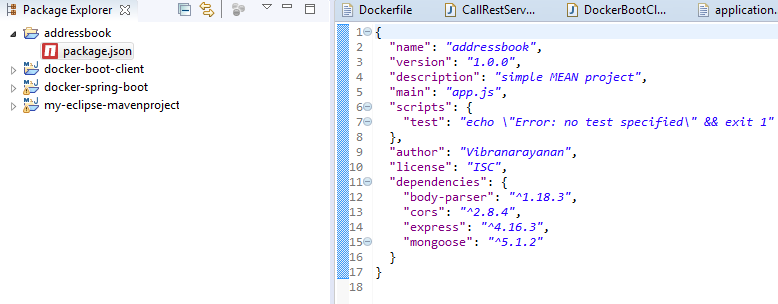
### Install dependencies

execute below command to install dependencies D:\Vibra\Worklocation\MEAN\projects\addressbook>npm install express mongoose cors body-parser --save

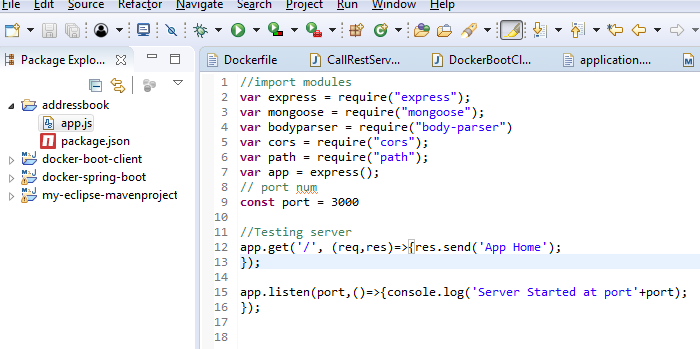




Dependencies updated automatically in package.json

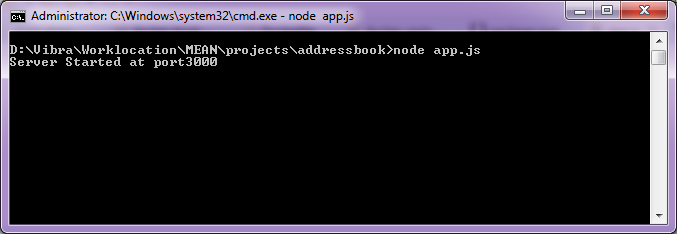


### Build app.js

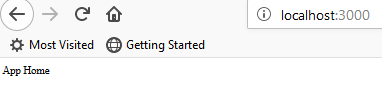


### Start application

execute node app.js. this will start the application 3000 port as mention above.

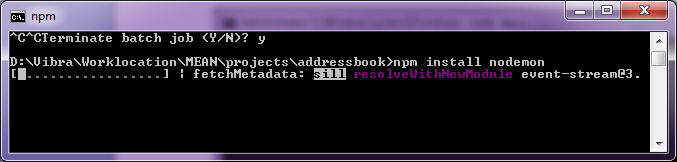


### Application in browser



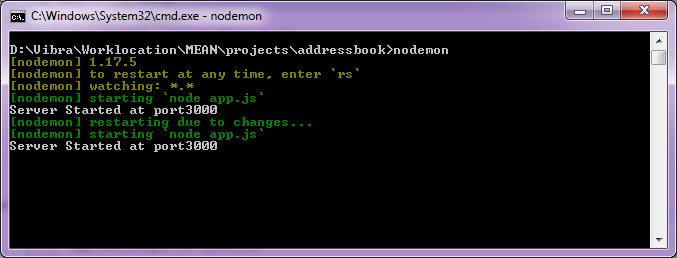
### Install npm monitor

This will help you to monitor code changes to rebuild the application based on new changes.



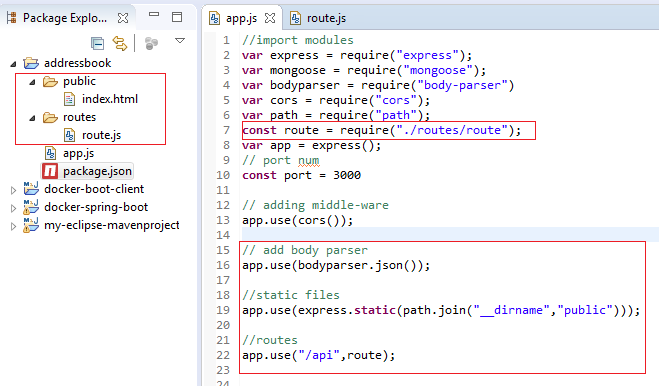
### Monitoring server

if we do any changes in our project files, this tool automatically builds it own. This will get server js mapped in package.js under main tag, and start the serever.

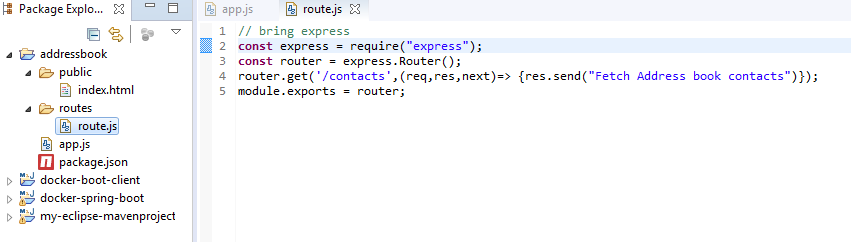


### Building Routs

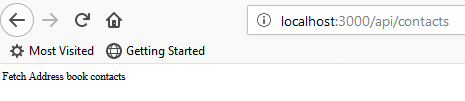
### Changes in app.js



### Update in Router.js

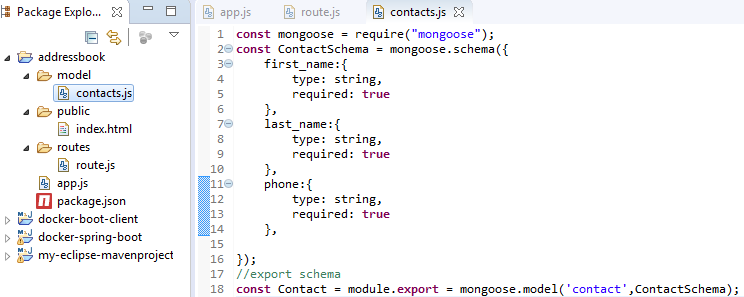


### API output in browser

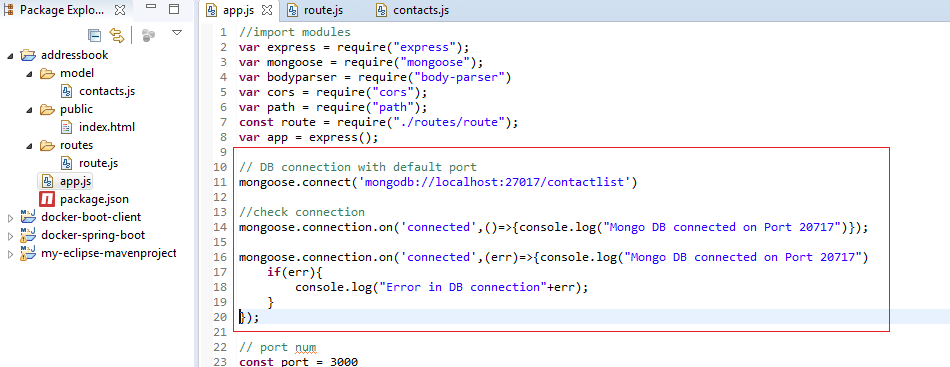


### Establish Mongo DB connection

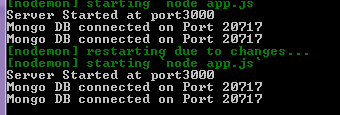
### Create a model and export



### Update to App.js

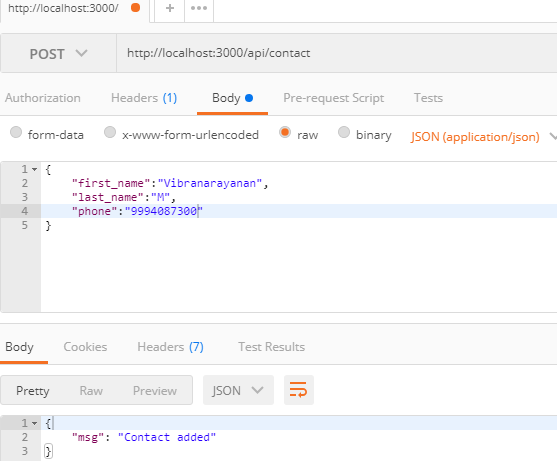


### Connection status

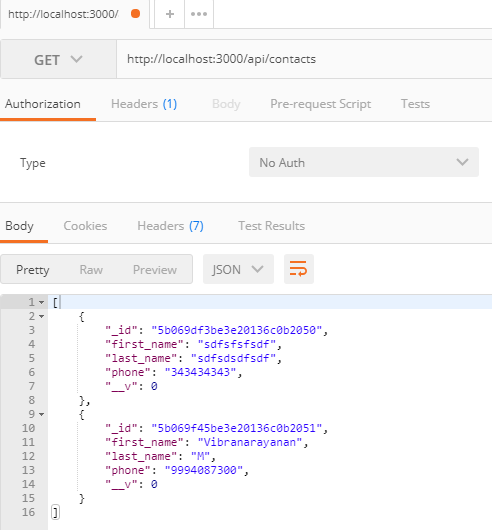


### Test ADD/Delete APIs with POSTMAN Chrome plug-in

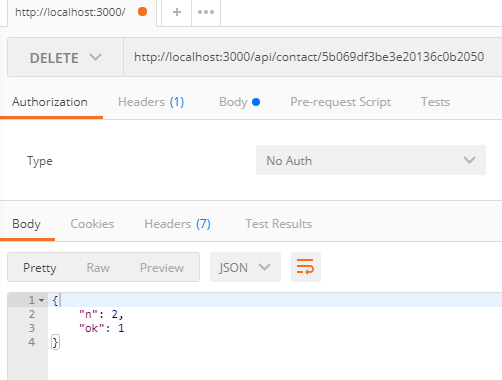
#### Testing Add Contact API



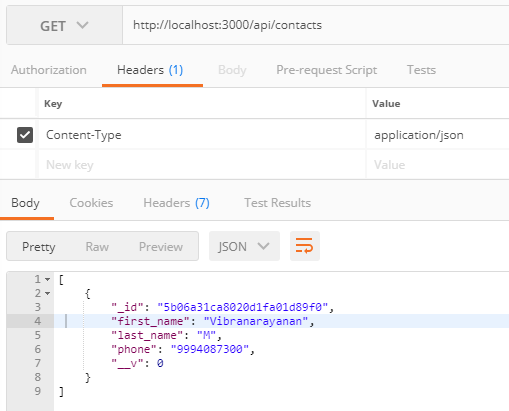
#### Test get Contact list API



#### Test Delete API



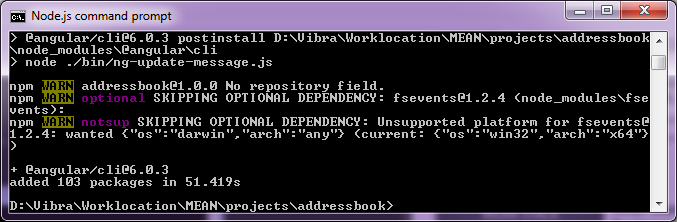
After delete list count reduced to 2 to 1

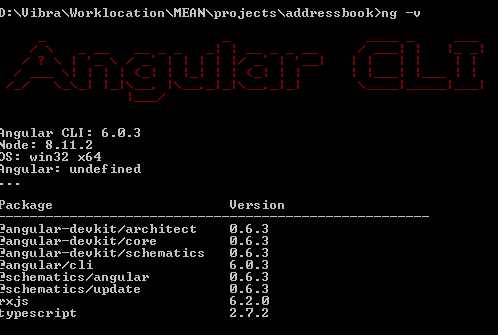


## Building Angular Client project

### Install angular client

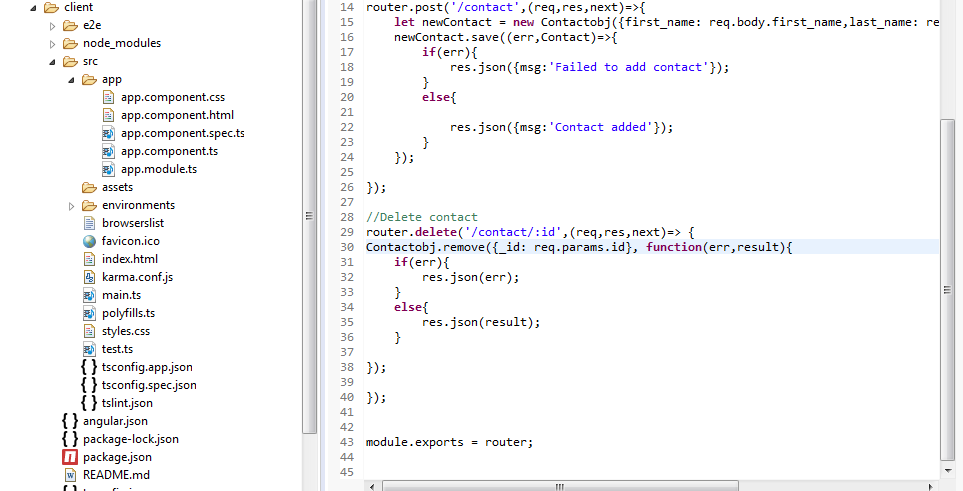
install angular client



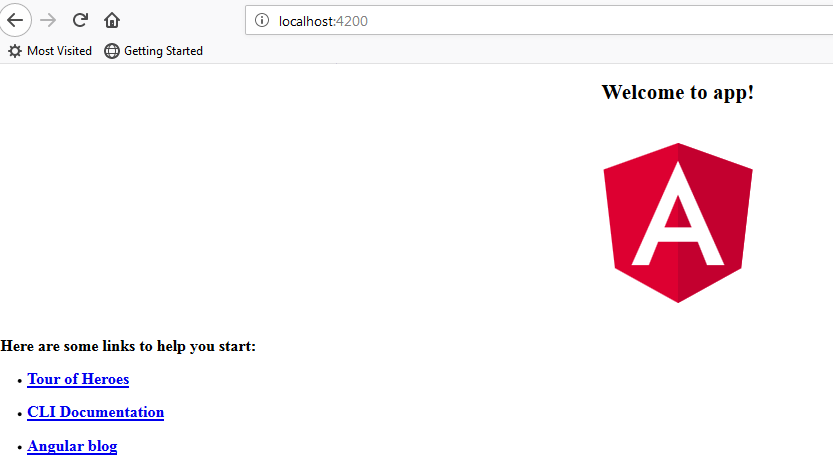


### Create angular project client using ng

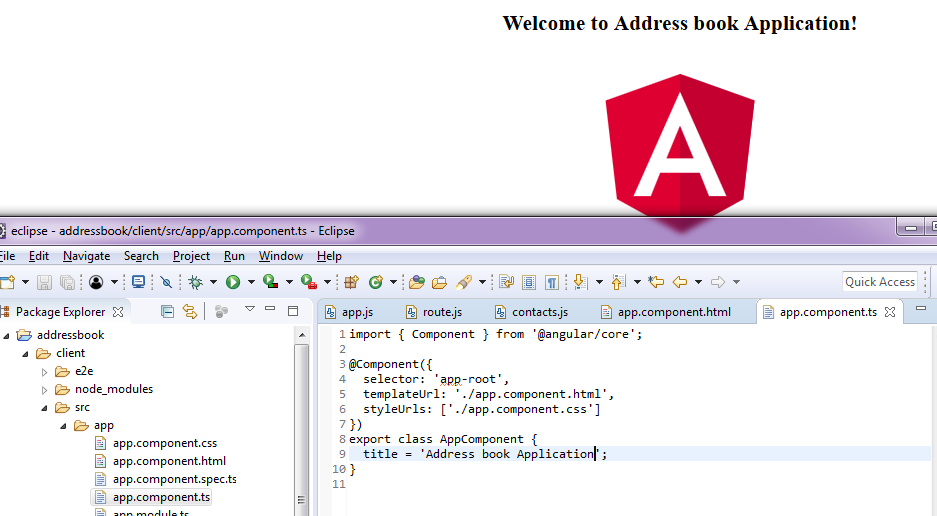
Type "ng new **client"** and execute this command. This will create required folder structure and files for angular client. refer below eclipse screen.



### Run Client Project

Client side app is ready for our action.

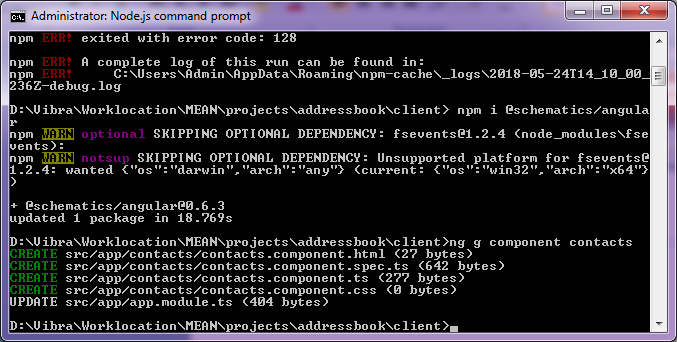
### Do changes in app and verify



### Create component

create component using ng g component contact. if you get any error like @schematics/angular run below command in admin mode of cmd line.

npm i @schematics/angular



### Create service

execute ng g service contact to generate serves for contact list module.

service and modules for contacts added after executing above commabds.

