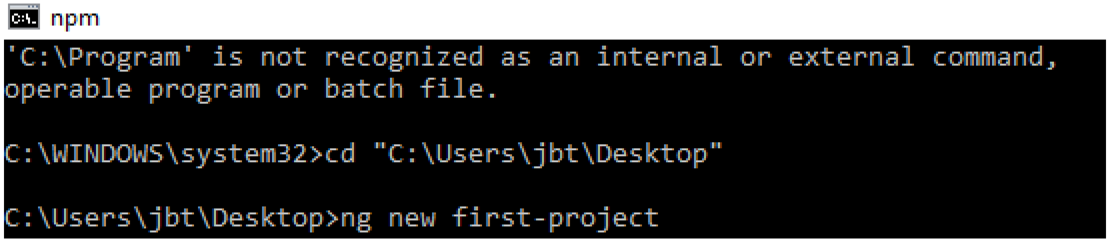
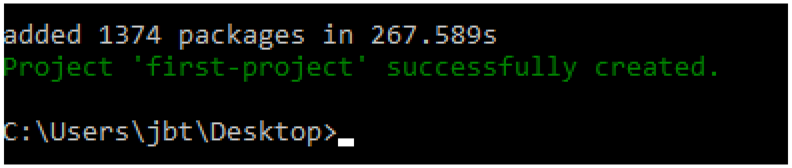
Step 1 – angular project

Create a new angular project with the following commands:



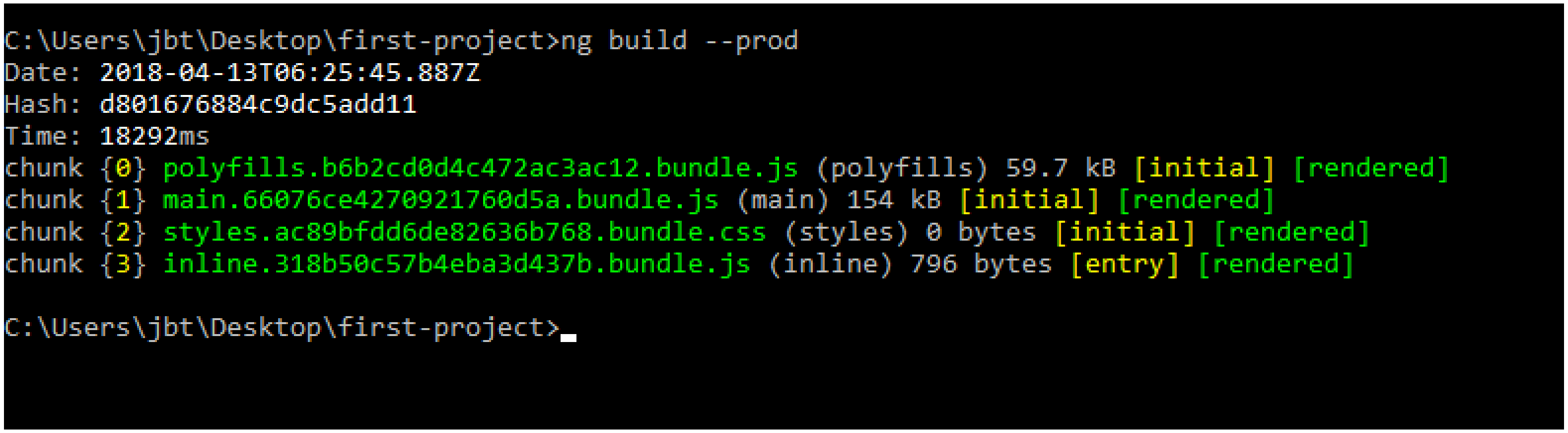
```
'C:\Program' is not recognized as an internal or external command,
operable program or batch file.

C:\WINDOWS\system32>cd "C:\Users\jbt\Desktop"

C:\Users\jbt\Desktop>ng new first-project
```

When the project is created successfully, you will see the following cli:



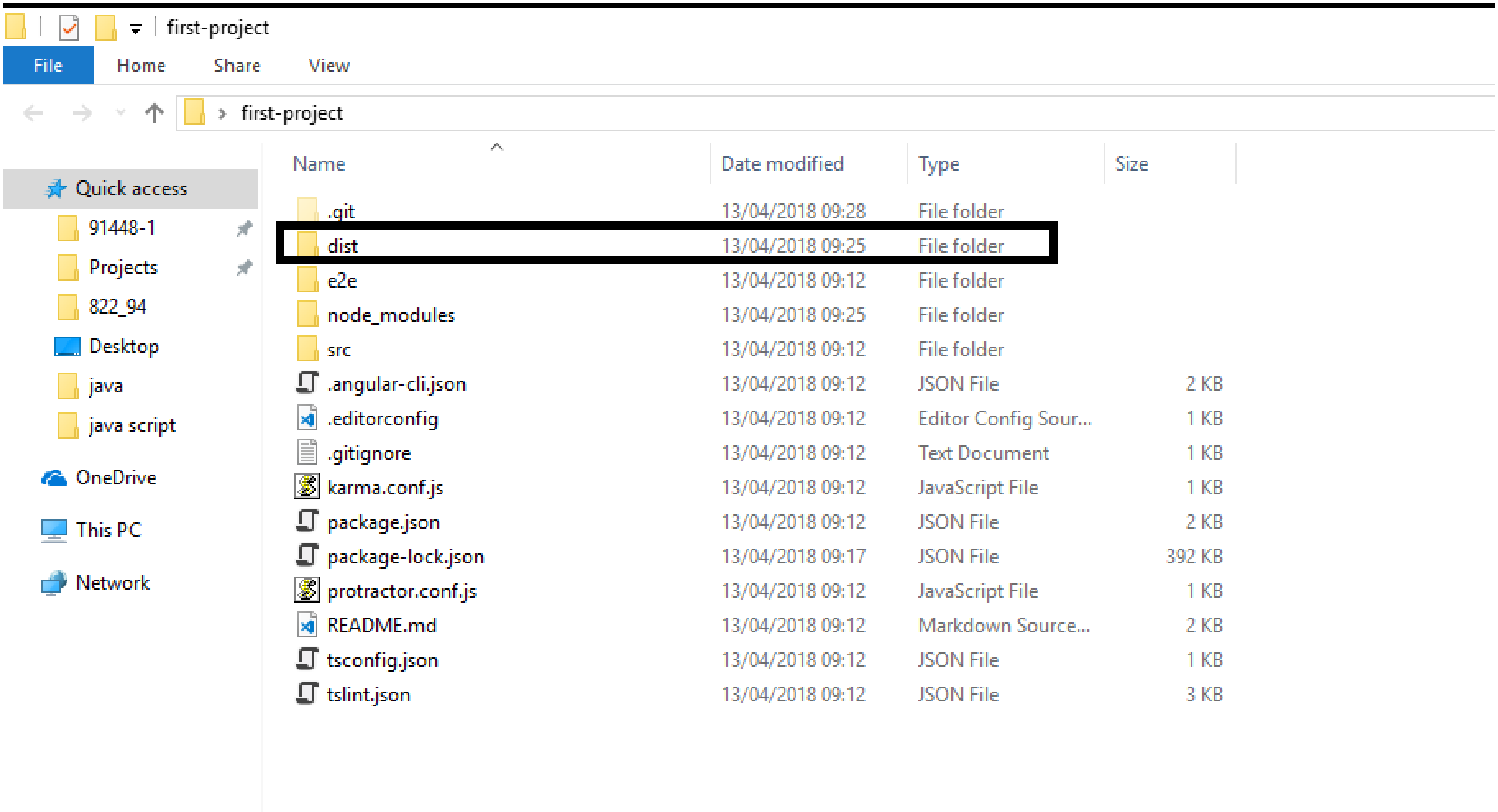
```
added 1374 packages in 267.589s
Project 'first-project' successfully created.

C:\Users\jbt\Desktop>_
```

Now, build the project, with the following command:



```
C:\Users\jbt\Desktop\first-project>ng build --prod
Date: 2018-04-13T06:25:45.887Z
Hash: d801676884c9dc5add11
Time: 18292ms
chunk {0} polyfills.b6b2cd0d4c472ac3ac12.bundle.js (polyfills) 59.7 kB [initial] [rendered]
chunk {1} main.66076ce4270921760d5a.bundle.js (main) 154 kB [initial] [rendered]
chunk {2} styles.ac89bfdd6de82636b768.bundle.css (styles) 0 bytes [initial] [rendered]
chunk {3} inline.318b50c57b4eba3d437b.bundle.js (inline) 796 bytes [entry] [rendered]

C:\Users\jbt\Desktop\first-project>
```

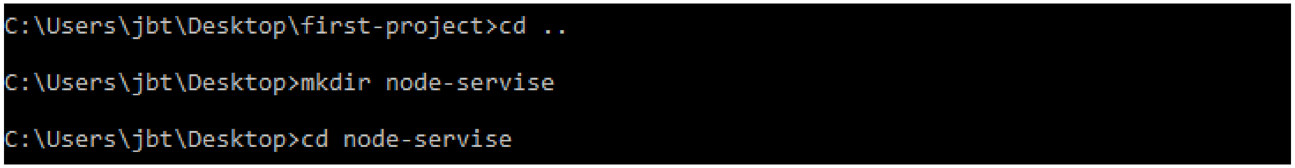
And you will get the "dist" directory:



Name	Date modified	Type	Size
.git	13/04/2018 09:28	File folder	
dist	13/04/2018 09:25	File folder	
e2e	13/04/2018 09:12	File folder	
node_modules	13/04/2018 09:25	File folder	
src	13/04/2018 09:12	File folder	
.angular-cli.json	13/04/2018 09:12	JSON File	2 KB
.editorconfig	13/04/2018 09:12	Editor Config Sour...	1 KB
.gitignore	13/04/2018 09:12	Text Document	1 KB
karma.conf.js	13/04/2018 09:12	JavaScript File	1 KB
package.json	13/04/2018 09:12	JSON File	2 KB
package-lock.json	13/04/2018 09:17	JSON File	392 KB
protractor.conf.js	13/04/2018 09:12	JavaScript File	1 KB
README.md	13/04/2018 09:12	Markdown Source...	2 KB
tsconfig.json	13/04/2018 09:12	JSON File	1 KB
tslint.json	13/04/2018 09:12	JSON File	3 KB

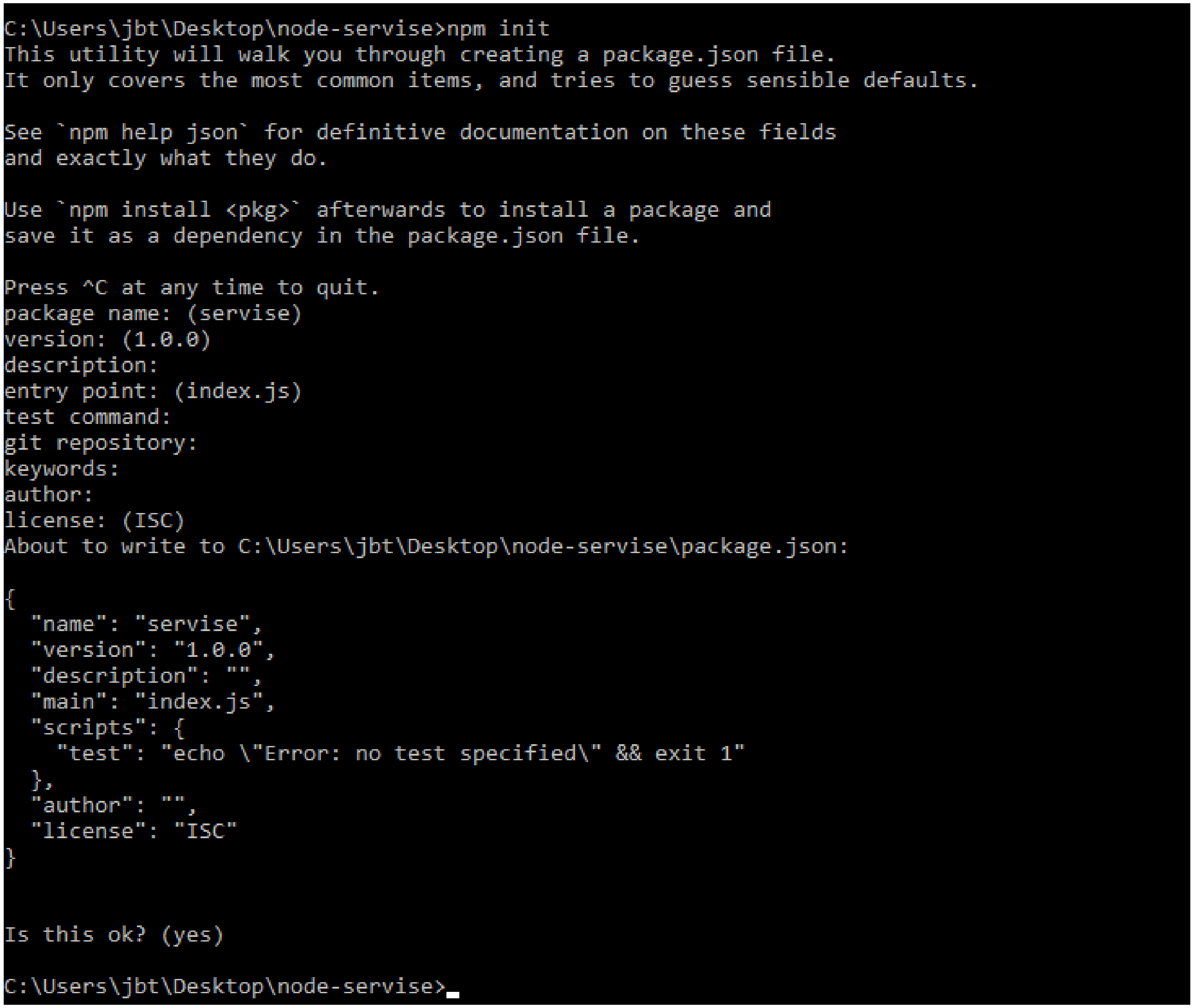
Step 2 – node project

Create a new folder for the node project, and change the cli path to this folder:



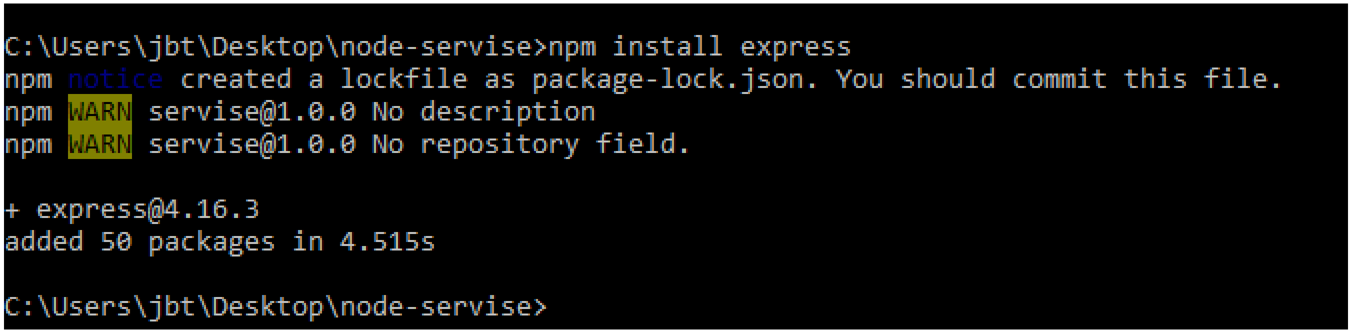
```
C:\Users\jbt\Desktop\first-project>cd ..

C:\Users\jbt\Desktop>mkdir node-servise

C:\Users\jbt\Desktop>cd node-servise
```

Add a "package.json" file, to the project folder, with the following command:



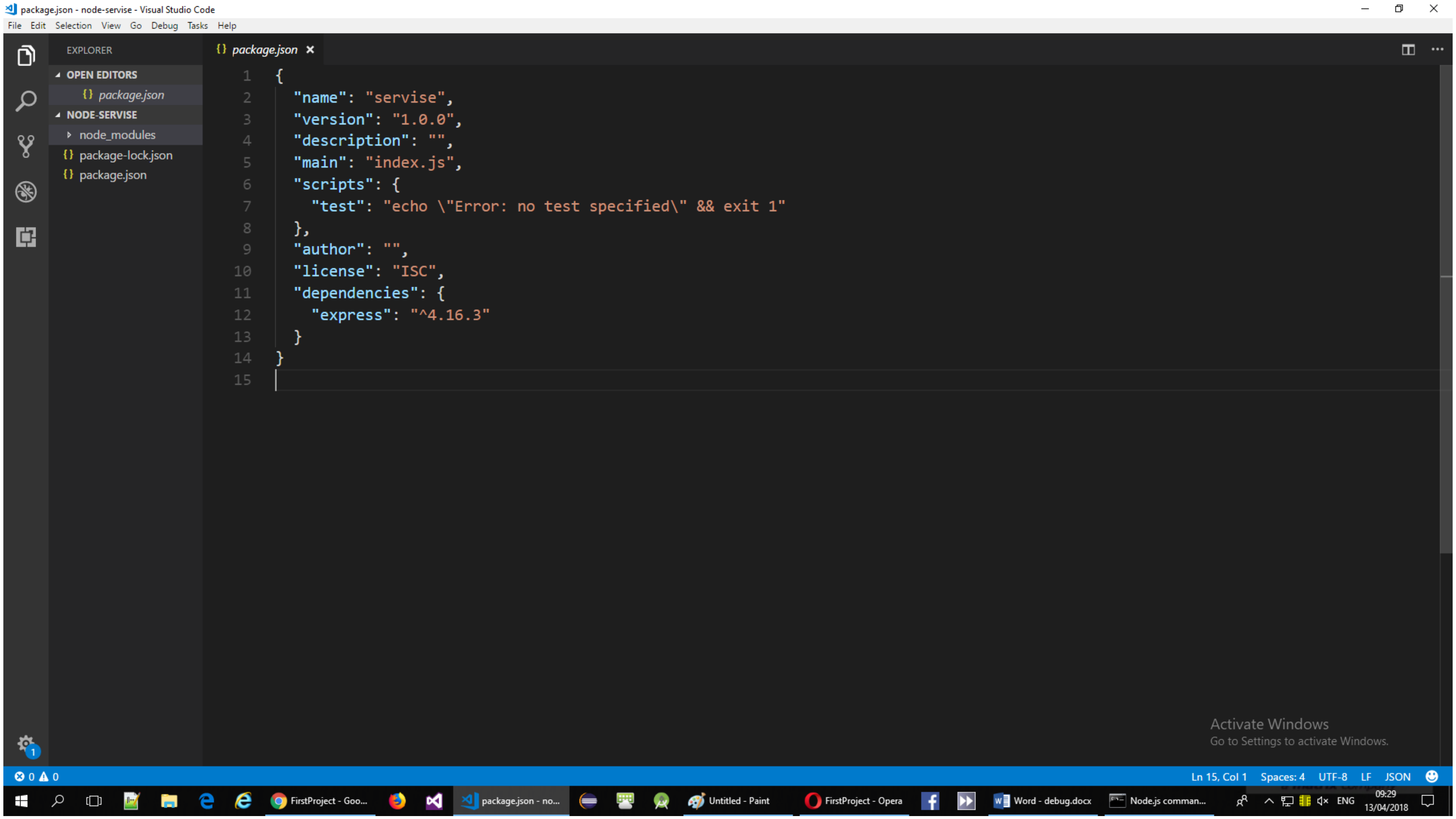
```
C:\Users\jbt\Desktop\node-servise>npm init
This utility will walk you through creating a package.json file.
It only covers the most common items, and tries to guess sensible defaults.

See `npm help json` for definitive documentation on these fields
and exactly what they do.

Use `npm install <pkg>` afterwards to install a package and
save it as a dependency in the package.json file.

Press ^C at any time to quit.
package name: (servise)
version: (1.0.0)
description:
entry point: (index.js)
test command:
git repository:
keywords:
author:
license: (ISC)
About to write to C:\Users\jbt\Desktop\node-servise\package.json:

{
  "name": "servise",
  "version": "1.0.0",
  "description": "",
  "main": "index.js",
  "scripts": {
    "test": "echo \"Error: no test specified\" && exit 1"
  },
  "author": "",
  "license": "ISC"
}


Is this ok? (yes)

C:\Users\jbt\Desktop\node-servise>
```

Next, install the "express" package, through the cli:



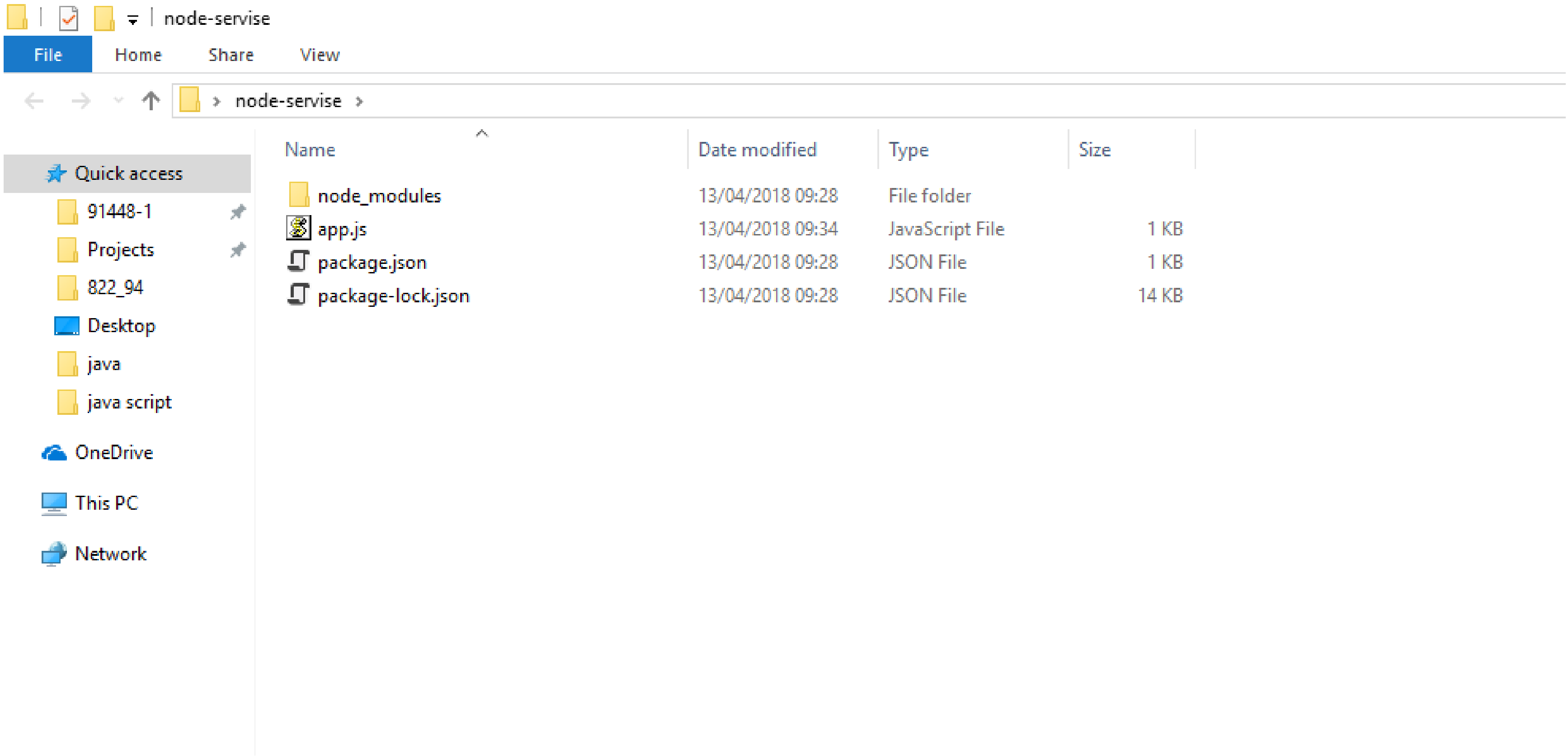
```
C:\Users\jbt\Desktop\node-servise>npm install express
npm notice created a lockfile as package-lock.json. You should commit this file.
npm WARN servise@1.0.0 No description
npm WARN servise@1.0.0 No repository field.

+ express@4.16.3
added 50 packages in 4.515s

C:\Users\jbt\Desktop\node-servise>
```

And open the project in visual studio code:



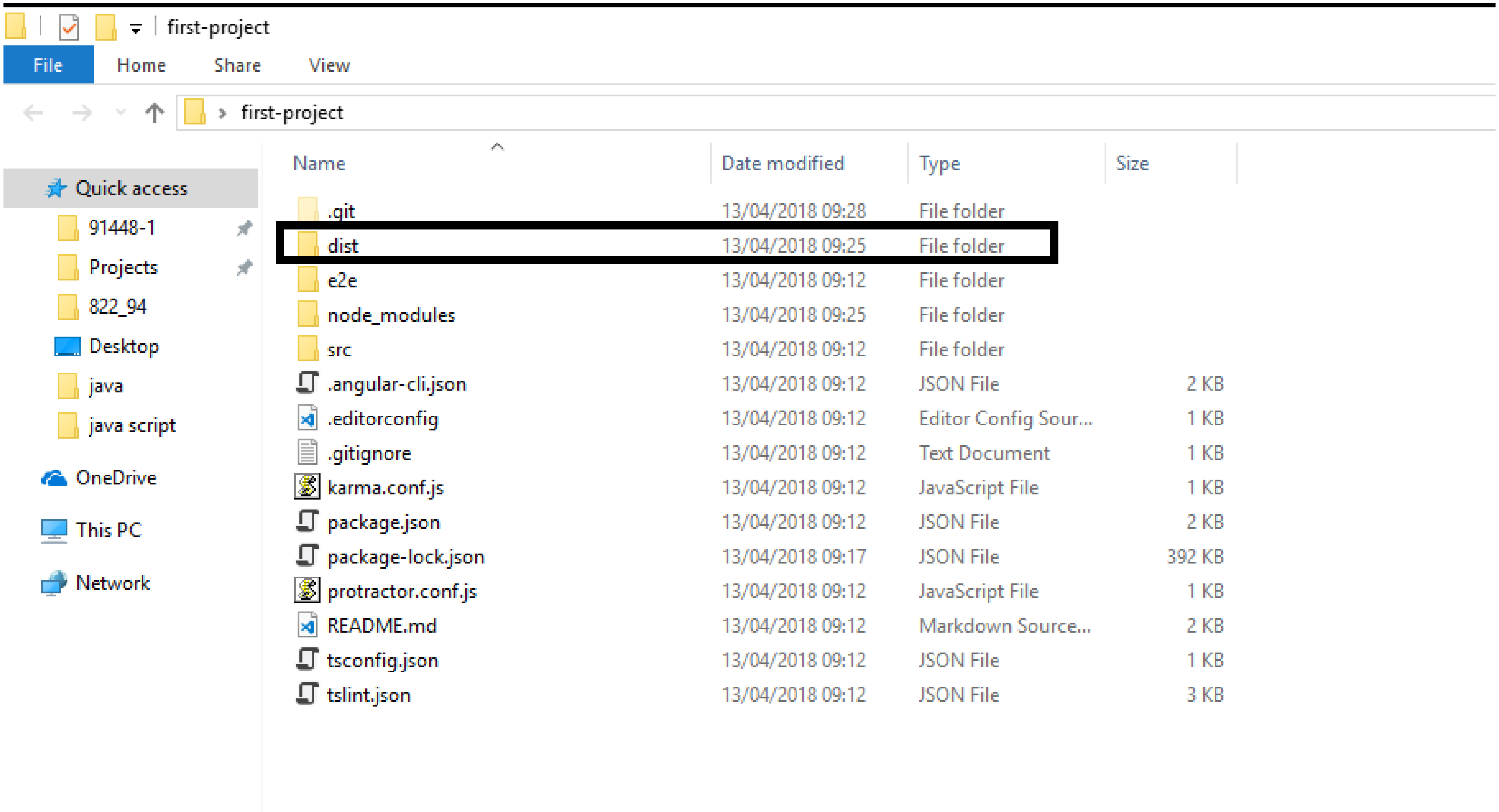
```
{
  "name": "servise",
  "version": "1.0.0",
  "description": "",
  "main": "index.js",
  "scripts": {
    "test": "echo \"Error: no test specified\" && exit 1"
  },
  "author": "",
  "license": "ISC",
  "dependencies": {
    "express": "^4.16.3"
  }
}

```

Open the project's folder:



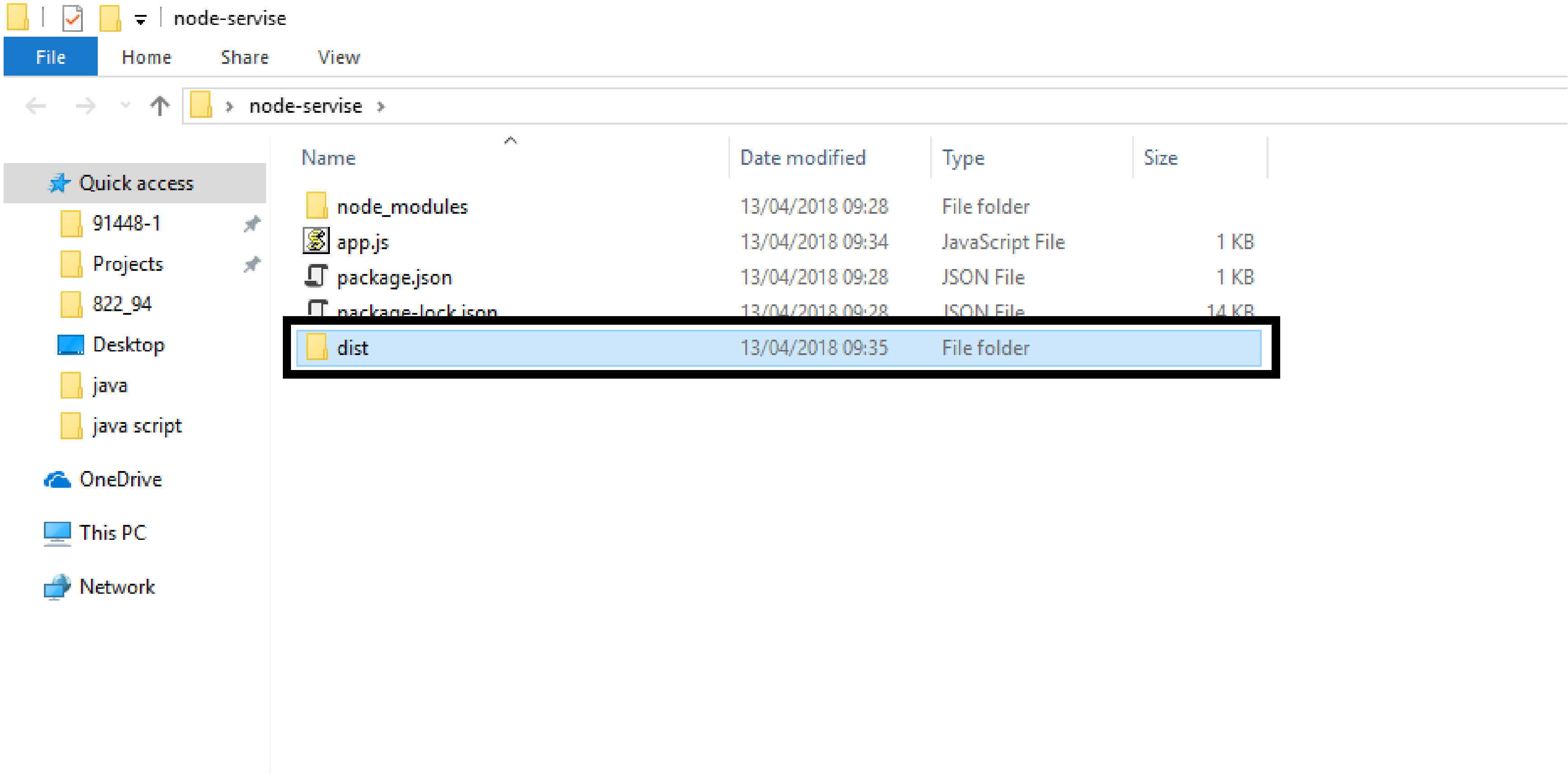
Name	Date modified	Type	Size
node_modules	13/04/2018 09:28	File folder	
app.js	13/04/2018 09:34	JavaScript File	1 KB
package.json	13/04/2018 09:28	JSON File	1 KB
package-lock.json	13/04/2018 09:28	JSON File	14 KB

Now, go to the angular folder that we created at step 1, and copy the "dist" folder that contains the production version of the angular project:



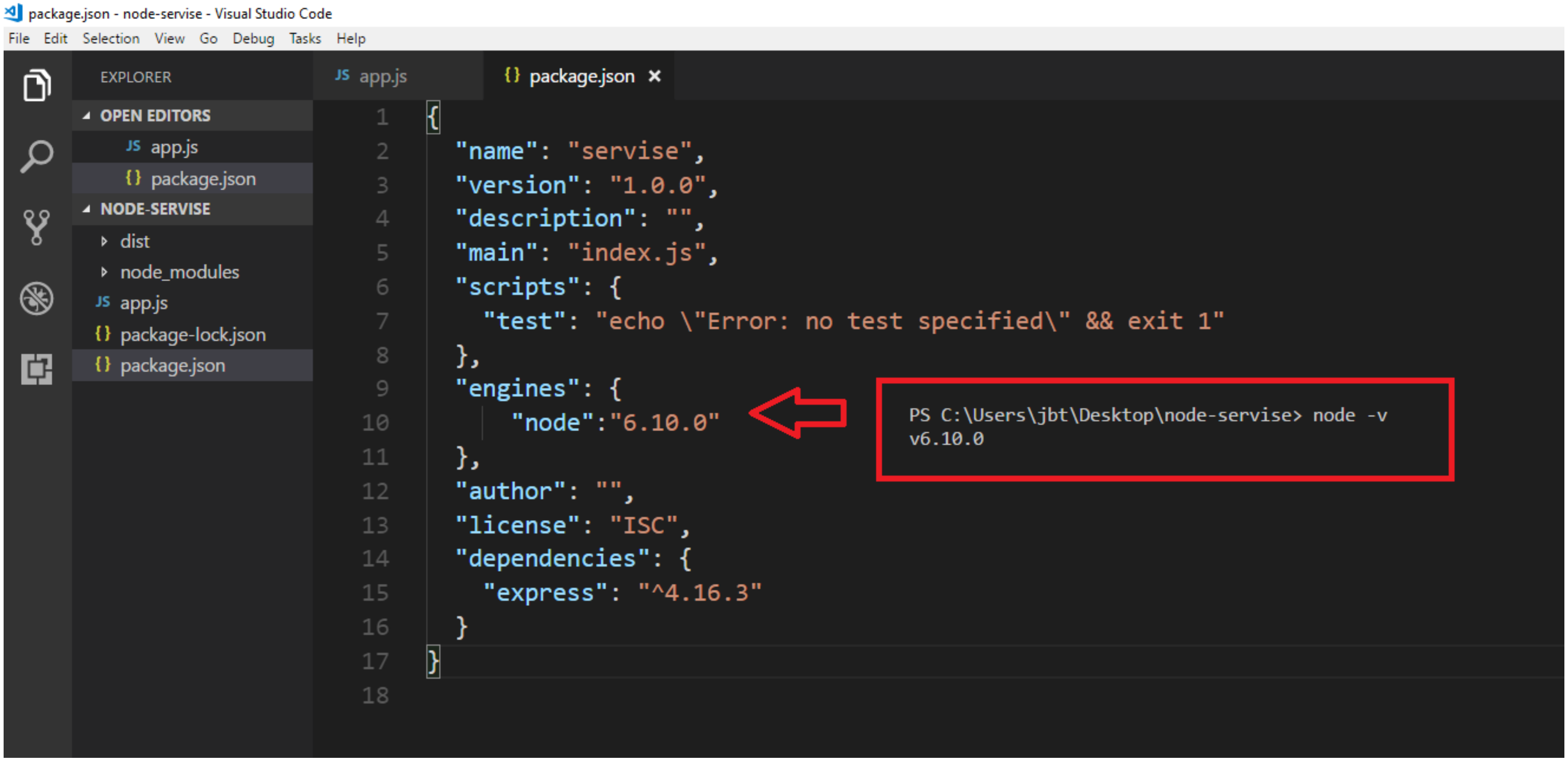
Name	Date modified	Type	Size
.git	13/04/2018 09:28	File folder	
dist	13/04/2018 09:25	File folder	
e2e	13/04/2018 09:12	File folder	
node_modules	13/04/2018 09:25	File folder	
src	13/04/2018 09:12	File folder	
.angular-cli.json	13/04/2018 09:12	JSON File	2 KB
.editorconfig	13/04/2018 09:12	Editor Config Sour...	1 KB
.gitignore	13/04/2018 09:12	Text Document	1 KB
karma.conf.js	13/04/2018 09:12	JavaScript File	1 KB
package.json	13/04/2018 09:12	JSON File	2 KB
package-lock.json	13/04/2018 09:17	JSON File	392 KB
protractor.conf.js	13/04/2018 09:12	JavaScript File	1 KB
README.md	13/04/2018 09:12	Markdown Source...	2 KB
tsconfig.json	13/04/2018 09:12	JSON File	1 KB
tslint.json	13/04/2018 09:12	JSON File	3 KB

Paste the folder into the node project's folder:



Name	Date modified	Type	Size
node_modules	13/04/2018 09:28	File folder	
app.js	13/04/2018 09:34	JavaScript File	1 KB
package.json	13/04/2018 09:28	JSON File	1 KB
package-lock.json	13/04/2018 09:28	JSON File	14 KB
dist	13/04/2018 09:35	File folder	

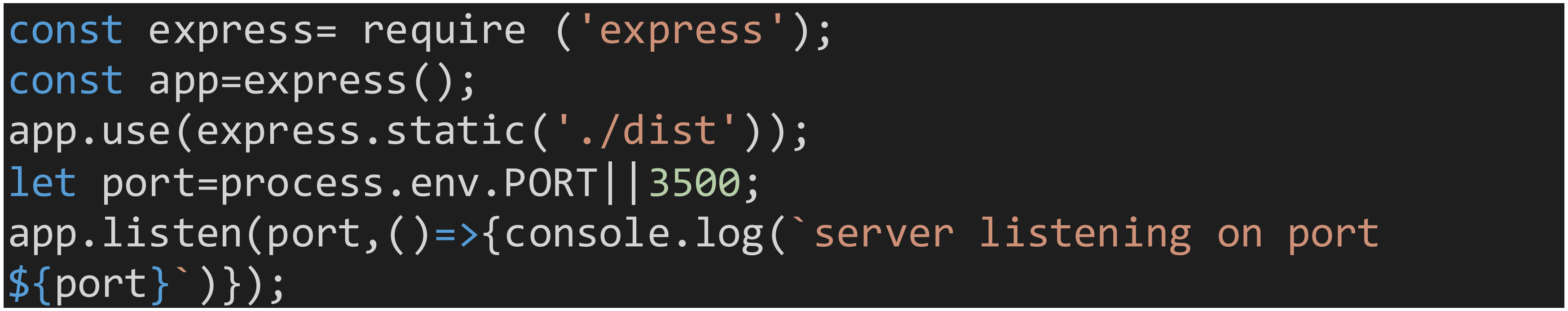
Check your local node version, and add it to the "package.config" file:



```json
{
  "name": "servise",
  "version": "1.0.0",
  "description": "",
  "main": "index.js",
  "scripts": {
    "test": "echo \"Error: no test specified\" && exit 1"
  },
  "engines": {
      "node":"6.10.0"
  },
  "author": "",
  "license": "ISC",
  "dependencies": {
    "express": "^4.16.3"
  }
}
```

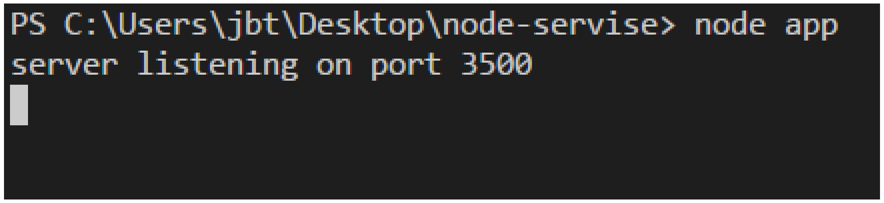
```
PS C:\Users\jbt\Desktop\node-servise> node -v
v6.10.0
```

Add a new file, named "app.js", and paste into it the following code:



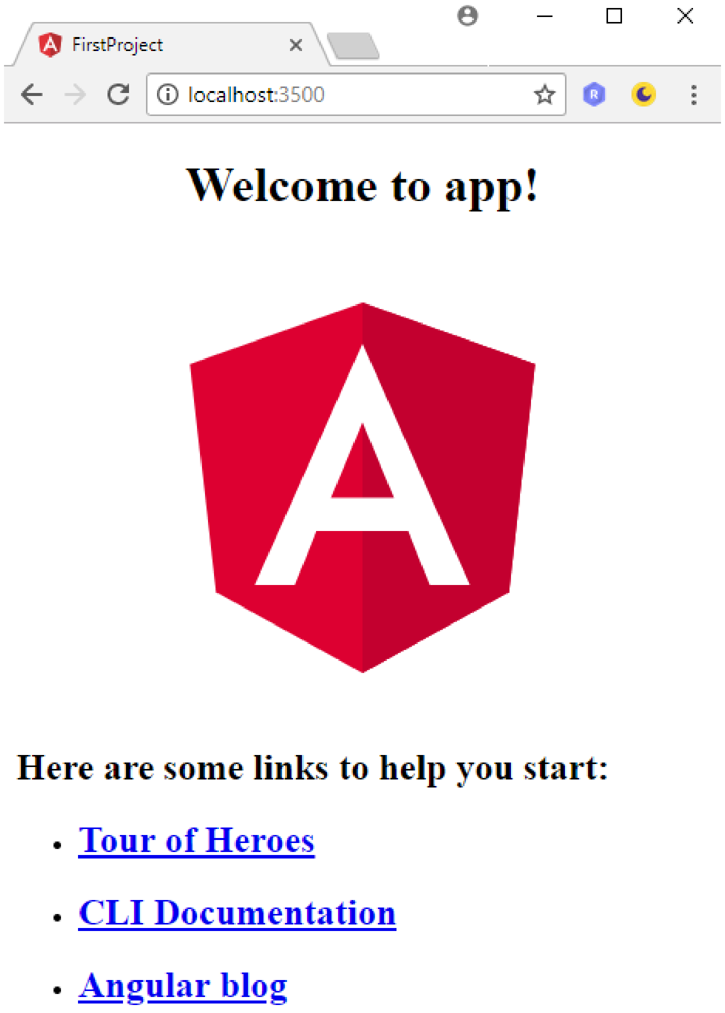
```
const express= require ('express');
const app=express();
app.use(express.static('./dist'));
let port=process.env.PORT||3500;
app.listen(port,()=>{console.log(`server listening on port
${port}`)});
```

now, you can simply run the application with the following command:



```
PS C:\Users\jbt\Desktop\node-servise> node app
server listening on port 3500
```

Open " http://localhost:3500/" in the browser:

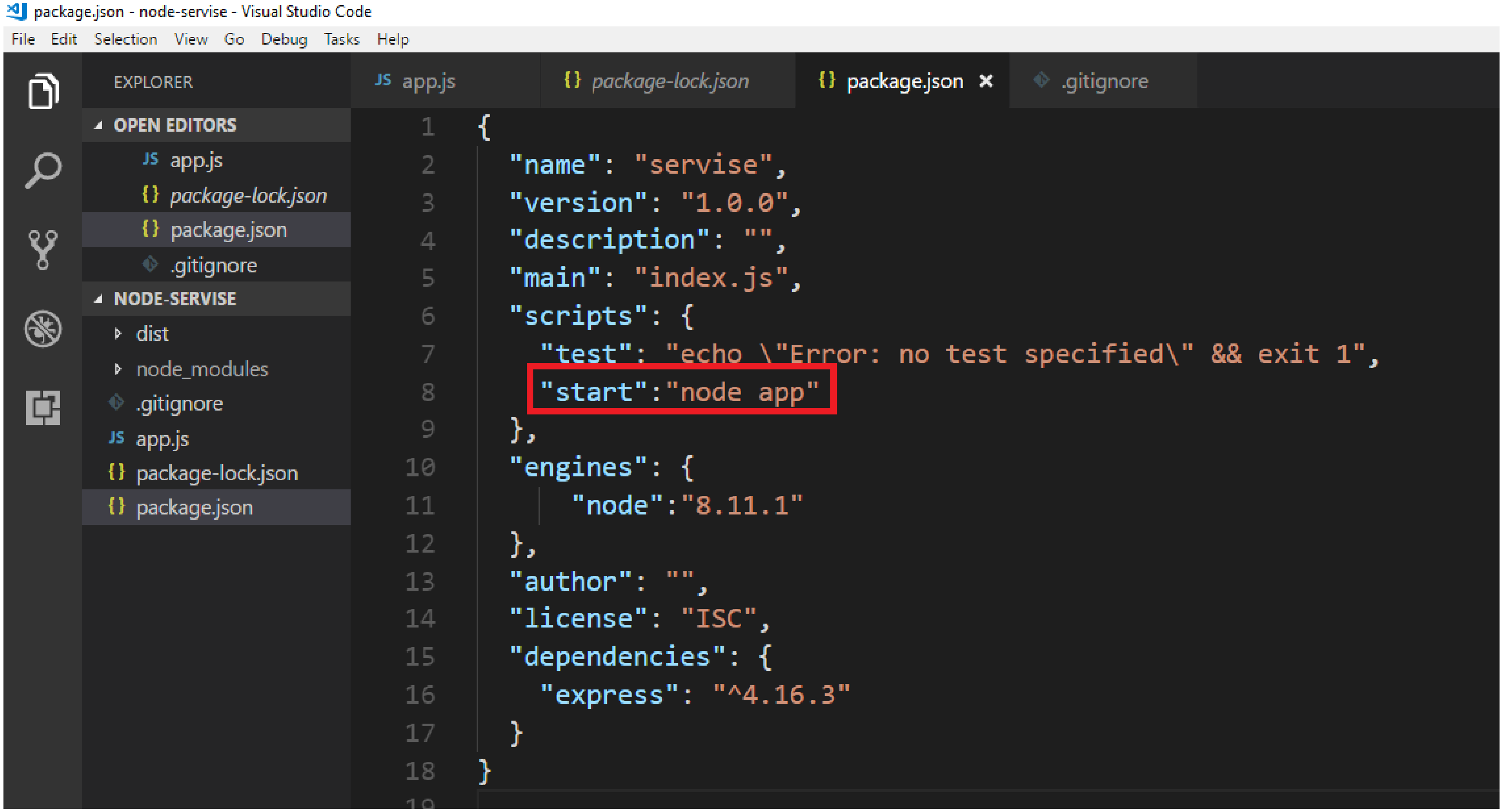


Welcome to app!

Here are some links to help you start:

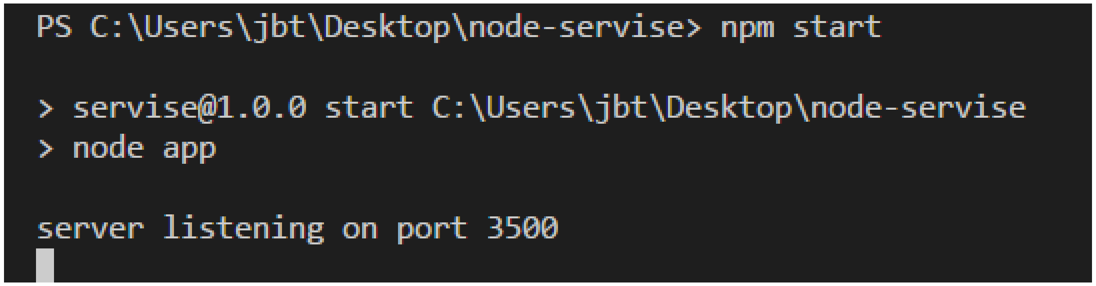
- Tour of Heroes
- CLI Documentation
- Angular blog

Add the "node app" running command to the "package.json" file, as shown in the following screenshot:



```
{
  "name": "servise",
  "version": "1.0.0",
  "description": "",
  "main": "index.js",
  "scripts": {
    "test": "echo \"Error: no test specified\" && exit 1",
    "start":"node app"
  },
  "engines": {
    "node":"8.11.1"
  },
  "author": "",
  "license": "ISC",
  "dependencies": {
    "express": "^4.16.3"
  }
}
```

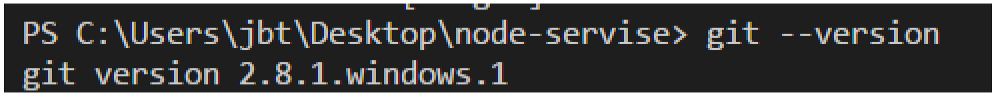
now, try to run the application with the "npm start" command, if you get the following output, your configuration are done successfully:



```
PS C:\Users\jbt\Desktop\node-servise> npm start

> servise@1.0.0 start C:\Users\jbt\Desktop\node-servise
> node app

server listening on port 3500
```

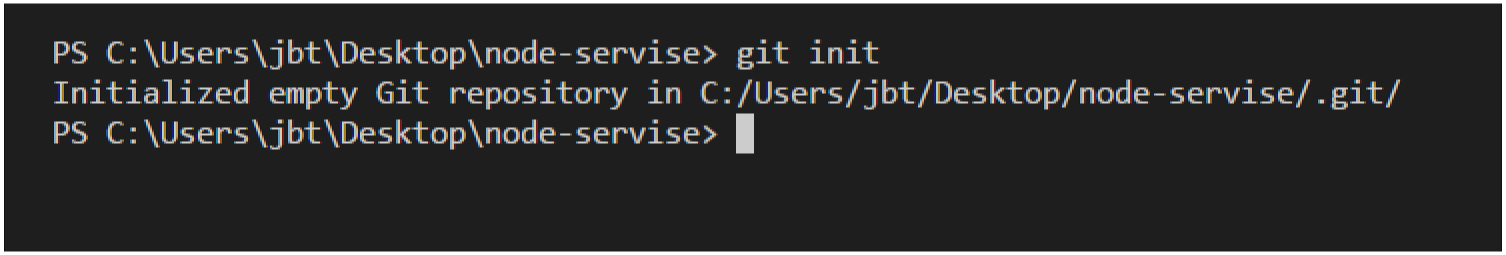
Step 3 – git & node-project

Check in the cli that you have git installed in your machine:



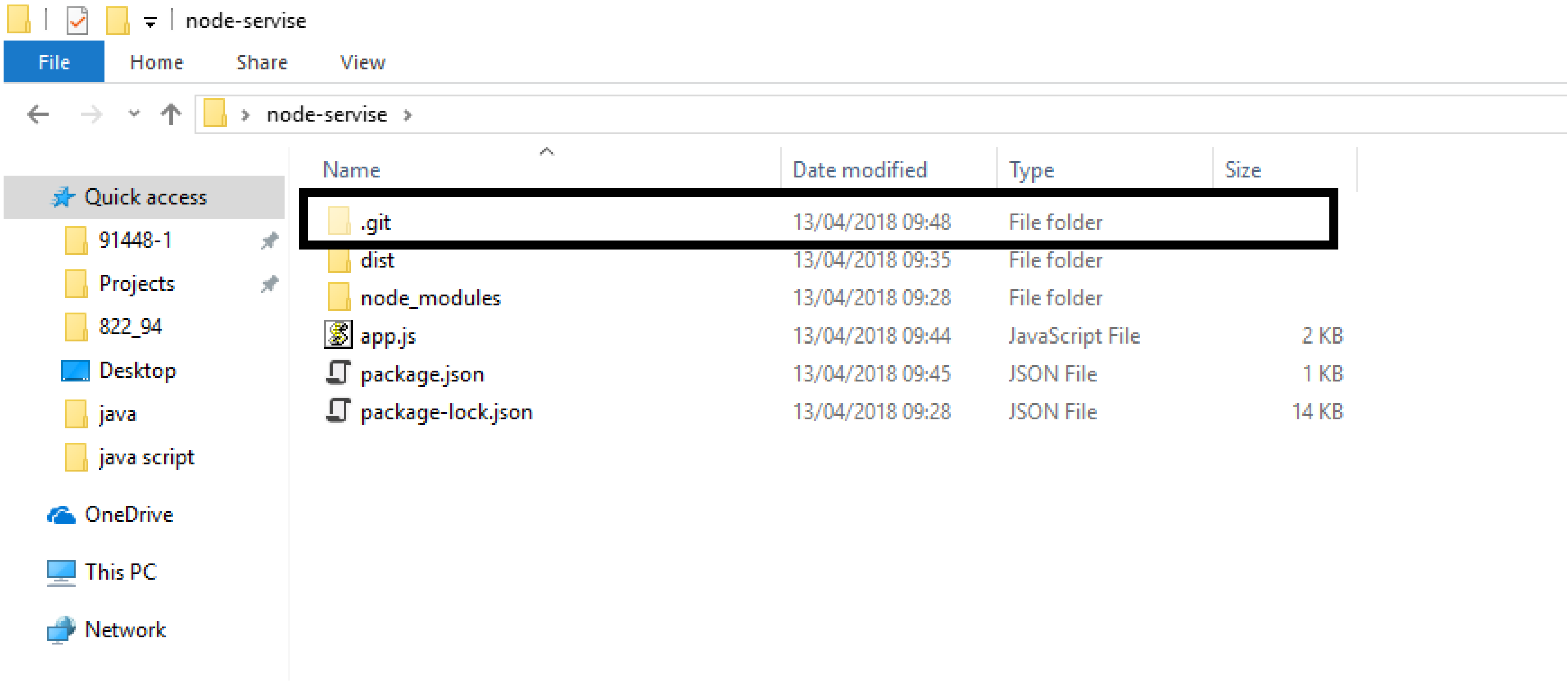
```
PS C:\Users\jbt\Desktop\node-servise> git --version
git version 2.8.1.windows.1
```

Add your node project to the git:



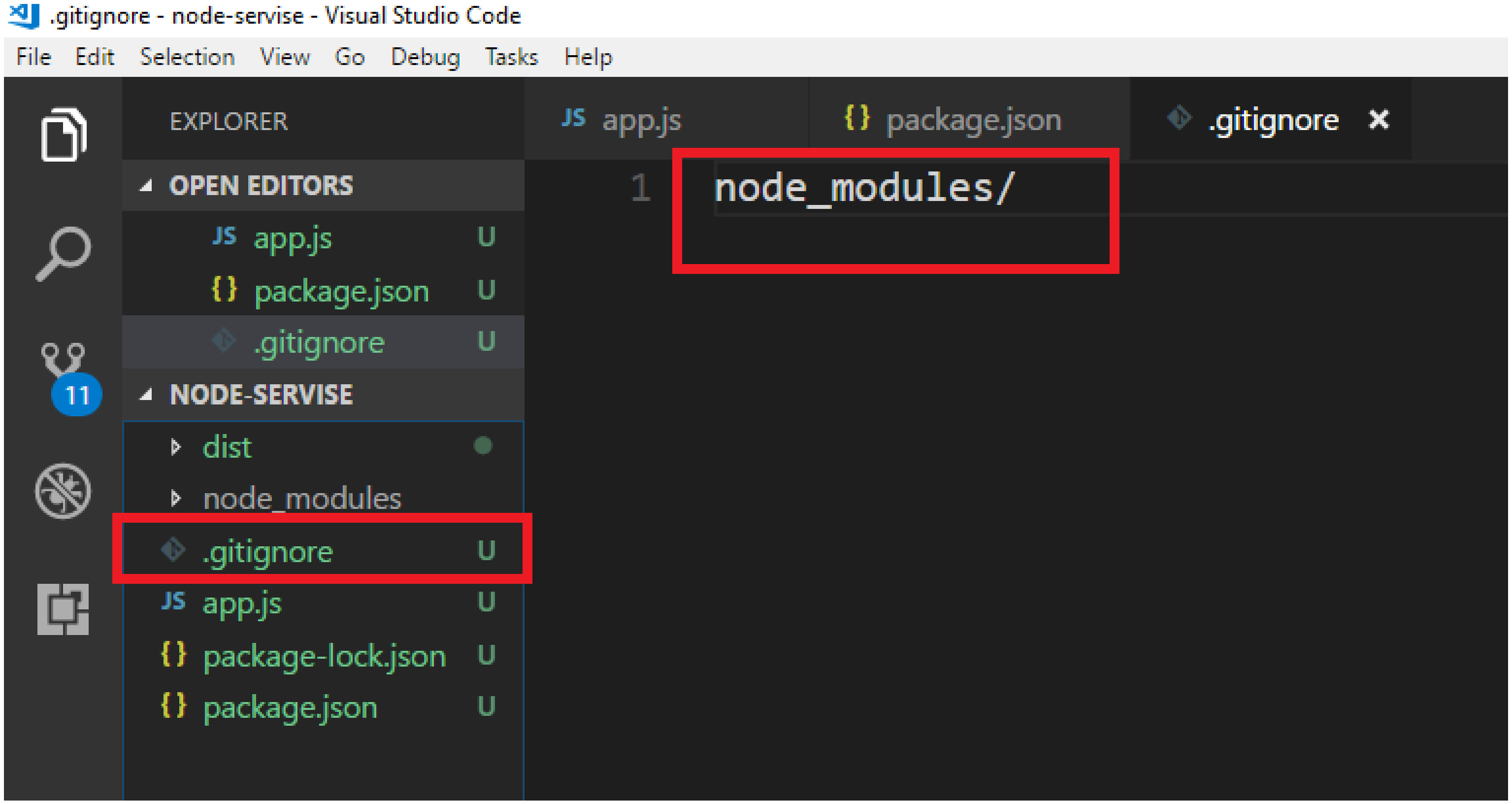
```
PS C:\Users\jbt\Desktop\node-servise> git init
Initialized empty Git repository in C:/Users/jbt/Desktop/node-servise/.git/
PS C:\Users\jbt\Desktop\node-servise>
```

Check that you have a new "git" folder in the project:



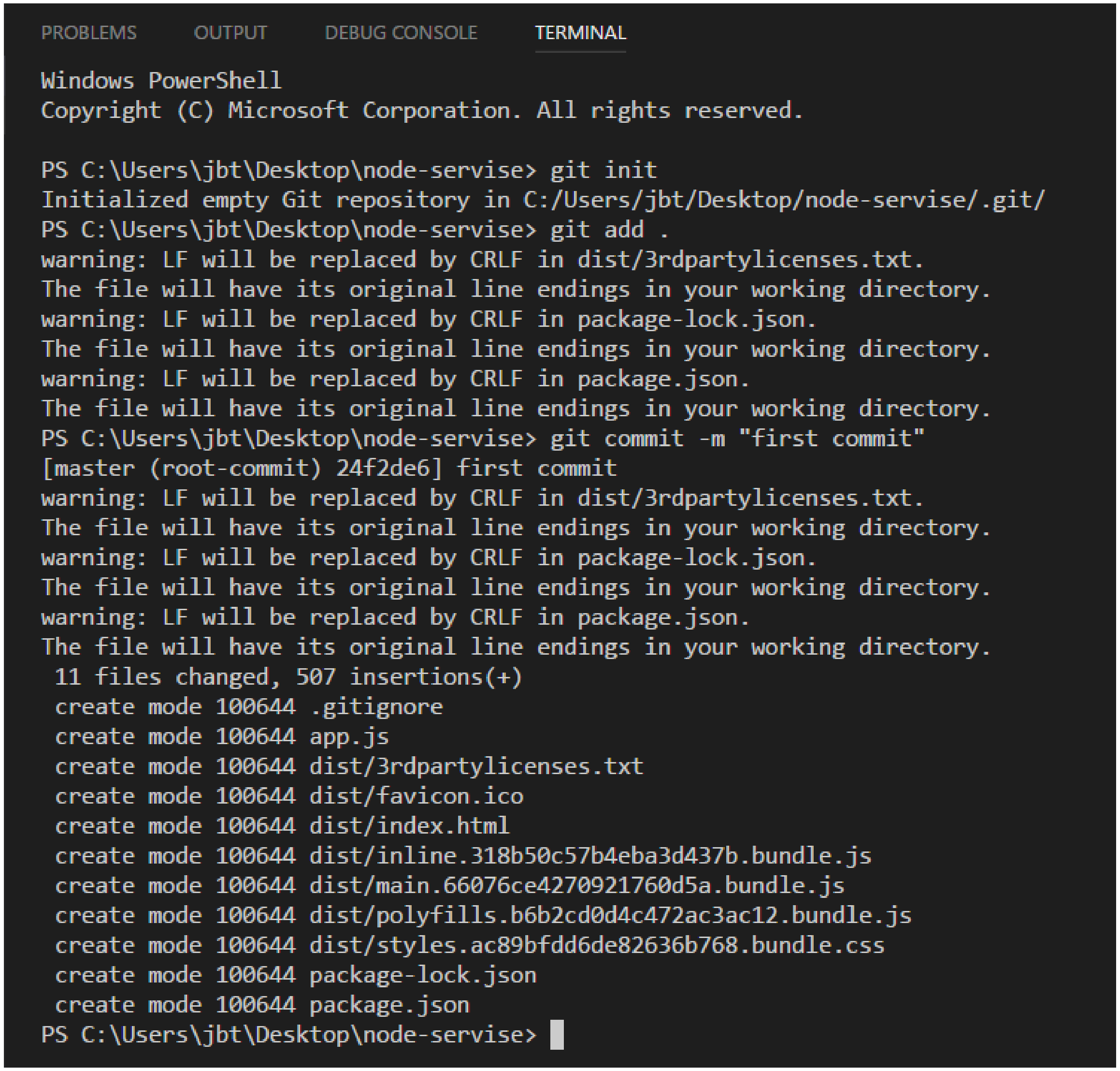
Name	Date modified	Type	Size
.git	13/04/2018 09:48	File folder	
dist	13/04/2018 09:35	File folder	
node_modules	13/04/2018 09:28	File folder	
app.js	13/04/2018 09:44	JavaScript File	2 KB
package.json	13/04/2018 09:45	JSON File	1 KB
package-lock.json	13/04/2018 09:28	JSON File	14 KB

Now, add to the node project a ".gitignore" folder, to ignore all the npm packages:



```
node_modules/
```

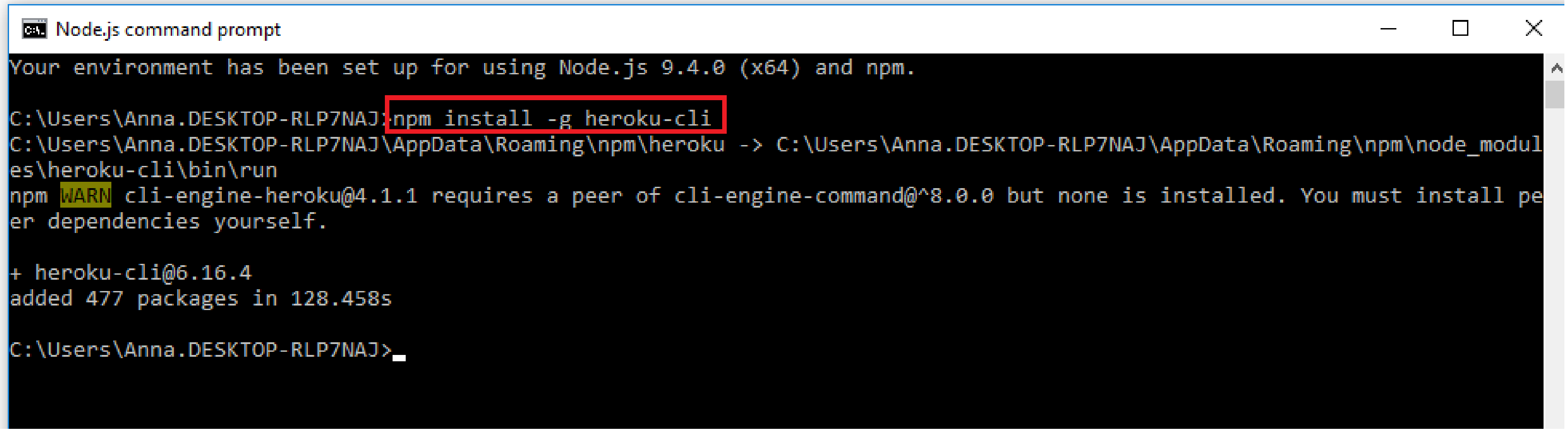
And commit all the content:



```
Windows PowerShell
Copyright (C) Microsoft Corporation. All rights reserved.

PS C:\Users\jbt\Desktop\node-servise> git init
Initialized empty Git repository in C:/Users/jbt/Desktop/node-servise/.git/
PS C:\Users\jbt\Desktop\node-servise> git add .
warning: LF will be replaced by CRLF in dist/3rdpartylicenses.txt.
The file will have its original line endings in your working directory.
warning: LF will be replaced by CRLF in package-lock.json.
The file will have its original line endings in your working directory.
warning: LF will be replaced by CRLF in package.json.
The file will have its original line endings in your working directory.
PS C:\Users\jbt\Desktop\node-servise> git commit -m "first commit"
[master (root-commit) 24f2de6] first commit
warning: LF will be replaced by CRLF in dist/3rdpartylicenses.txt.
The file will have its original line endings in your working directory.
warning: LF will be replaced by CRLF in package-lock.json.
The file will have its original line endings in your working directory.
warning: LF will be replaced by CRLF in package.json.
The file will have its original line endings in your working directory.
 11 files changed, 507 insertions(+)
 create mode 100644 .gitignore
 create mode 100644 app.js
 create mode 100644 dist/3rdpartylicenses.txt
 create mode 100644 dist/favicon.ico
 create mode 100644 dist/index.html
 create mode 100644 dist/inline.318b50c57b4eba3d437b.bundle.js
 create mode 100644 dist/main.66076ce4270921760d5a.bundle.js
 create mode 100644 dist/polyfills.b6b2cd0d4c472ac3ac12.bundle.js
 create mode 100644 dist/styles.ac89bfdd6de82636b768.bundle.css
 create mode 100644 package-lock.json
 create mode 100644 package.json
PS C:\Users\jbt\Desktop\node-servise>
```

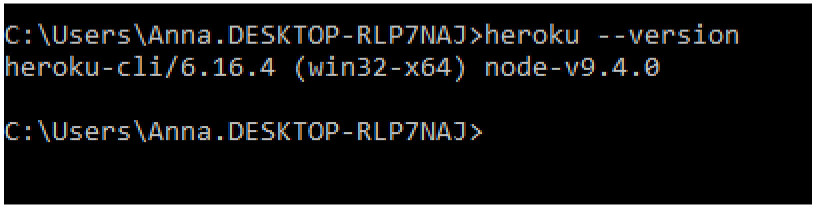
Step 4 – heroku & node-project

Install heroku-cli globally on your machine:



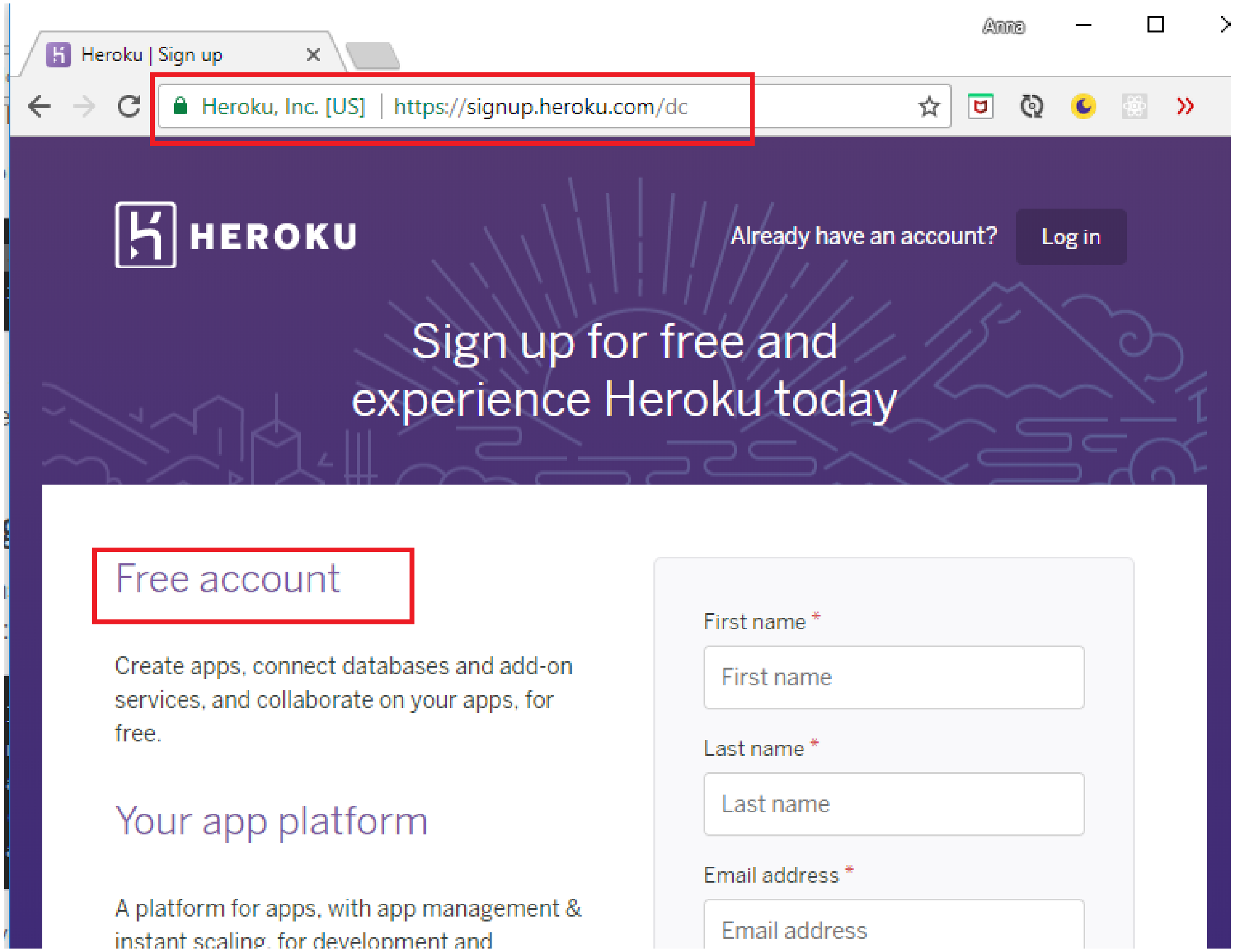
```
Your environment has been set up for using Node.js 9.4.0 (x64) and npm.

C:\Users\Anna.DESKTOP-RLP7NAJ>npm install -g heroku-cli
C:\Users\Anna.DESKTOP-RLP7NAJ\AppData\Roaming\npm\heroku -> C:\Users\Anna.DESKTOP-RLP7NAJ\AppData\Roaming\npm\node_modul
es\heroku-cli\bin\run
npm WARN cli-engine-heroku@4.1.1 requires a peer of cli-engine-command@^8.0.0 but none is installed. You must install pe
er dependencies yourself.

+ heroku-cli@6.16.4
added 477 packages in 128.458s

C:\Users\Anna.DESKTOP-RLP7NAJ>_
```

To confirm the installation check the version:



```
C:\Users\Anna.DESKTOP-RLP7NAJ>heroku --version
heroku-cli/6.16.4 (win32-x64) node-v9.4.0

C:\Users\Anna.DESKTOP-RLP7NAJ>
```

Open an account in heroku:



Sign up for free and experience Heroku today

Free account

Create apps, connect databases and add-on services, and collaborate on your apps, for free.

Your app platform

A platform for apps, with app management & instant scaling, for development and

First name *

First name

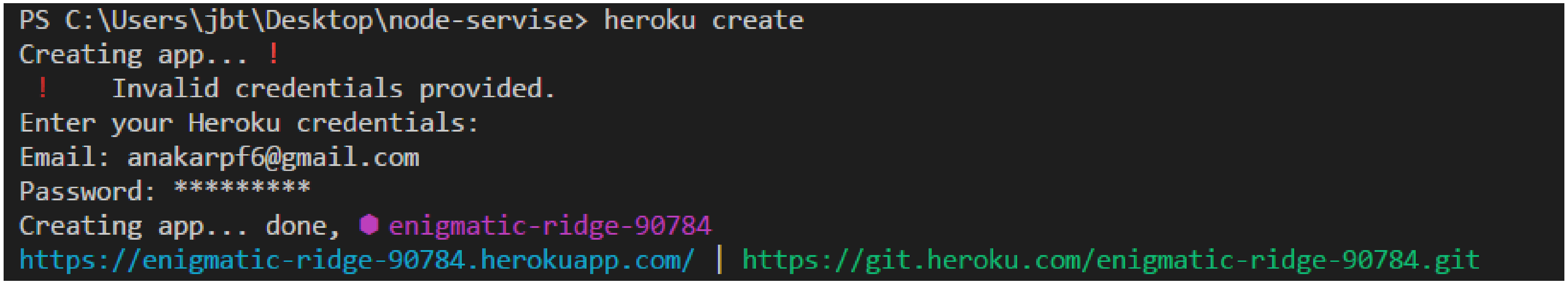
Last name *

Last name

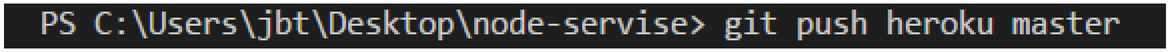
Email address *

Email address

Upload the node project to heroku, with the following commands:

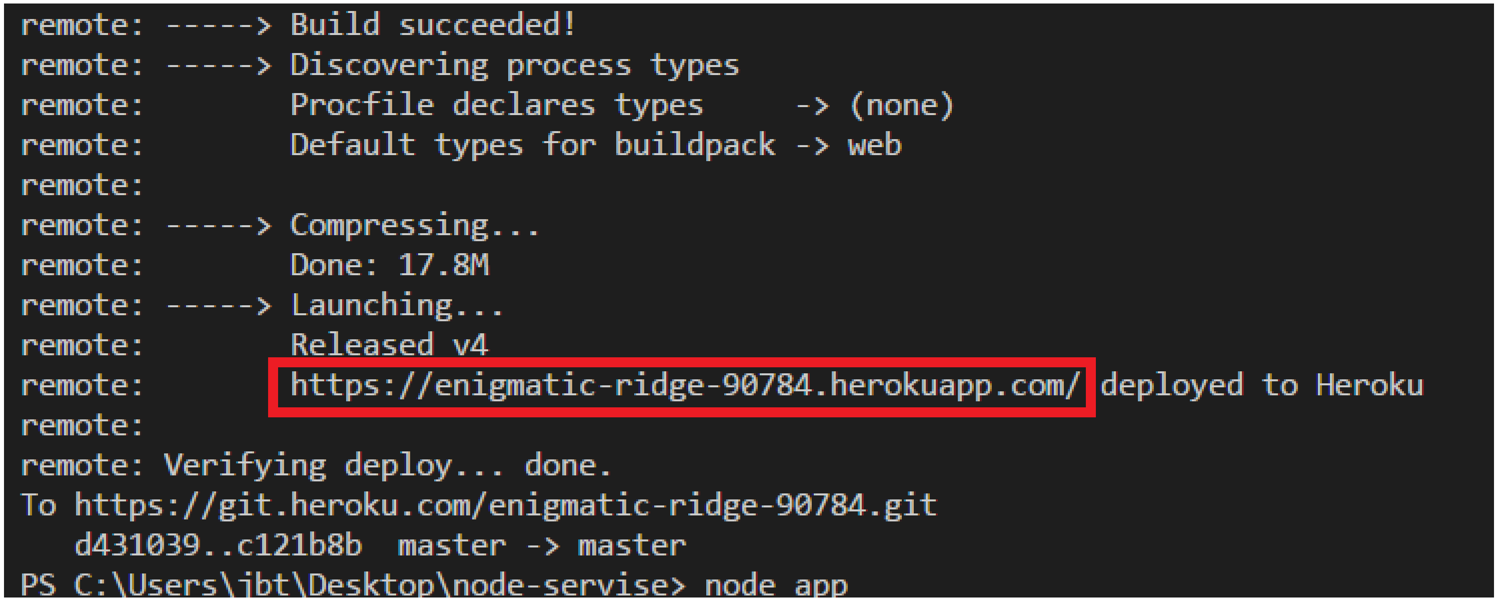


```
PS C:\Users\jbt\Desktop\node-servise> heroku create
Creating app... !
 !    Invalid credentials provided.
Enter your Heroku credentials:
Email: anakarpf6@gmail.com
Password: *********
Creating app... done, ⬢ enigmatic-ridge-90784
https://enigmatic-ridge-90784.herokuapp.com/ | https://git.heroku.com/enigmatic-ridge-90784.git
```



```
 PS C:\Users\jbt\Desktop\node-servise> git push heroku master
```

At the end, you will get the link of your new site:



```
remote: -----> Build succeeded!
remote: -----> Discovering process types
remote:        Procfile declares types     -> (none)
remote:        Default types for buildpack -> web
remote:
remote: -----> Compressing...
remote:        Done: 17.8M
remote: -----> Launching...
remote:        Released v4
remote:        https://enigmatic-ridge-90784.herokuapp.com/ deployed to Heroku
remote:
remote: Verifying deploy... done.
To https://git.heroku.com/enigmatic-ridge-90784.git
   d431039..c121b8b  master -> master
PS C:\Users\jbt\Desktop\node-servise> node app
```