

Running a Node App and Installing 3rd-party Modules

If you have node installed, you can run a JavaScript file (your "Node app") like so:

```
$ node app.js
```

Often you will want to use 3rd-party modules from npm, and to do that you will need to create a `package.json` configuration file which will store information on which modules you need installed for your Node app, as well as other information for more complex apps, like which script will be used for testing, deploying your app, etc.

Creating a new configuration file with npm

If you are creating your own app from scratch, first create an empty `app.js` file in your directory. Then, you can easily create a `package.json` file using the command:

```
$ npm init
```

This will guide you through the steps of creating your configuration file. If you want to just use the default values, you can **just press enter** through all of the prompts.

Installing and using 3rd-party modules from npm

npm has many 3rd-party modules available to use in your Node app (<https://www.npmjs.com/>).

If you want to use one of them, you just need the name of the module, and then you can install them using the following command:

```
$ npm install <module_name>
```

This will do two things - it will download and install the module into the `node_modules` folder, and it will add an entry into your `package.json` file.

Reinstalling modules from a `package.json` file (and running our lecture notes)

The `node_modules` folder **should not be committed to your git repo**, since you can always reinstall all of the modules indicated in your `package.json` file using the command:

```
$ npm install
```

This means you only need to commit your configuration file, since all modules and dependencies will be installed when you run the above command. You can then run Node as usual with your app.