SIT 725 Prac 2 Express

Contents

- Basics of Nodejs
- Installing Nodejs on Ubuntu
- Understanding Express
- Bootstrapping Express App
- Understanding REST
- Creating REST API's
- Questions

Basics of Nodejs

Node.js is a cross-platform, open-source back-end JavaScript runtime environment that uses the V8 engine to execute JavaScript code outside of a web browser. Node.js allows developers to utilise JavaScript to create command-line tools and server-side scripting, which involves running scripts on the server to generate dynamic web page content before sending the page to the user's browser.

Installing Nodejs on Ubuntu

Nodejs is prerequisite for any express based application In Order to install nodejs on an ubuntu based environment you can follow the guide from digital ocean.

Personally i prefer to always install node js using NVM as it provide you the ability to add multiple different versions of Node Js on your environment and easily switch between them when needed.

Understanding Express

Express is a Node.js web application framework that offers a comprehensive range of functionality for both web and mobile apps.

In Order to install express to your project you can use :

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

Bootstrapping Express App

In order to start with bootstrapping the express app you first need to create a github repo which we discussed in last practical, please follow the same instructions which we did during the practical.

You would also need to have nodejs installed before this step.

Once you have created the github repo, clone the repo in your system

```
$ git clone <link to repo>
```

After you have done that move into the folder

```
$ cd sit725-2021-t2-prac2
```

Bootstrapping Express App Cont ...

Now that you are inside the folder we can start bootstrapping the express app

First step is to initiate a node environment inside your repo for doing that run the command

```
$ npm init
```

Once you have run the command it would ask you a few question which you can answer and it would create a file called package.json inside your application.

Next we add express to our application, for that we run the command

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

Bootstrapping Express App Cont ...

Doing this create a folder called node_modules in your application and will also update the package.json.

Now we are ready to create our first express app.

Next we create a file called server.js in our application, this file will be responsible for creating our server and handling all the process for our application.

Your server.js should look like this:

```
var express = require("express")
var app = express()
var port = process.env.port || 3000;

app.listen(port,()=>{
    console.log("App listening to: "+port)
})
```

Bootstrapping Express App Cont ...

Once we have done this we need to update our package.json In the scripts section of our package.json we add a new script to start our application. Your package.json should look like this

```
{
  "name": "sit725-2021-t2-prac",
  "version": "1.0.0",
  "description": "sit 725 T2 week 2 prac",
  "main": "server.js",
  "scripts": {
    "start": "node server.js"
  },
  "repository": {
    "type": "git",
    "url": "git+https://github.com/ChoudharyNavit22/sit725-2021-t2-prac2.git"
  },
  "author": "Navit Choudhary",
  "license": "MIT",
  "bugs": {
    "url": "https://github.com/ChoudharyNavit22/sit725-2021-t2-prac2/issues"
  },
  "homepage": "https://github.com/ChoudharyNavit22/sit725-2021-t2-prac2#readme",
  "dependencies": {
    "express": "^4.17.1"
  }
}
```

Bootstrapping Express App Cont ...

Please make sure you only update the script section to look the same as example rest should not be changed as they are specific to your own repo the example is from my personal repo so it has the git url and author names from my repo

Once you have done all this you are ready to run your first express app.

In order to do that in your terminal run the command

```
$ npm run start
```

And you should have something printed in your terminal saying that the application is running on port 3000, Now you have your first running Express App. **Please don't forget to commit your code at this step.**

Understanding REST

REST (representational state transfer) is a software architectural style that was developed to guide the design and development of the World Wide Web's architecture. REST establishes a set of guidelines for how an Internet-scale distributed hypermedia system, such as the Web, should be designed. The REST architectural style emphasises scalability of component interactions, uniform interfaces, component deployment independence, and the establishment of a layered architecture to enable caching components to reduce user-perceived latency, enforce security, and encapsulate legacy systems. REST is a generally established set of rules for constructing stateless, dependable web services that has been used throughout the IT industry.

Understanding REST Cont ...

When using REST there are 4 main type of endpoint we create

- POST - to create data in our application
- GET - to retrieve the created data in our application
- PUT - to update/modify the created data in our application
- DELETE - to delete created data in our application

Thanks