

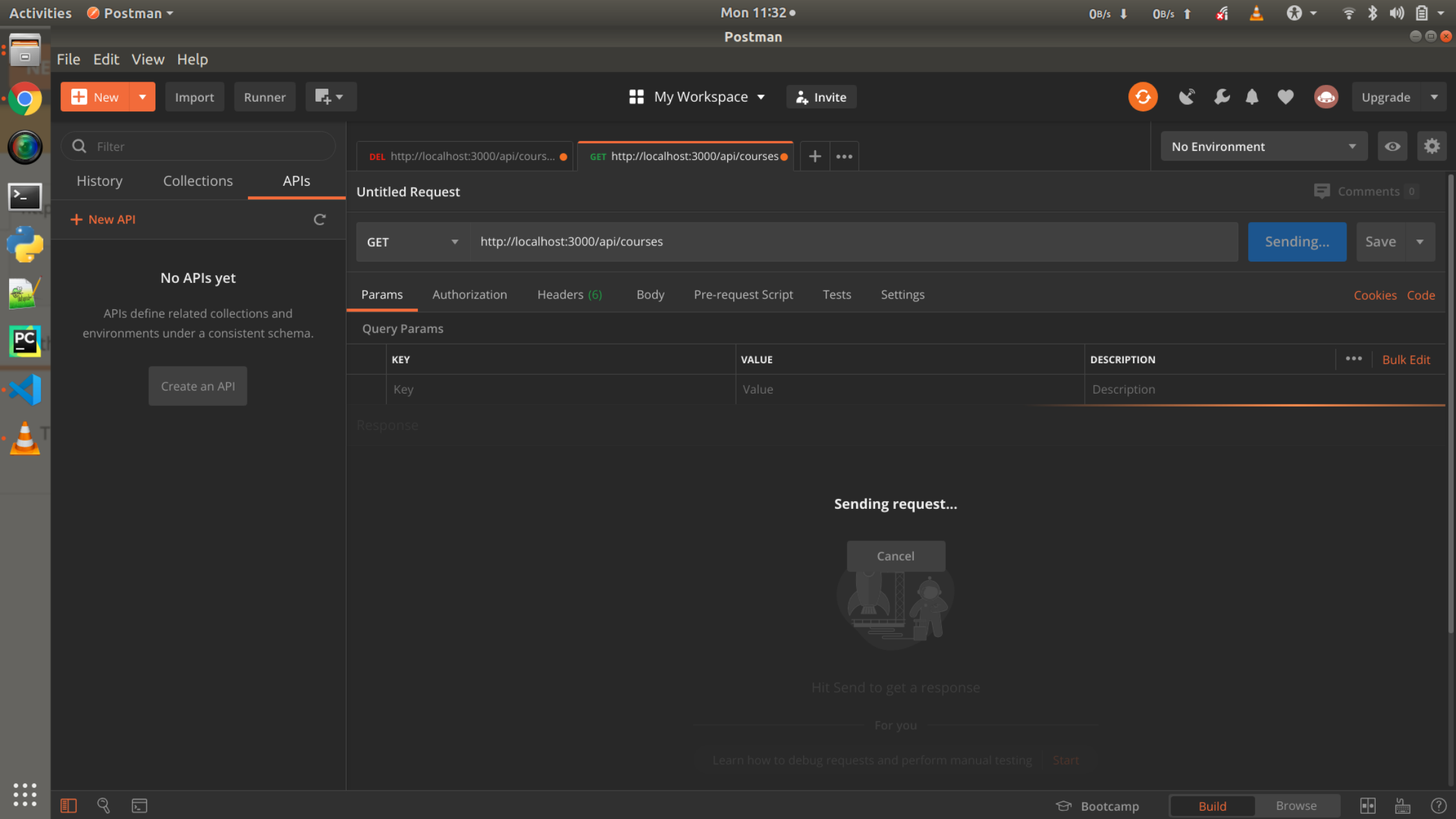
Building Custom Middleware



Mosh Hamedani
programmingwithmosh.com

> 14. Express- Advanced Topics Recap

> OUTLINE



File Edit View Help

My Workspace

Invite

Upgrade

Filter

DEL http://localhost:3000/api/cours...

GET http://localhost:3000/api/courses

No Environment

History

Collections

APIs

+ New API

No APIs yet

APIs define related collections and environments under a consistent schema.

Create an API

Untitled Request

Comments 0

GET

http://localhost:3000/api/courses

Sending...

Save

Params

Authorization

Headers (6)

Body

Pre-request Script

Tests

Settings

Cookies Code

Query Params

KEY

VALUE

DESCRIPTION

...

Bulk Edit

Key

Value

Description

Response

Sending request...

Cancel

Hit Send to get a response

For you

Learn how to debug requests and perform manual testing

Start

Bootcamp

Build

Browse

ActivitiesVisual Studio Code

index.js - Applications - Visual Studio Code

0B/s ↓ 204B/s ↑

FileEditSelectionViewGoRunTerminalHelp

EXPLORER

OPEN EDITORS

APPLICATIONS

01. Introduction

02. Middleware

03. Creating Custom Middleware

JS index.js

04. Built-In Middleware

05. Third-party Middleware

06. Environments

07. Configuration

08. Debugging

09. Templating Engines

10. Database Integration

11. Authentication

12. Structuring Express Applications

13. Project- Restructure the App

14. Express- Advanced Topics Recap

OUTLINE

JS index.js

03. Creating Custom Middleware > JS index.js > [0] courses

```
0
1
2
3
4
5
6
7 app.use(function(req, res, next){
8   console.log('Logging...');
9   //next(); // This is used to pass control to the next middleware function.
10 });
11
12 const courses = [
13   { id: 1, name: 'course1' },
14   { id: 2, name: 'course2' },
15   { id: 3, name: 'course3' }
16 ];
17
18 app.get('/', (req, res) => {
19   res.send('Hello World !!! Its corona-time');
20 });
21 app.get('/api/courses', (req, res) => {
22   res.send(courses);
23 });
24
25 Complexity is 4 Everything is cool!
26 app.get('/api/courses/:id', (req, res) => {
27   const course = courses.find(c => c.id === parseInt(req.params.id));
28   if (!course) return res.status(404).send('The course with the given ID was not found.');
```

DEBUG CONSOLE

PROBLEMS

OUTPUT

TERMINAL

1: node

x.js

[nodemon] 2.0.2

[nodemon] to restart at any time, enter `rs`

[nodemon] watching dir(s): *.*

[nodemon] watching extensions: js,mjs,json

[nodemon] starting `node index.js`

Listening to port 3000

Logging....

master* 0 0 0 ✓ javascript | ✓ index.js

Ln 12, Col 18 Spaces: 4 UTF-8 LF Babel JavaScript JavaScript Standard Style

> 14. Express- Advanced Topics Recap

+

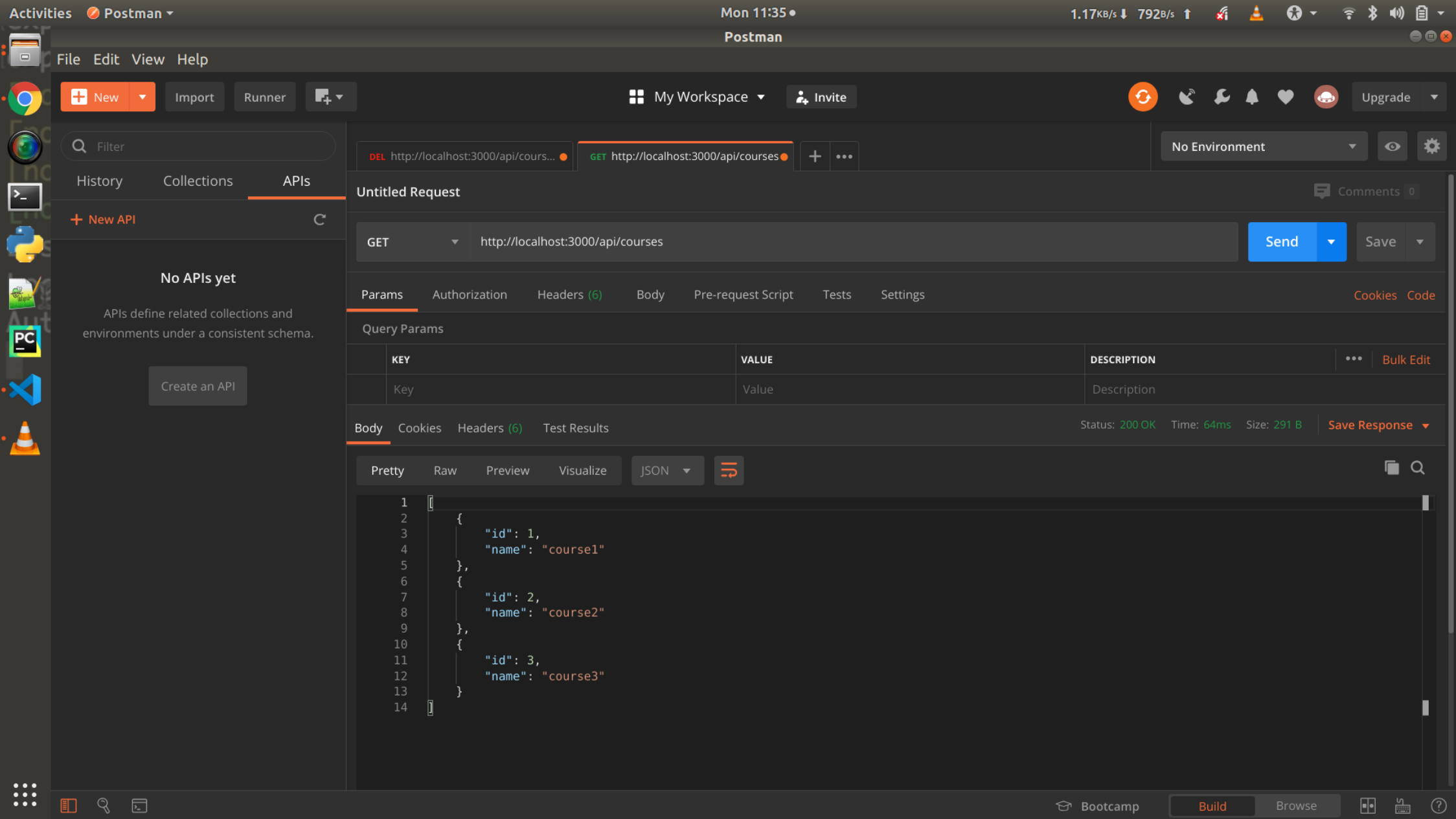
☐

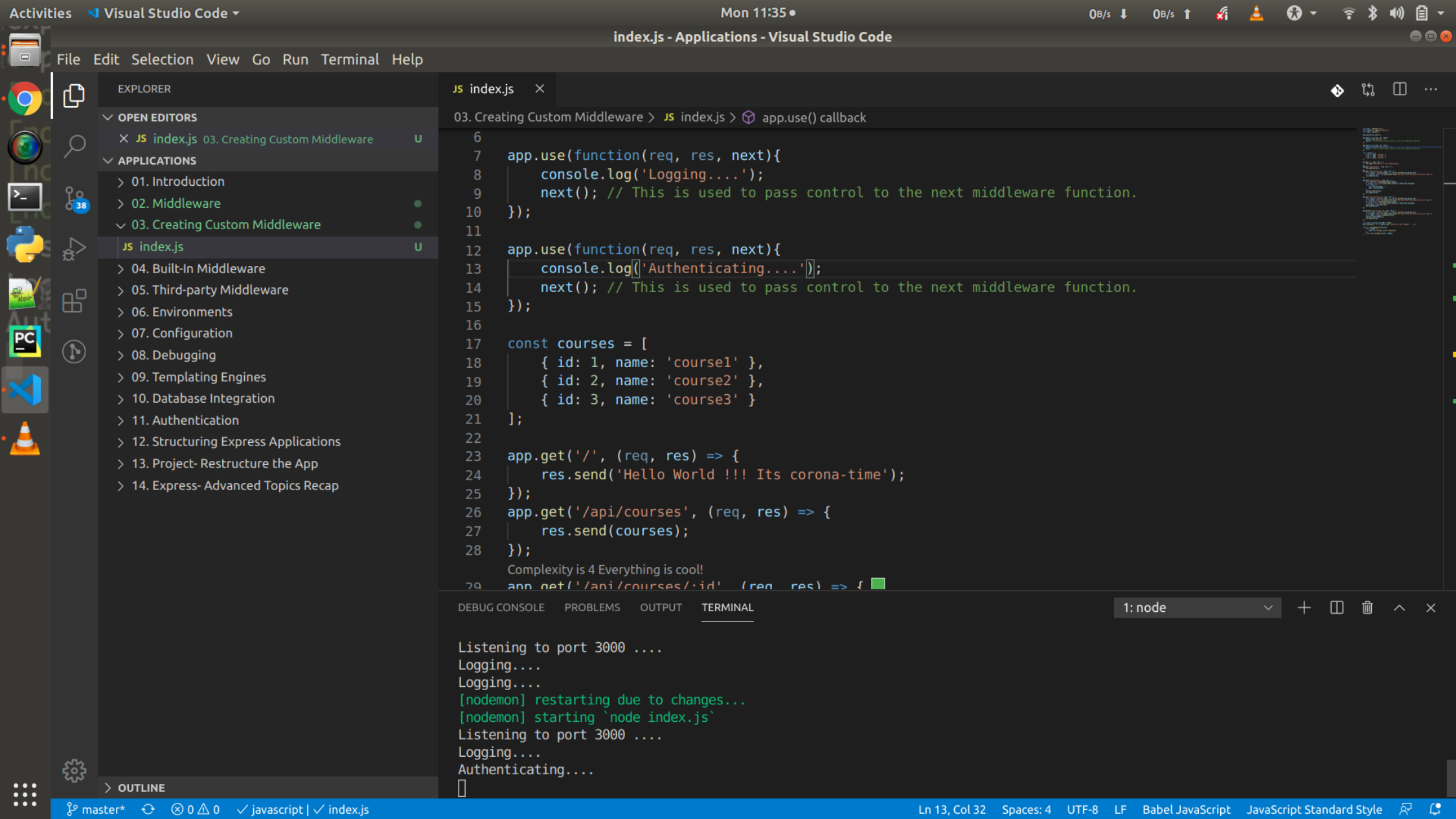
🗑

^

×







```
Mon 11:35 •
index.js - Applications - Visual Studio Code
File Edit Selection View Go Run Terminal Help
EXPLORER
OPEN EDITORS
  X JS index.js 03. Creating Custom Middleware U
APPLICATIONS
  > 01. Introduction
  > 02. Middleware
  v 03. Creating Custom Middleware
    JS index.js U
      > 04. Built-In Middleware
      > 05. Third-party Middleware
      > 06. Environments
      > 07. Configuration
      > 08. Debugging
      > 09. Templating Engines
      > 10. Database Integration
      > 11. Authentication
      > 12. Structuring Express Applications
      > 13. Project- Restructure the App
      > 14. Express- Advanced Topics Recap
JS index.js
6
7 app.use(function(req, res, next){
8   console.log('Logging...');
9   next(); // This is used to pass control to the next middleware function.
10 });
11
12 app.use(function(req, res, next){
13   console.log('Authenticating...');
14   next(); // This is used to pass control to the next middleware function.
15 });
16
17 const courses = [
18   { id: 1, name: 'course1' },
19   { id: 2, name: 'course2' },
20   { id: 3, name: 'course3' }
21 ];
22
23 app.get('/', (req, res) => {
24   res.send('Hello World !!! Its corona-time');
25 });
26 app.get('/api/courses', (req, res) => {
27   res.send(courses);
28 });
29 Complexity is 4 Everything is cool!
30 app.get('/api/courses/:id' (req, res) => {
DEBUG CONSOLE PROBLEMS OUTPUT TERMINAL
1: node
Listening to port 3000 ....
Logging...
Logging...
[nodemon] restarting due to changes...
[nodemon] starting `node index.js`
Listening to port 3000 ....
Logging...
Authenticating...
[]
Ln 13, Col 32 Spaces: 4 UTF-8 LF Babel JavaScript JavaScript Standard Style
```

ActivitiesVisual Studio Code

Mon 11:49

0B/s0B/s

index.js - Applications - Visual Studio Code

FileEditSelectionViewGoRunTerminalHelp

EXPLORER

OPEN EDITORS

APPLICATIONS

OUTLINE

JS index.js03. Creating Custom Middleware

JS logger.js03. Creating Custom Middleware

01. Introduction

02. Middleware

03. Creating Custom Middleware

JS index.js

JS logger.js

04. Built-In Middleware

05. Third-party Middleware

06. Environments

07. Configuration

08. Debugging

09. Templating Engines

10. Database Integration

11. Authentication

12. Structuring Express Applications

13. Project- Restructure the App

14. Express- Advanced Topics Recap

JS index.js

03. Creating Custom Middleware > JS index.js > ...

1const Joi = require('joi');

2const express = require('express');

3const app = express();

4const logger = require('./logger')

5

6app.use(express.json());

7

8app.use(logger.log);

9app.use(logger.authenticate);

10

11// app.use(function(req, res, next){

12// console.log('Logging...');

13// next(); // This is used to pass control to the next middleware function.

14// });

15

16// app.use(function(req, res, next){

17// console.log('Authenticating...');

18// next();

19// });

20// // Middleware functions are called in sequence

21

22const courses = [

23 { id: 1, name: 'course1' },

24 { id: 2, name: 'course2' },

25 { id: 3, name: 'course3' }

DEBUG CONSOLEPROBLEMSOUTPUTTERMINAL

1: node

at Function.Module._load (module.js:497:3)

at Function.Module.runMain (module.js:693:10)

at startup (bootstrap_node.js:188:16)

at bootstrap_node.js:609:3

[nodemon] app crashed - waiting for file changes before starting...

[nodemon] restarting due to changes...

[nodemon] starting `node index.js`

Listening to port 3000

master*0 0 0javascript | index.js

Ln 8, Col 20Spaces: 4UTF-8LFBabel JavaScriptJavaScript Standard Style

ActivitiesVisual Studio Code

Mon 11:49 •0B/s ↓0B/s ↑

logger.js - Applications - Visual Studio Code

FileEditSelectionViewGoRunTerminalHelp

EXPLORER

OPEN EDITORS

APPLICATIONS

OUTLINE

JS index.js03. Creating Custom MiddlewareU

JS logger.js03. Creating Custom MiddlewareU

01. Introduction

02. Middleware

03. Creating Custom Middleware

JS index.jsU

JS logger.jsU

04. Built-In Middleware

05. Third-party Middleware

06. Environments

07. Configuration

08. Debugging

09. Templating Engines

10. Database Integration

11. Authentication

12. Structuring Express Applications

13. Project- Restructure the App

14. Express- Advanced Topics Recap

JS index.jsJS logger.js

03. Creating Custom Middleware > JS logger.js > authenticate

1function log(req, res, next){

2 console.log('Logging...');

3 next();

4}

5

6function authenticate(req, res, next){

7 console.log('Authenticating...');

8 next();

9}

10

11module.exports = {log, authenticate};

DEBUG CONSOLEPROBLEMSOUTPUTTERMINAL

1: node

at Function.Module._load (module.js:497:3)

at Function.Module.runMain (module.js:693:10)

at startup (bootstrap_node.js:188:16)

at bootstrap_node.js:609:3

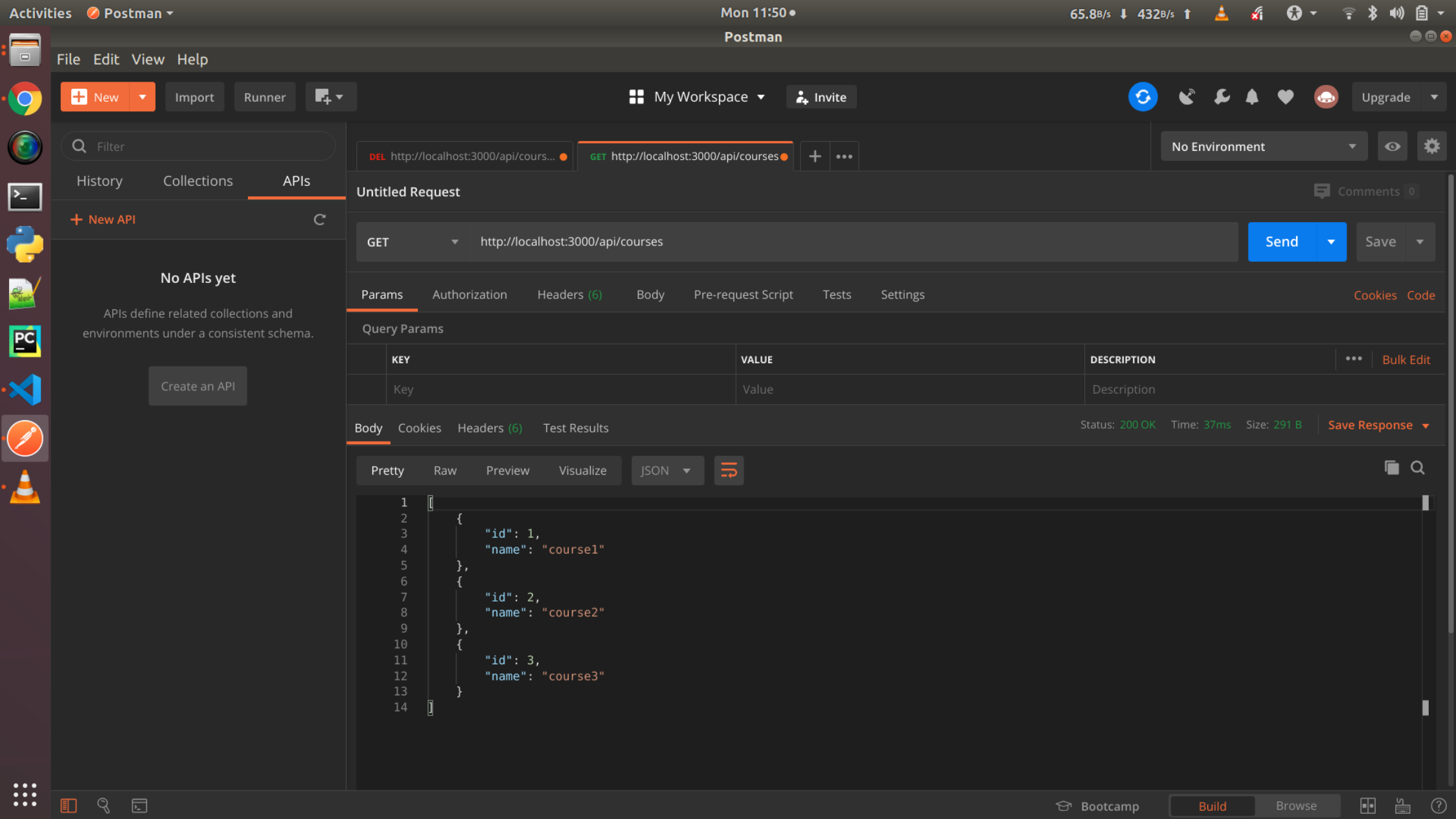
[nodemon] app crashed - waiting for file changes before starting...

[nodemon] restarting due to changes...

[nodemon] starting `node index.js`








Listening to port 3000

master*0 0 0javascript | logger.jsLn 6, Col 37Spaces: 4UTF-8LFBabel JavaScriptJavaScript Standard Style



	KEY	VALUE	DESCRIPTION	...	Bulk Edit
	Key	Value	Description		

```
1 [
2   {
3     "id": 1,
4     "name": "course1"
5   },
6   {
7     "id": 2,
8     "name": "course2"
9   },
10  {
11    "id": 3,
12    "name": "course3"
13  }
14 ]
```

```
22 const courses = [
23   { id: 1, name: 'course1' },
24   { id: 2, name: 'course2' },
25   { id: 3, name: 'course3' }
```

1

Ln 8, Col 20 Spaces: 4 UTF-8 LF Babel JavaScript JavaScript Standard Style