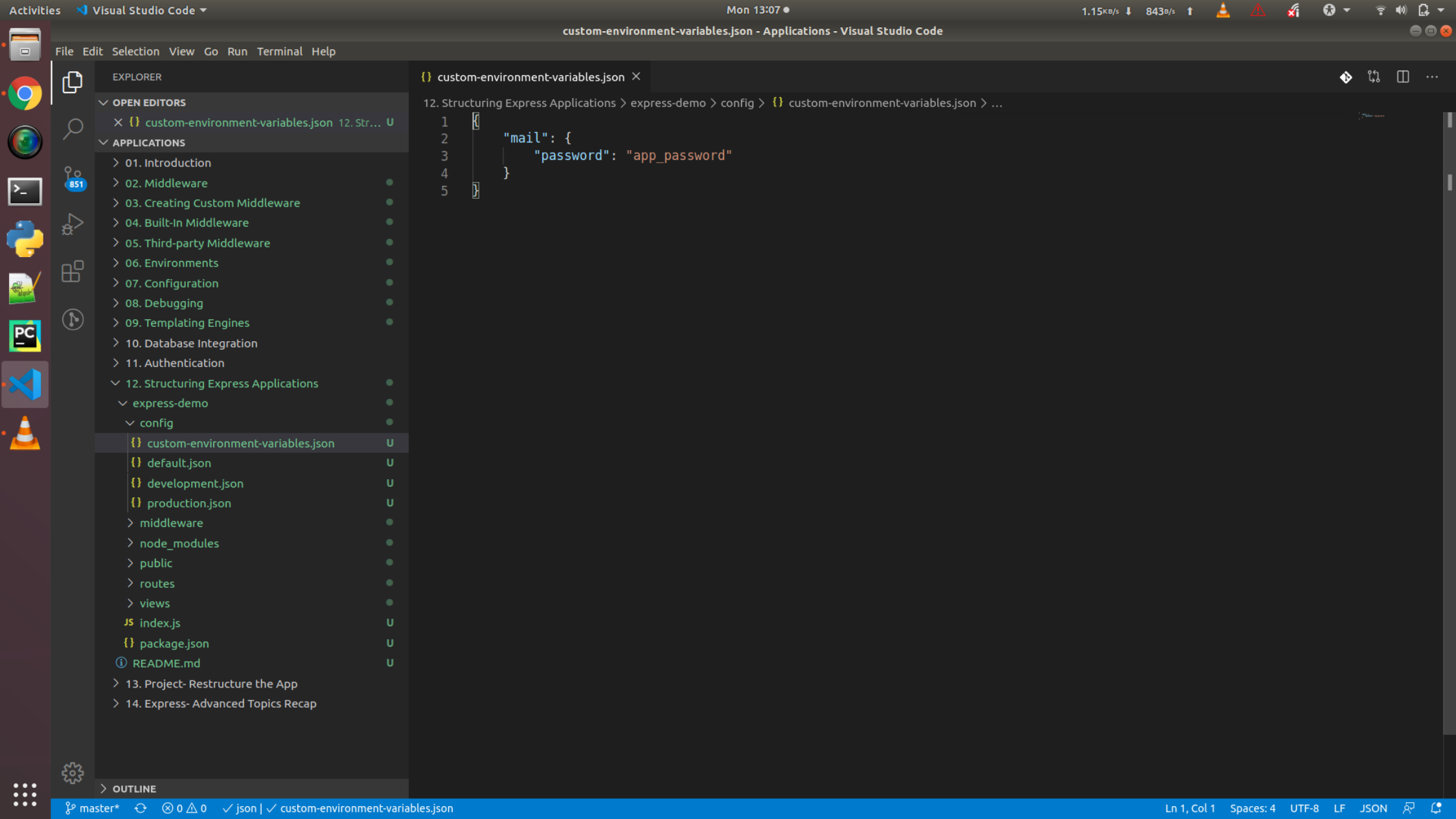


# Building Maintainable Routes



**Mosh Hamedani**  
[programmingwithmosh.com](https://programmingwithmosh.com)



ActivitiesVisual Studio Codemon Mon 13:072.23KB/s ↓ 4.61KB/s ↑

default.json - Applications - Visual Studio Code

FileEditSelectionViewGoRunTerminalHelp

EXPLORER

OPEN EDITORS

custom-environment-variables.json 12. Str... U

default.json 12. Structuring Express Applicati... U

APPLICATIONS

> 01. Introduction

> 02. Middleware

> 03. Creating Custom Middleware

> 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

> express-demo

> config

custom-environment-variables.json U

default.json U

deve

prod

> middleware

> node\_modules

> public

> routes

> views

index.js U

package.json U

README.md U

> 13. Project- Restructure the App

> 14. Express- Advanced Topics Recap

OUTLINE

custom-environment-variables.jsondefault.json X

12. Structuring Express Applications > express-demo > config > default.json > ...

1

2

3

{

"name": "My Express Application"

}

~/Keep/Node.js/05. Express- Advanced Topics/Applications/12. Structuring Express Applications/express-demo/config/default.json • Untracked

master\*0 0 0json | default.json

Ln 1, Col 1Spaces: 4UTF-8LFJSON

ActivitiesVisual Studio Code

development.json - Applications - Visual Studio Code

6.06KB/s ↓ 2.92KB/s ↑

Mon 13:07

File Edit Selection View Go Run Terminal Help

EXPLORER

OPEN EDITORS

custom-environment-variables.json 12. Str... U

development.json 12. Structuring Express A... U

APPLICATIONS

01. Introduction

02. Middleware

03. Creating Custom Middleware

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

express-demo

config

custom-environment-variables.json U

default.json U

development.json U

production.json U

middleware

node\_modules

public

routes

views

index.js U

package.json U

README.md U

13. Project- Restructure the App

14. Express- Advanced Topics Recap

OUTLINE

custom-environment-variables.json

development.json

12. Structuring Express Applications > express-demo > config > development.json > ...

1 {

2     "name": "My Express Application - Development",

3     "mail": {

4         "host": "dev-mail-server"

5     }

6 }

master\*

0 0 0

json | development.json

Ln 1, Col 1

Spaces: 4

UTF-8

LF

JSON

Activities

Visual Studio Code

Mon 13:08

0B/s ↓ 0B/s ↑

production.json - Applications - Visual Studio Code

File Edit Selection View Go Run Terminal Help

EXPLORER

OPEN EDITORS

custom-environment-variables.json 12. Str... U

development.json 12. Structuring Express A... U

production.json 12. Structuring Express Appli... U

APPLICATIONS

> 01. Introduction

> 02. Middleware

> 03. Creating Custom Middleware

> 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

12. Structuring Express Applications > express-demo > config > production.json > ...

1 {

2     "name": "My Express Application - Production",

3     "mail": {

4         "host": "prod-mail-server"

5     }

6 }

master\*

0 0 0

json | production.json

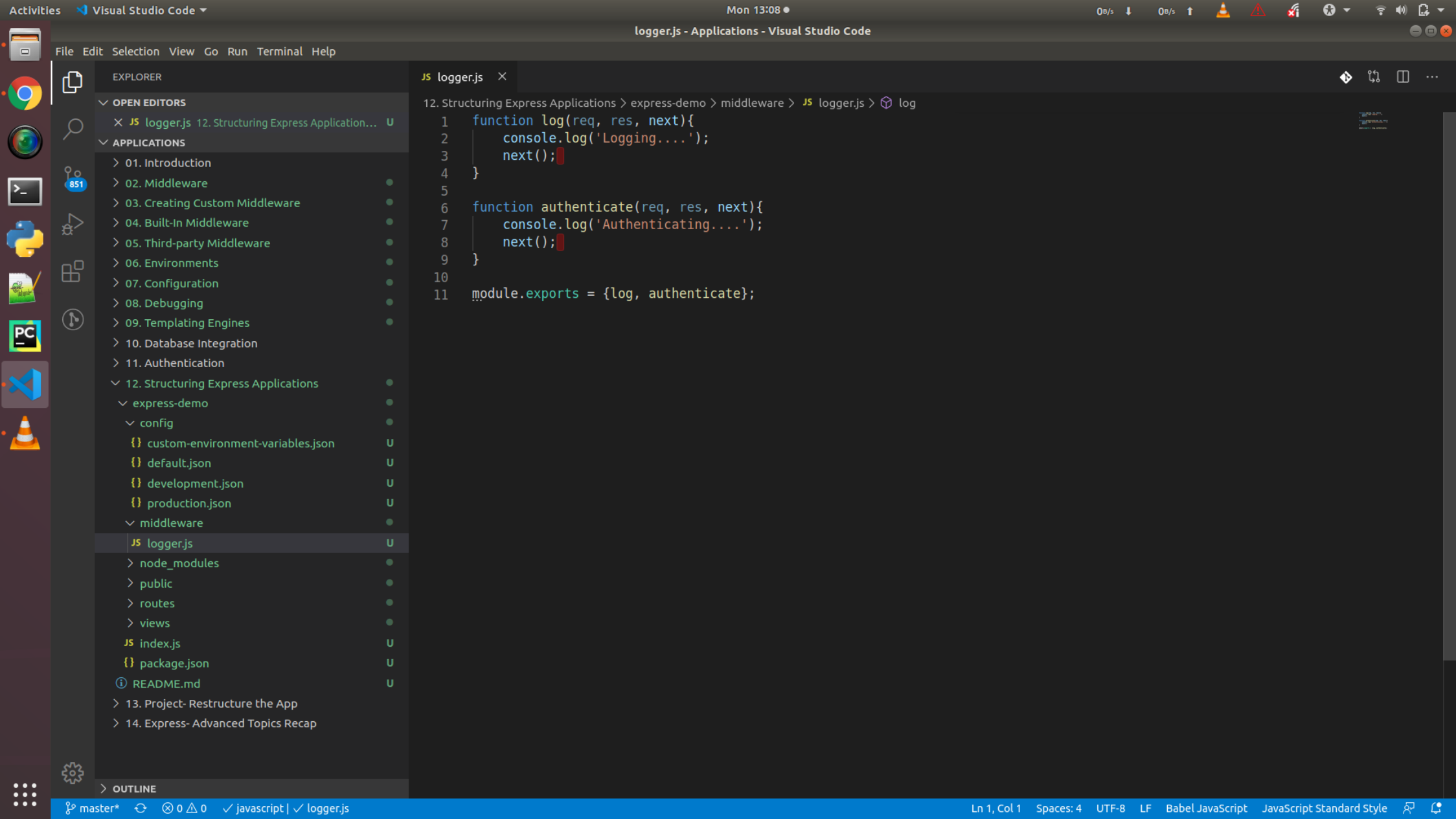
Ln 1, Col 1

Spaces: 4

UTF-8

LF

JSON



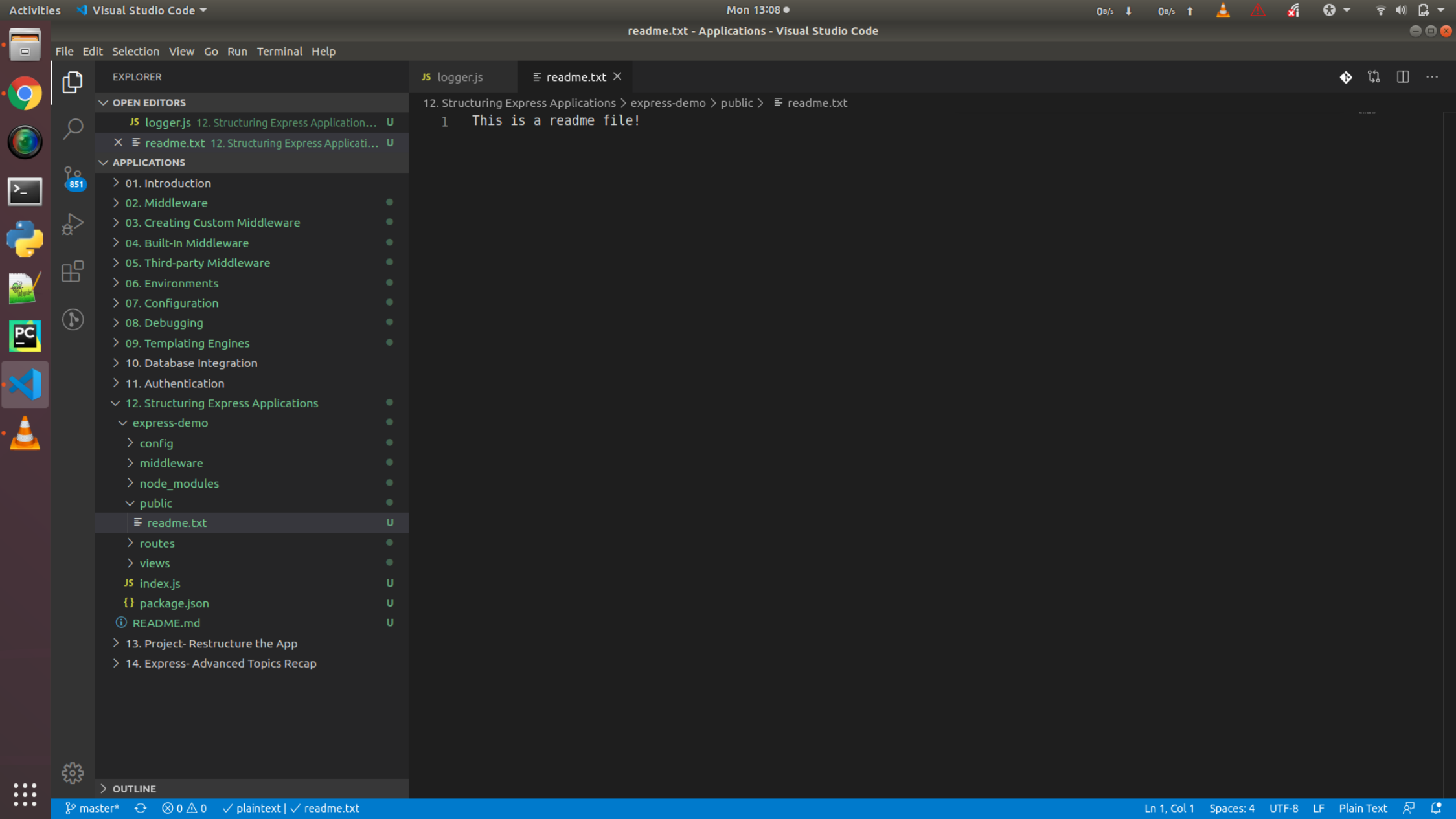
```
Mon 13:08
0b/s
0b/s
logger.js - Applications - Visual Studio Code

File Edit Selection View Go Run Terminal Help

EXPLORER
OPEN EDITORS
  X JS logger.js 12. Structuring Express Application...
APPLICATIONS
  > 01. Introduction
  > 02. Middleware
  > 03. Creating Custom Middleware
  > 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
    > express-demo
      > config
        {} custom-environment-variables.json
        {} default.json
        {} development.json
        {} production.json
      > middleware
        JS logger.js
        > node_modules
        > public
        > routes
        > views
        JS index.js
        {} package.json
        ⓘ README.md
  > 13. Project- Restructure the App
  > 14. Express- Advanced Topics Recap

12. Structuring Express Applications > express-demo > middleware > JS logger.js > log
1 function log(req, res, next){
2   console.log('Logging....');
3   next();
4 }
5
6 function authenticate(req, res, next){
7   console.log('Authenticating....');
8   next();
9 }
10
11 module.exports = {log, authenticate};

Ln 1, Col 1 Spaces: 4 UTF-8 LF Babel JavaScript JavaScript Standard Style
```



ActivitiesVisual Studio Code

Mon 13:09

907B/s228B/s

courses.js - Applications - Visual Studio Code

FileEditSelectionViewGoRunTerminalHelp

EXPLORER

OPEN EDITORS1 UNSAVED

JS courses.js 12. Structuring Express Applicatio... U

APPLICATIONS

01. Introduction

02. Middleware

03. Creating Custom Middleware

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

express-demo

config

middleware

node\_modules

public

routes

JS courses.js U

JS home.js U

views

JS index.js U

package.json U

README.md U

13. Project- Restructure the App

14. Express- Advanced Topics Recap

OUTLINE

JS courses.js

12. Structuring Express Applications > express-demo > routes > JS courses.js > router.get('/') callback

1const express = require('express');

2//const app = express(); // This doesn't work when we separate the route in a different module.

3const router = express.Router();

4const Joi = require('joi');

5

6

7const courses = [

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

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

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

11];

12

13router.get('/', (req, res) => {

14// router.get('/api/courses', (req, res) => {

15// Because we have used 'app.use('/api/courses', courses);' in index.js

16res.send(courses);

17});

18

Complexity is 3 Everything is cool!

19router.post('/', (req, res) => {

20const { error } = validateCourse(req.body);

21if (error) return res.status(400).send(error.details[0].message);

22

23const course = {

24id: courses.length + 1,

25name: req.body.name

26};

27courses.push(course);

28res.send(course);

29});

30

Complexity is 6 It's time to do something...

31router.put('/:id', (req, res) => {

32const course = courses.find(c => c.id === parseInt(req.params.id));

33if (!course) return res.status(404).send('The course with the given ID was not found.');

34

35const { error } = validateCourse(req.body);

36if (error) return res.status(400).send(error.details[0].message);

37

38course.name = req.body.name;

Ln 15, Col 9Spaces: 4UTF-8LFBabel JavaScriptJavaScript Standard Style



Activities

Visual Studio Code

Mon 13:09

0a/s

0a/s

courses.js - Applications - Visual Studio Code

File Edit Selection View Go Run Terminal Help

EXPLORER

OPEN EDITORS 1 UNSAVED

JS courses.js 12. Structuring Express Applicatio...

APPLICATIONS

01. Introduction

02. Middleware

03. Creating Custom Middleware

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

express-demo

config

middleware

node\_modules

public

routes

JS courses.js

JS home.js

views

JS index.js

package.json

README.md

13. Project- Restructure the App

14. Express- Advanced Topics Recap

OUTLINE

JS courses.js

12. Structuring Express Applications > express-demo > routes > JS courses.js > router.get('/') callback

Complexity is 4 It's time to do something...

Complexity is 4 Everything is cool!

Complexity is 4 Everything is cool!

```
31 router.put('/:id', (req, res) => {
32     const course = courses.find(c => c.id === parseInt(req.params.id));
33     if (!course) return res.status(404).send('The course with the given ID was not found.');
```

```
34
35     const { error } = validateCourse(req.body);
36     if (error) return res.status(400).send(error.details[0].message);
37
38     course.name = req.body.name;
39     res.send(course);
40 });
41
42 router.delete('/:id', (req, res) => {
43     const course = courses.find(c => c.id === parseInt(req.params.id));
44     if (!course) return res.status(404).send('The course with the given ID was not found.');
```

```
45
46     const index = courses.indexOf(course);
47     courses.splice(index, 1);
48
49     res.send(course);
50 });
51
52 router.get('/:id', (req, res) => {
53     const course = courses.find(c => c.id === parseInt(req.params.id));
54     if (!course) return res.status(404).send('The course with the given ID was not found.');
```

```
55     res.send(course);
56 });
57
58 function validateCourse(course) {
59     const schema = {
60         name: Joi.string().min(3).required()
61     };
62
63     return Joi.validate(course, schema);
64 }
65
66 module.exports = router;
```

Ln 15, Col 9 Spaces: 4 UTF-8 LF Babel JavaScript JavaScript Standard Style

ActivitiesVisual Studio Code

Mon 13:1085.8B/s584B/s

home.js - Applications - Visual Studio Code

FileEditSelectionViewGoRunTerminalHelp

EXPLORER

OPEN EDITORS1 UNSAVED

JS home.js12. Structuring Express Applications...

APPLICATIONS

> 01. Introduction

> 02. Middleware

> 03. Creating Custom Middleware

> 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

express-demo

> config

> middleware

> node\_modules

> public

routes

JS courses.js

JS home.js

> views

JS index.js

{ } package.json

i README.md

> 13. Project- Restructure the App

> 14. Express- Advanced Topics Recap

OUTLINE

JS home.js

12. Structuring Express Applications > express-demo > routes > JS home.js > ...

1const express = require('express');

2//const app = express(); // This doesn't work when we separate the route in a different module.

3const router = express.Router();

4

5app.get('/', (req, res) => {

6|res.render('index',{title: 'My Express App',message:'Hello'});

7});

8

9module.exports = router;

master\*

00

javascript | home.js

Ln 2, Col 1

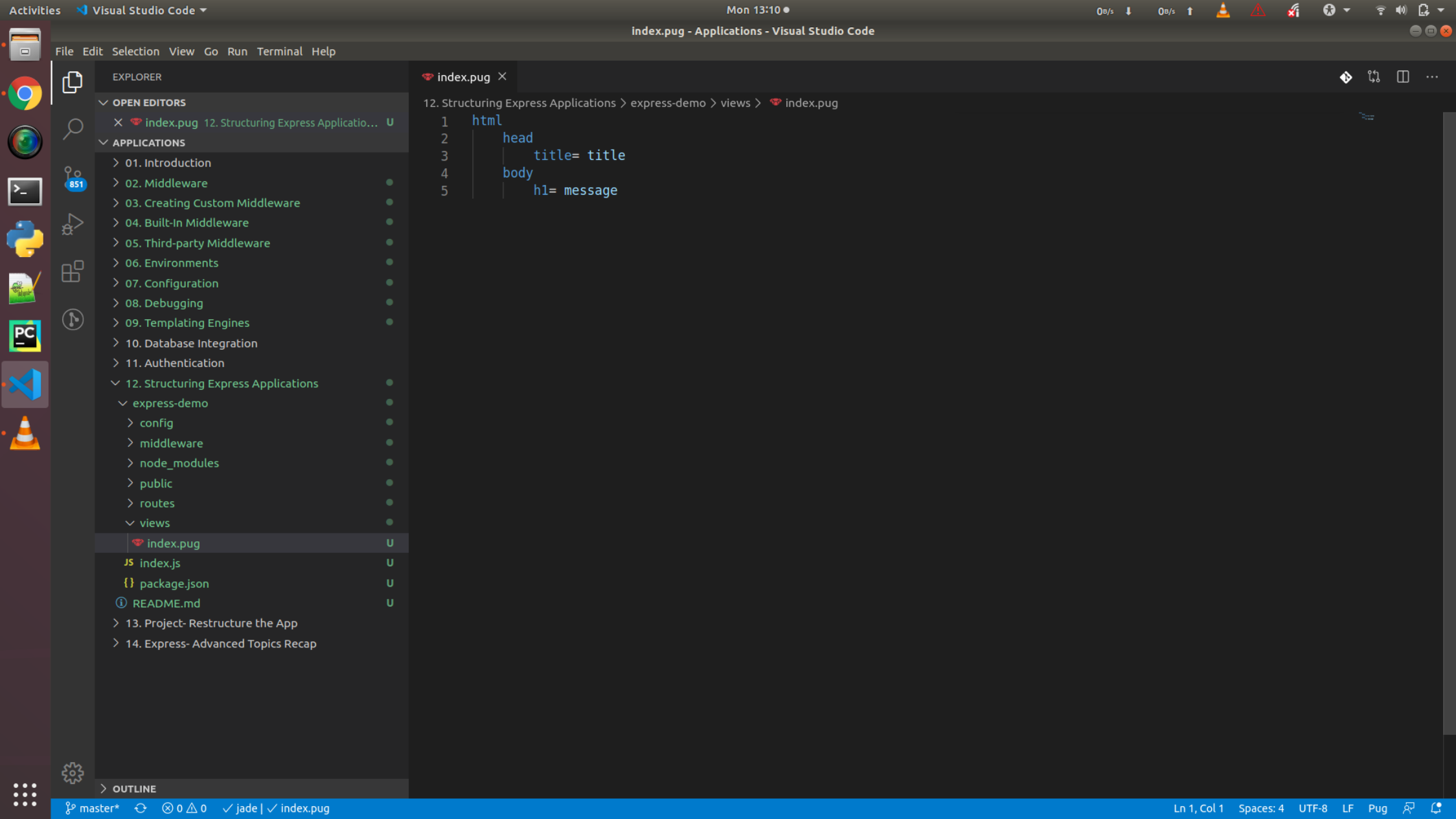
Spaces: 4

UTF-8

LF

Babel JavaScript

JavaScript Standard Style



ActivitiesVisual Studio Code

Mon 13:10

0b/s0b/s

index.js - Applications - Visual Studio Code

FileEditSelectionViewGoRunTerminalHelp

EXPLORER

OPEN EDITORS

JS index.js 12. Structuring Express Applications/... U

APPLICATIONS

01. Introduction

02. Middleware

03. Creating Custom Middleware

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

express-demo

config

middleware

node\_modules

public

routes

views

JS index.js U

package.json U

README.md U

13. Project- Restructure the App

14. Express- Advanced Topics Recap

OUTLINE

JS index.js

12. Structuring Express Applications > express-demo > JS index.js > ...

```
1  const debug = require('debug')('app:startup');
2  const config = require('config')
3  const morgan = require('morgan');// Morgan : HTTP request logger middleware for node.js.
4  const helmet = require('helmet');// Helmet helps you secure your Express.js apps by setting various HTTP headers.
5  const Joi = require('joi');
6  const logger = require('./middleware/logger');
7  const courses = require('./routes/courses');
8  const courses = require('./routes/home');
9  const express = require('express');
10 const app = express();
11
12 app.set('view engine','pug');
13 app.set('views','./views'); // default : './views'
14
15 app.use(express.json()); // req.body
16 app.use(express.urlencoded({ extended: true })); // key=value&key=value
17 app.use(express.static('public'));
18 app.use(helmet());
19 app.use('/api/courses', courses); // Now we can change all the '/api/courses with' '/' in routes/courses.js
20 app.use('/', home);
21
22 // Configuration
23 console.log('Application Name: '+config.get('name'));
24 console.log('Mail Server: '+config.get('mail.host'));
25 console.log('Mail Password: '+config.get('mail.password'));
26
27 if (app.get('env') == 'development') {
28     app.use(morgan('tiny'));
29     debug('Morgan enabled...');
30 }
31
32 app.use(logger);
33
34 const port = process.env.PORT || 3000;
35 app.listen(port, () => console.log(`Listening to port ${port} ....`));
```

Ln 11, Col 1

Spaces: 4

UTF-8

LF

Babel JavaScript

JavaScript Standard Style

ActivitiesVisual Studio Code

Mon 13:10

0B/s0B/s

package.json - Applications - Visual Studio Code

FileEditSelectionViewGoRunTerminalHelp

EXPLORER

OPEN EDITORS

package.json12. Structuring Express Applica... U

APPLICATIONS

> 01. Introduction

> 02. Middleware

> 03. Creating Custom Middleware

> 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

express-demo

> config

> middleware

> node\_modules

> public

> routes

> views

JSindex.js U

{}package.json U

①README.md U

> 13. Project- Restructure the App

> 14. Express- Advanced Topics Recap

{} package.json X

12. Structuring Express Applications > express-demo > {} package.json > {} dependencies

```
1 {
2   "name": "express-demo",
3   "version": "1.0.0",
4   "main": "index.js",
5   "scripts": {
6     "test": "echo \"Error: no test specified\" && exit 1"
7   },
8   "keywords": [],
9   "author": "",
10  "license": "ISC",
11  "dependencies": {
12    "config": "^3.3.1",
13    "debug": "^4.1.1",
14    "express": "^4.17.1",
15    "helmet": "^3.22.0",
16    "joi": "^14.3.1",
17    "morgan": "^1.10.0"
18  },
19  "devDependencies": {},
20  "description": ""
21 }
22
```

Ln 14, Col 26Spaces: 2UTF-8LFJSON

master\*0 0 0json | package.json