

hash-demo — node • node /usr/local/bin/nodemon — 80x22
~/Documents/GA_stuff/hash-demo — node • node /usr/local/bin/nodemon
~/Documents/GA_stuff/blog-made-in-class — mongo
lParser: true } to MongoClient.connect.
APP IS GO
MONGODB ACTIVE

localhost x G how to use bc x Hashing Pass x bcrypt - npm x G how to do som x Mongoose v5 x Mongoose v5 x node.js - Mon x Pre-save hook x

EXPLORER JS Index.js JS user.js <> index.ejs JS db.js

OPEN EDITOR... 1 UNSAVED

- JS index.js
- JS user.js models
- <> index.ejs views
- JS db.js db

HASH-DEMO

- db
- models
- JS user.js
- node_modules
- views
- <> index.ejs

JS index.js

- () package-lock.json
- () package.json

```
26 await newUser.save();
27 res.redirect('/')
28 })
29
30 app.post('/login', async (req, res, next) => {
31   const userTryingToLogIn = await User.findOne({username: req.body.username})
32   if(!userTryingToLogIn){
33     next(new Error("User not found"))
34     res.redirect("/")
35   }
36   const validLogin = await bcrypt.compare(req.body.password, userTryingToLogIn.password)
37   if(!validLogin){
38     res.redirect("/")
39   } else {
40     res.redirect('/success')
41   }
42 })
43
44 app.get('/success', (req, res) => {
45   res.send("congratulations")
46 })
47
48 app.listen(3000, () => {
49   console.log("APP IS GO")
50 })
```

Restart Visual Studio Code to apply the latest update.

Update Now Later Release Notes

Ln 32, Col 42 Spaces: 4 UTF-8 LF JavaScript

(node:21769) DeprecationWarning: current URL string parser is deprecated, and will be removed in a future version. To use the new parser, pass option { useNewUrlParser: true } to MongoClient.connect.
APP IS GO

EXPLORER

OPEN EDITORS

JS index.js
JS user.js models
<> index.ejs views
JS db.js db

HASH-DEMO

db
models
JS user.js
node_modules
views
<> index.ejs
JS index.js
{ } package-lock.json
{ } package.json

JS index.js

JS user.js

<> index.ejs

JS db.js

```
1 const mongoose = require('mongoose')
2
3 const UserSchema = new mongoose.Schema({
4   username: String,
5   password: String
6 })
7
8 UserSchema.pre('save', async function(next){
9   const existingUser = await UserSchema.findOne({username: this.username})
10  if(existingUser){
11    console.log("USER ALREADY EXISTS DONT HASH")
12    next();
13  } else {
14    console.log("NEW USER TIME TO HASH")
15  }
16  next();
17 })
18 module.exports = mongoose.model('User', UserSchema);
```

Restart Visual Studio Code to apply the latest update.

Update Now

Later

Release Notes

OUTLINE

Ln 16, Col 6 (6 selected)

Spaces: 4

UTF-8

LF

JavaScript

1

...
p_two_mojito ranches p_three_mojito

[nodemon] starting 'node index.js'
(node:21756) DeprecationWarning: current URL string parser is deprecated, and will be removed in a future version. To use the new parser, pass option { useNewURLParser: true } to MongoClient.connect.

EXPLORER

OPEN EDITORS

JS index.js

JS user.js models

JS index.js views

JS db.js db

HASH-DEMO

db

models

JS user.js

node_modules

views

JS index.js

JS index.js

(1) package-lock.json

(1) package.json

OUTLINE

require('./db/db.js');

const User = require('../models/user.js');

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

res.render('index.ejs');

});

app.post('/register', async (req, res) => {

const newUser = await User.create(req.body);

const newPassword = bcrypt.hashSync(newUser.password, 10);

newUser.password = newPassword;

await newUser.save();

res.redirect('/');

});

app.post('/login', async (req, res) => {

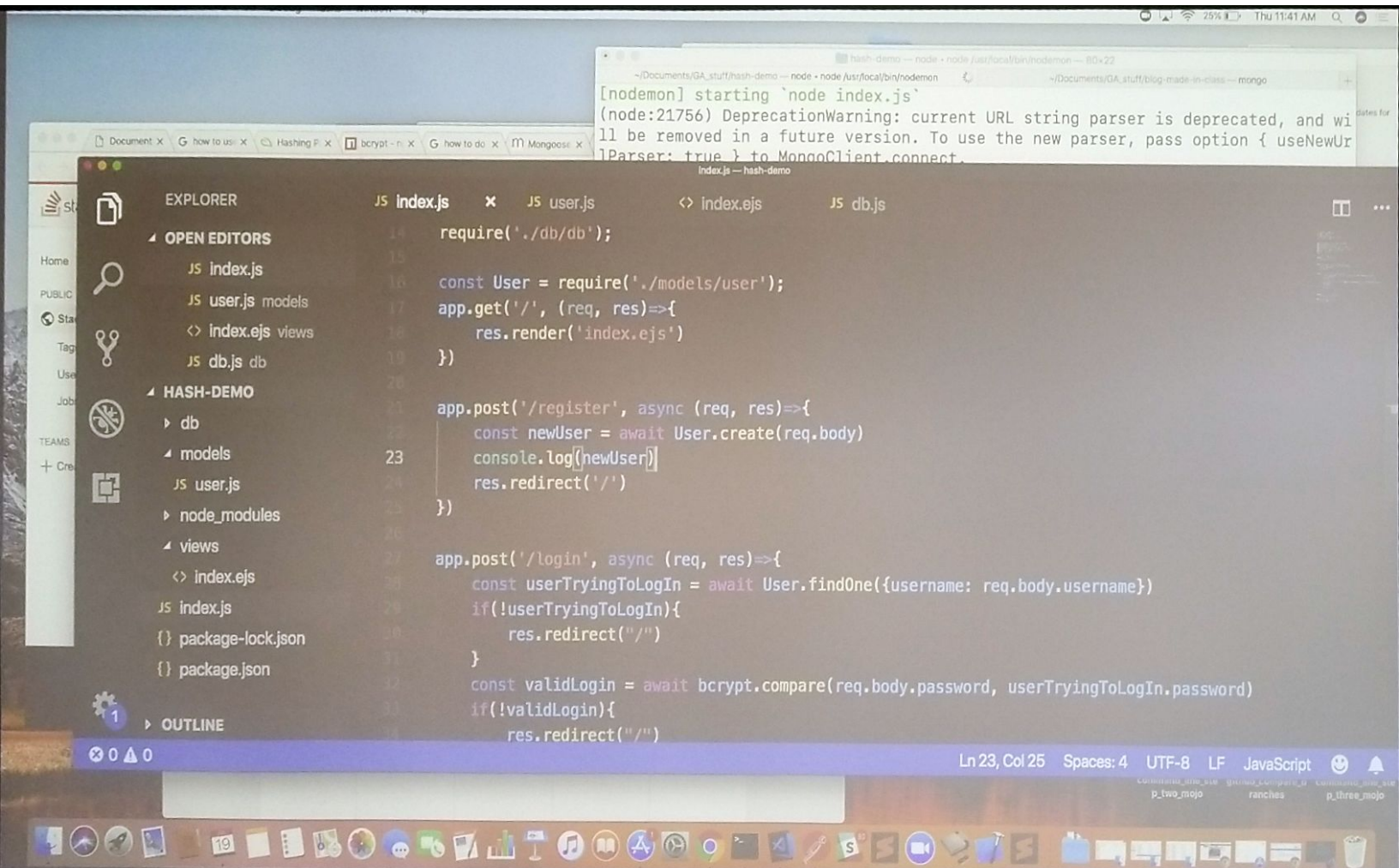
const userTryingToLogin = await User.findOne({username: req.body.username});

if(!userTryingToLogin) {

res.redirect('/');

};

const validLogin = await bcrypt.compare(req.body.password, userTryingToLogin.password);



hash-demo — node • node /usr/local/bin/node — 80x22
~/Documents/GA_stuff/hash-demo — node • node /usr/local/bin/node —
~/Documents/GA_stuff/log-made-in-class — mongo
[nodemon] starting `node index.js`
(node:21756) DeprecationWarning: current URL string parser is deprecated, and will be removed in a future version. To use the new parser, pass option { useNewUrlParser: true } to MongoClient.connect.
user.js — hash-demo

Document x how to use x Hashing P x bcrypt - n x G how to do x M Mongoose x

EXPLORER

- OPEN EDITORS
 - JS index.js
 - JS user.js models
 - <> index.ejs views
 - JS db.js db
- HASH-DEMO
 - db
 - models
 - JS user.js
 - node_modules
 - views
 - <> index.ejs
 - JS index.js
 - { } package-lock.json
 - { } package.json

OUTLINE

JS index.js

```
1 const mongoose = require('mongoose')
2 const bcrypt = require('bcrypt')
3
4 const UserSchema = new mongoose.Schema({
5   username: String,
6   password: String
7 })
8
9 UserSchema.pre('save', async function(next){
10   const existingUser = await UserSchema.findOne({username: this.username})
11   if(!existingUser){
12     this.password = await bcrypt.hash(this.password, 10);
13   }
14   next();
15 })
16 module.exports = mongoose.model('User', UserSchema);
```

JS user.js x <> index.ejs JS db.js

Ln 11, Col 23 Spaces: 4 UTF-8 LF JavaScript

