

Node Auth Tutorial (JWT)

#2 - Auth Routes & Controllers



In this node auth tutorial we'll start to create our authentication routes & controller functions.



express

Auth Routes

- Using an MVC (model, view & controller) approach

Auth Routes

/signup	GET	sign up page
/login	GET	log in page
/signup	POST	create a new user in db
/login	POST	authenticate a current user
/logout	GET	log a user out

- Create folder "routes"
- Create file "authRoutes.js" in the folder "routes"
- Create folder "controllers"
- Create file "authController.js" in the folder "controllers"



```
const { Router } = require('express');
const authController = require('../controllers/authController');

const router = Router();

router.get('/signup', authController.signup_get);
router.post('/signup', authController.signup_post);
router.get('/login', authController.login_get);
router.post('/login', authController.login_post);

module.exports = router;
```

```
// controller actions
module.exports.signup_get = (req, res) => {
  res.render('signup');
}

module.exports.login_get = (req, res) => {
  res.render('login');
}

module.exports.signup_post = async (req, res) => {
  res.send('new signup');
}

module.exports.login_post = async (req, res) => {
  res.send('user login');
}
```

- Create file "signup.ejs" in the folder "views"
- Create file "login.ejs" in the folder "views"



signup.ejs

```
<%- include('partials/header'); -%>
```

```
<h1>Sign up</h1>
```

```
<%- include('partials/footer'); -%>
```

login.ejs

```
<%- include('partials/header'); -%>
```

```
<h1>LogIn</h1>
```

```
<%- include('partials/footer'); -%>
```

```
const express = require('express');
const mongoose = require('mongoose');
const authRoutes = require('./routes/authRoutes');
```

← **Insert code**

```
const app = express();

// middleware
app.use(express.static('public'));

// view engine
app.set('view engine', 'ejs');

// database connection
const dbURI = 'mongodb+srv://marizza:demo1234@cluster0.nem3y.mongodb.net/node-auth';
mongoose.connect(dbURI, { useNewUrlParser: true, useUnifiedTopology: true })
  .then((result) => app.listen(3000))
  .catch((err) => console.log(err));

// routes
app.get('/', (req, res) => res.render('home'));
app.get('/smoothies', (req, res) => res.render('smoothies'));
app.use(authRoutes);
```

← **Insert code**

Go to <http://localhost:3000/signup>

Go to <http://localhost:3000/loginv>

Master Chef Smoothies

Sign up

Copyright 2021 Master Chef Smoothies

Master Chef Smoothies

Login

Copyright 2021 Master Chef Smoothies