

Node Auth Tutorial (JWT)

#3 - Testing Routes & Handling POST Requests



In this Node auth tutorial we'll test some of our routes using Postman, but later on we will send all of our requests from the browser.



express

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

```
const app = express();
```

```
// middleware
```

```
app.use(express.static('public'));
```

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



Insert code

```
// 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);
```

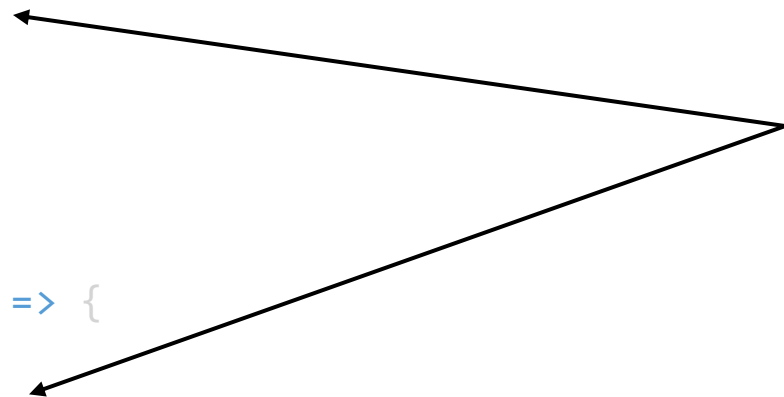
```
// controller actions
module.exports.signup_get = (req, res) => {
  res.render('signup');
}

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

module.exports.signup_post = async (req, res) => {
  const { email, password } = req.body;
  console.log(email, password);
  res.send('new signup');
}

module.exports.login_post = async (req, res) => {
  const { email, password } = req.body;
  console.log(email, password);
  res.send('user login');
}
```

Insert code



http://localhost:3000/

Save



GET

http://localhost:3000/

Send

Params Auth Headers (6) Body Pre-req. Tests Settings

Cookies

Query Params

	KEY	VALUE	DESCRIPTION	...	Bulk Edit
	Key	Value	Description		

Body Cookies Headers (7) Test Results

200 OK 26 ms 960 B Save Response

Pretty

Raw

Preview

Visualize

HTML



```
1 <html lang="en">
2
3 <head>
4   <meta charset="UTF-8">
5   <meta name="viewport" content="width=device-width, initial-scale=1.0">
6   <title>Document</title>
7   <link rel="stylesheet" href="/styles.css">
8 </head>
```



POSTMAN

http://localhost:3000/signup

Save



POST

http://localhost:3000/signup

Send

Params

Authorization

Headers (8)

Body

Pre-request Script

Tests

Settings

Cookies

none

form-data

x-www-form-urlencoded

raw

binary

GraphQL

JSON

Beautify

```
1 {  
2   "email": "mario@gmail.com",  
3   "password": "123321"
```

Body

Cookies

Headers (7)

Test Results



200 OK

16 ms

237 B

Save Response

Pretty

Raw

Preview

Visualize

HTML



1 new signup



POSTMAN

POST

http://localhost:3000/login

Send

Params

Authorization

Headers (8)

Body

Pre-request Script

Tests

Settings

Cookies

none

form-data

x-www-form-urlencoded

raw

binary

GraphQL

JSON

Beautify

```
1 {
2   "email": "luigi@gmail.com",
3   "password": "321123"
4 }
```

Body

Cookies

Headers (7)

Test Results



200 OK

6 ms

237 B

Save Response

Pretty

Raw

Preview

Visualize

HTML



1 user login



POSTMAN