



# Express Session

#Node.js Express Notes

# | Express Session



# What is Session ?

- Session is a Global Variable.



# What is use of Session ?

- User Authentication (Login)
- Pass Data From One Route to Another Route
- Save Data for Short Period
- Shopping Cart



# Install NPM

- Create New Express Generator Project
- Install Express Session Package
- <https://www.npmjs.com/package/express-session>
- `npm install express-session --save`



# Create Session Variable

- To store or access session data, simply use the request property `req.session`.
- Syntax :-
  - `req.session.SessionVariableName = "value" ;`
- Example:
  - `req.session.user = "akash" ;`
- Here user is session variable name



# Get Session Variable Value

- Syntax :
- Variable = req.session.SessionVariableName
- Example :
- myvar = req.session.user ;
- console.log(myvar);



# Check Session Variable is Set or not

Use below code to check session is set or not

```
if(req.session.user)
{
    myvar = req.session.user ; //Get Value
    console.log(`Session is set and Value is ' + myvar);
}
else
{
    console.log(`Session is not set`);
}
```





# Destroy Session Variable

- Destroys the session and will unset the req.session property
- Syntax :
  - `req.session.destroy();`



# Expire time

- `var hour = 3600000`
- `req.session.cookie.expires = new Date(Date.now() + hour)`
  
- `var hour = 3600000`
- `req.session.cookie.maxAge = hour`



# Middleware in Express Session

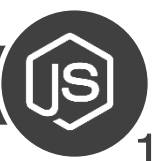
- Route :

```
req.session.cookie.expires = new Date(Date.now() + hour)
req.session.myemail=email;
req.session.userid=db_users_array._id;
req.session.username = db_users_array.user_name;
```

- App.js

```
app.use(session({ secret: 'keyboard cat',saveUninitialized: false,
resave: false, cookie: { maxAge:60000000000 }}}));
```

```
app.use(function(req, res, next) {
  res.locals.useremail = req.session.email;
  res.locals.userid = req.session.userid;
  res.locals.username =req.session.username;
  next();
});
```



# View Check Login Condition

```
<% if (locals.userid) { %>
    <li> <a href="/users/logout">MyAccount</a></li>
<% }else{ %>
    <li><a href="/users/login">Sign in</a></li>
<% } %>
```



# Steps

- Create New Project
- Install Express Session NPM In Project
- First we have to Specify Use of Session in App.js (App.Use)
- Also Add Secrete Key to Start Session in Nodejs Project you can also Specify Session MaxAge in app.use.



# App.js

- `//Call Express Session in App.js`
- `var session = require('express-session')`
- `//Secret Key Encrypt your Session Data`
- `//Session will Store Data till 60,000 MilliSeconds`
- `app.use(session({ secret: 'keyboard cat', cookie: { maxAge: 60000 } })))`



# To Test Session in Express Write Below Code in App.js

```
app.get('/sessiondemo', function(req, res, next) {  
  if (req.session.views) {  
    req.session.views++  
    res.setHeader('Content-Type', 'text/html')  
    res.write('<p>views: ' + req.session.views + '</p>')  
    res.write('<p>expires in: ' + (req.session.cookie.maxAge / 1000)  
    + 's</p>')  
    res.end()  
  } else {  
    req.session.views = 1  
    res.end('welcome to the session demo. refresh!')  
  }  
})
```



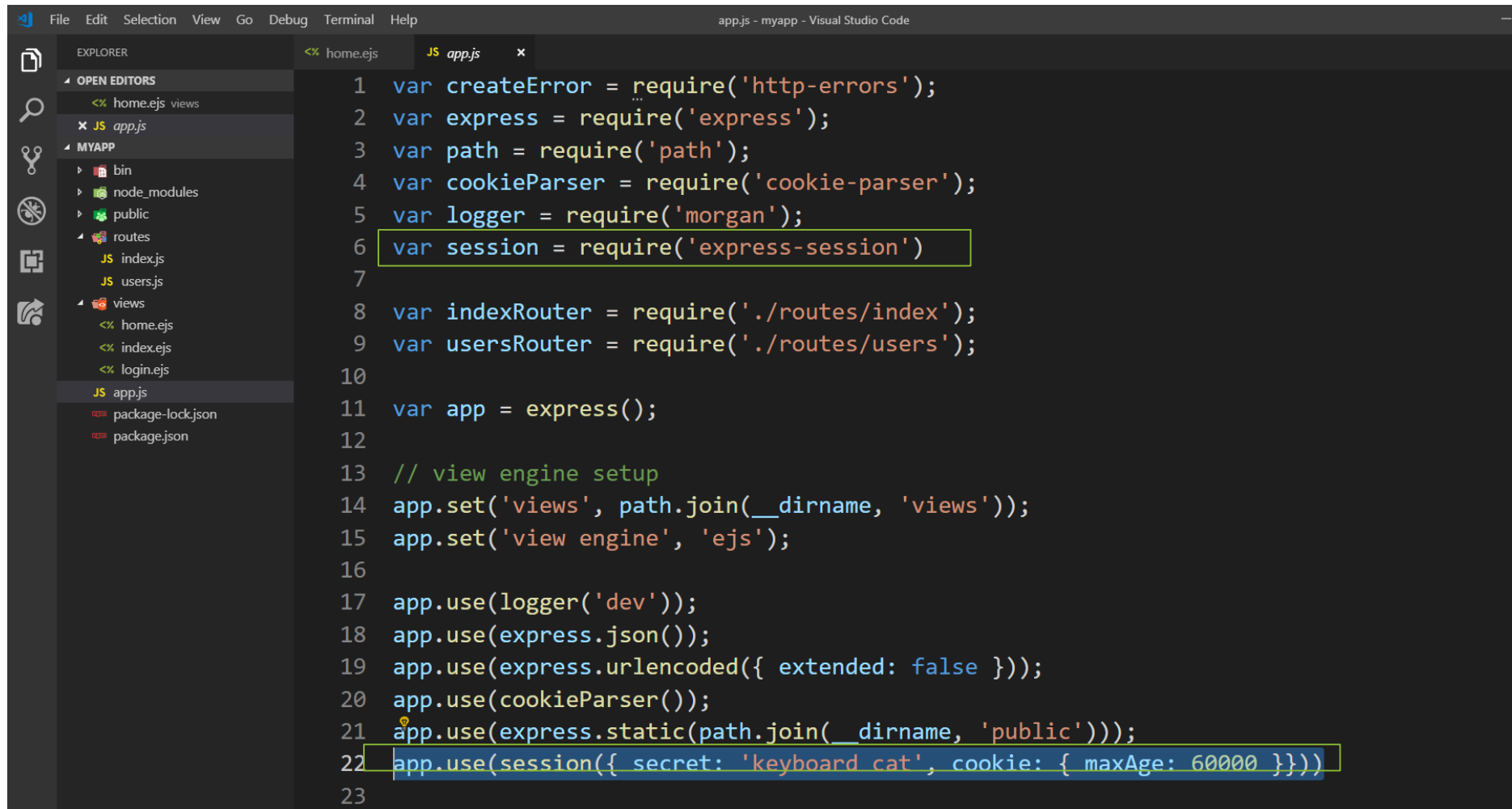
# Install Express Session

- First install Express Session in your project
- `> npm i express-session --save`





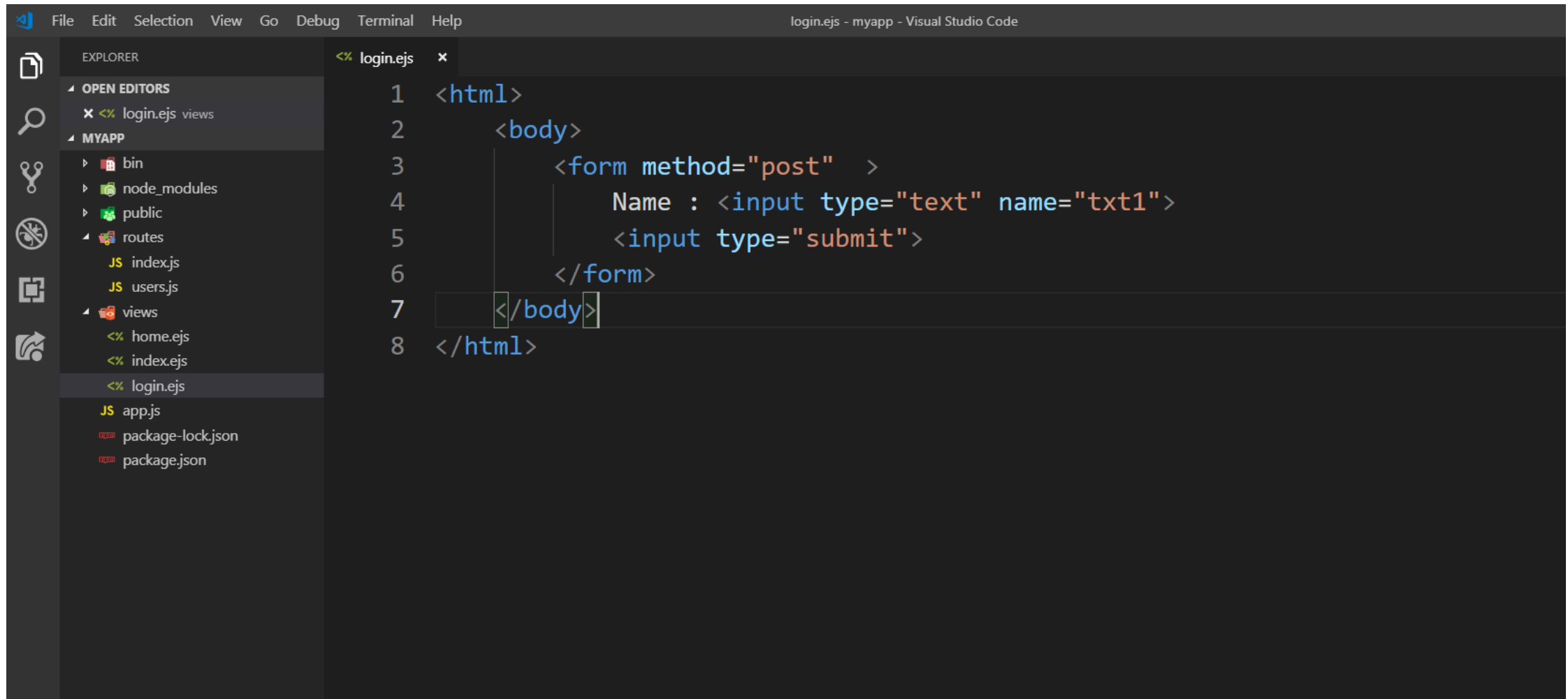
# Configure App.js



```
1 var createError = require('http-errors');
2 var express = require('express');
3 var path = require('path');
4 var cookieParser = require('cookie-parser');
5 var logger = require('morgan');
6 var session = require('express-session');
7
8 var indexRouter = require('./routes/index');
9 var usersRouter = require('./routes/users');
10
11 var app = express();
12
13 // view engine setup
14 app.set('views', path.join(__dirname, 'views'));
15 app.set('view engine', 'ejs');
16
17 app.use(logger('dev'));
18 app.use(express.json());
19 app.use(express.urlencoded({ extended: false }));
20 app.use(cookieParser());
21 app.use(express.static(path.join(__dirname, 'public')));
22 app.use(session({ secret: 'keyboard cat', cookie: { maxAge: 60000 }}}));
23
```



# Create Login.ejs File

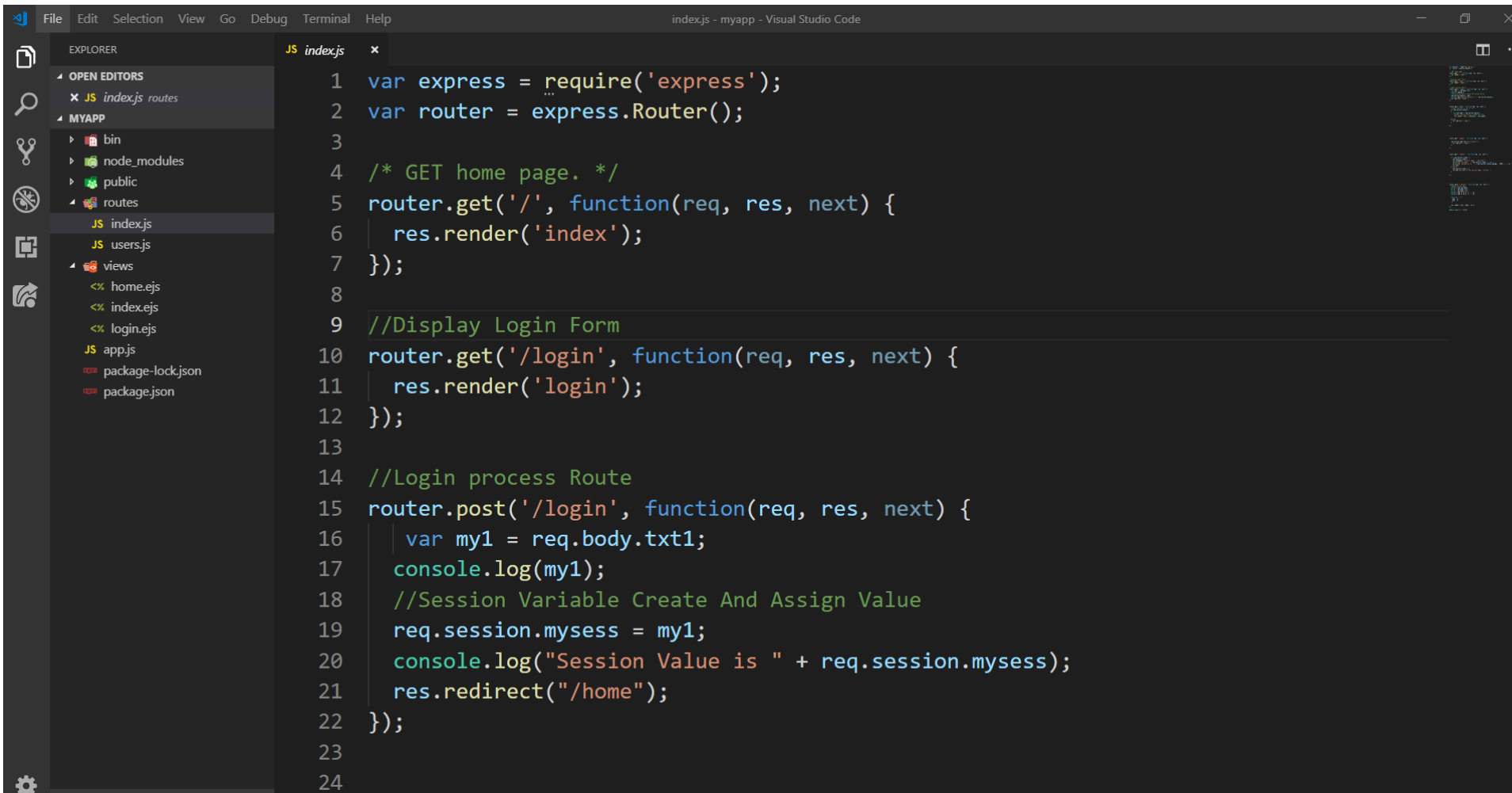


The screenshot shows the Visual Studio Code interface with a dark theme. The Explorer sidebar on the left displays a project structure for 'MYAPP' with folders 'bin', 'node\_modules', 'public', and 'routes'. The 'views' folder is expanded, showing files 'index.ejs', 'login.ejs', and 'app.js'. The 'login.ejs' file is selected and open in the main editor. The code in the editor is as follows:

```
1 <html>
2   <body>
3     <form method="post" >
4       Name : <input type="text" name="txt1">
5       <input type="submit">
6     </form>
7   </body>
8 </html>
```



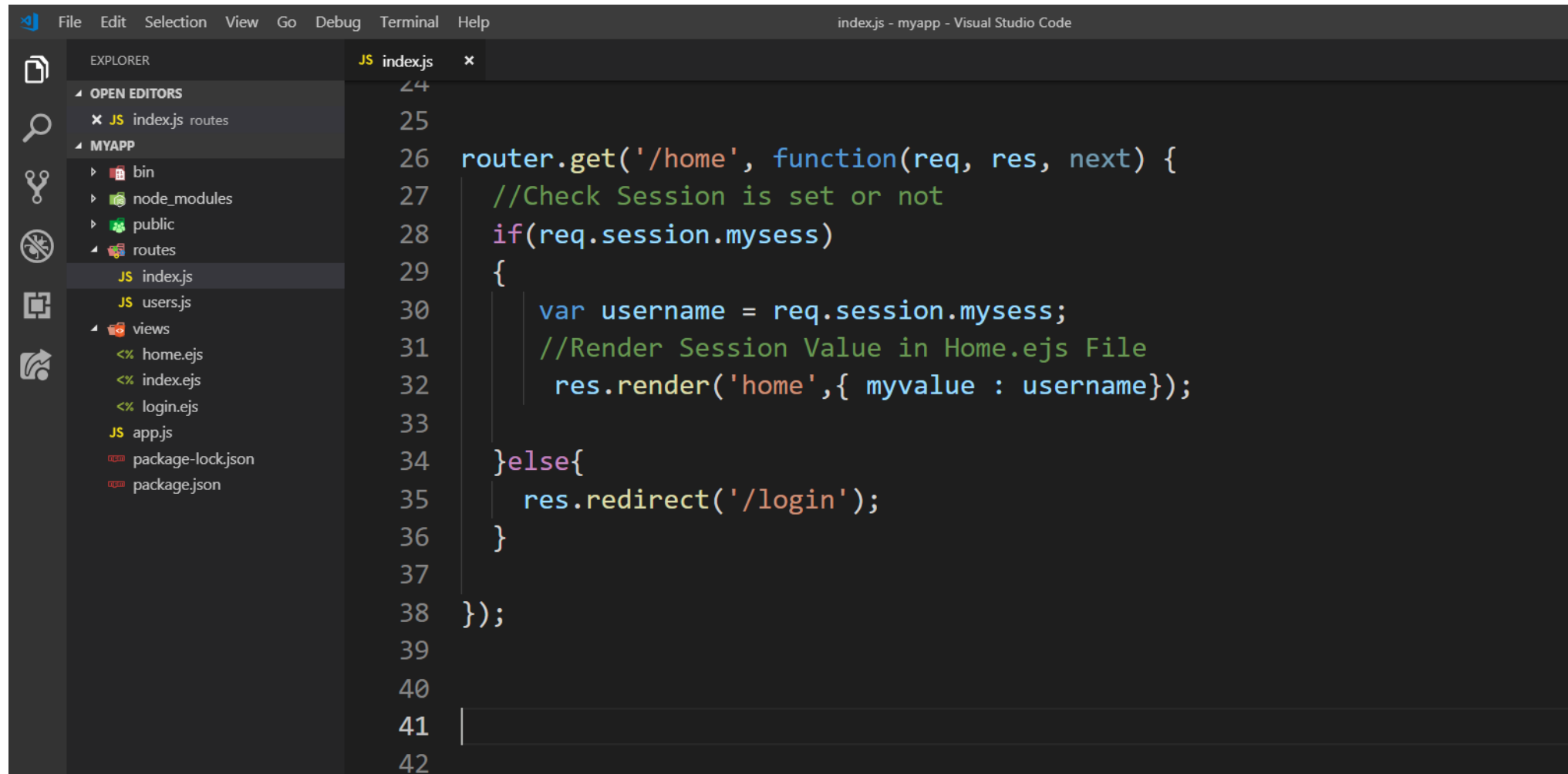
# Create Route for Login form (Get and Post)



```
1 var express = require('express');
2 var router = express.Router();
3
4 /* GET home page. */
5 router.get('/', function(req, res, next) {
6   res.render('index');
7 });
8
9 //Display Login Form
10 router.get('/login', function(req, res, next) {
11   res.render('login');
12 });
13
14 //Login process Route
15 router.post('/login', function(req, res, next) {
16   var my1 = req.body.txt1;
17   console.log(my1);
18   //Session Variable Create And Assign Value
19   req.session.mysess = my1;
20   console.log("Session Value is " + req.session.mysess);
21   res.redirect("/home");
22 });
23
24
```



# Create Home Route and Check Condition



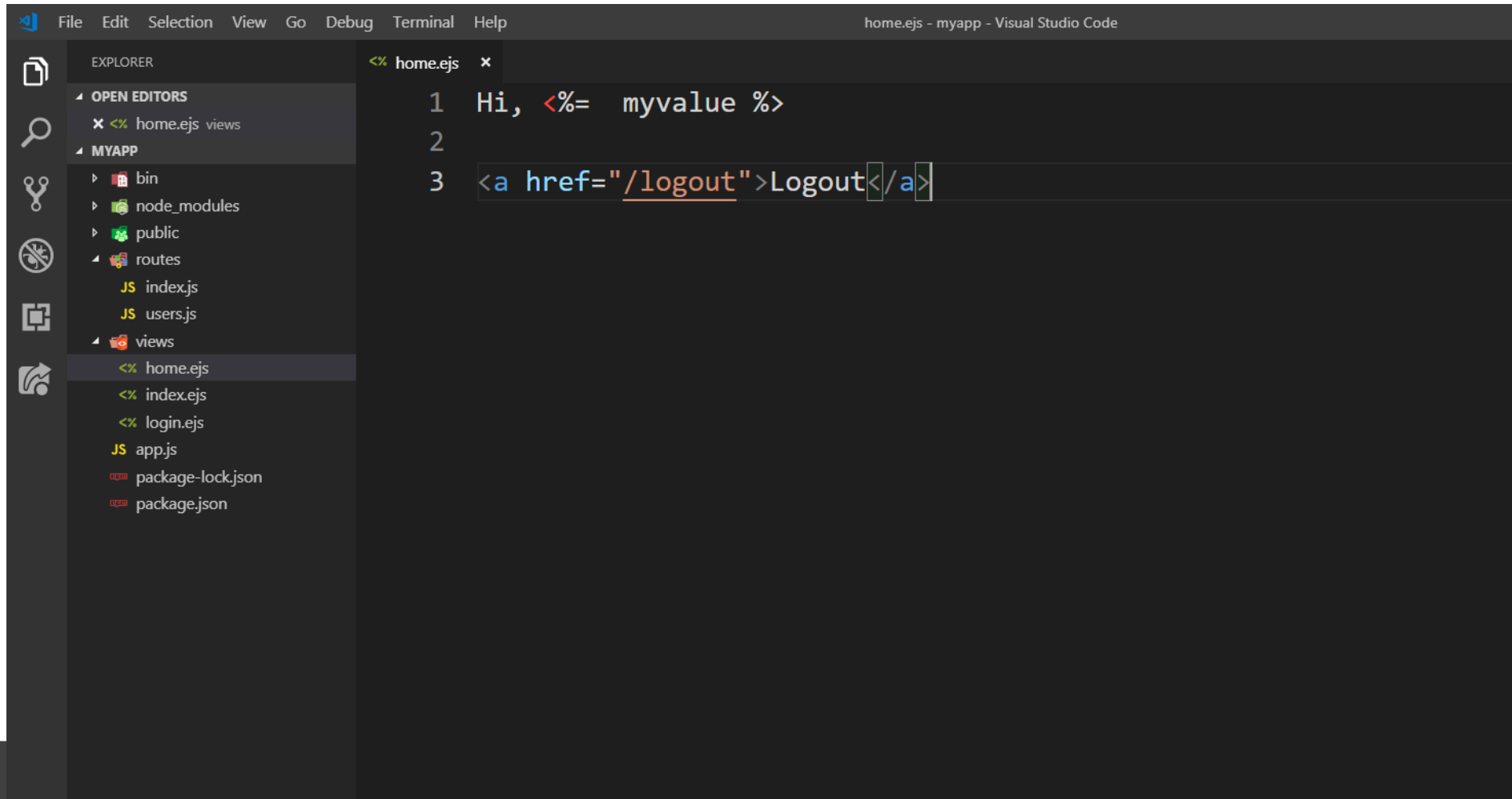
The screenshot shows the Visual Studio Code interface with a file explorer on the left and a code editor on the right. The file explorer shows a project structure with a 'routes' folder containing 'index.js'. The code editor displays the following JavaScript code in 'index.js':

```
24
25
26 router.get('/home', function(req, res, next) {
27     //Check Session is set or not
28     if(req.session.mysess)
29     {
30         var username = req.session.mysess;
31         //Render Session Value in Home.ejs File
32         res.render('home',{ myvalue : username});
33     }
34     else{
35         res.redirect('/login');
36     }
37
38 });
39
40
41
42
```



# Create Home.ejs

- Home Route will Check Condition Session Variable is Set or Not
- If Session Variable is Set then Render Home.ejs file
- Else It will redirect to Login Route which render login.ejs file




```
File Edit Selection View Go Debug Terminal Help
home.ejs - myapp - Visual Studio Code

EXPLORER
OPEN EDITORS
x <% home.ejs views
MYAPP
  bin
  node_modules
  public
  routes
    index.js
    users.js
  views
    <% home.ejs
    index.ejs
    login.ejs
  app.js
  package-lock.json
  package.json

1 Hi, <%= myvalue %>
2
3 <a href="/logout">Logout</a>
```





```
File Edit Selection View Go Debug Terminal Help
index.js - myapp - Visual Studio Code

EXPLORER
1
OPEN EDITORS 1 UNSAVED
  JS index.js routes
MYAPP
  bin
  node_modules
  public
  routes
    JS index.js
    JS users.js
  views
    <% home.ejs
    <% index.ejs
    <% login.ejs
  JS app.js
  package-lock.json
  package.json

45
46 router.get('/logout', function(req, res, next) {
47     req.session.destroy(function(err) {
48         res.redirect('/login');
49     });
50 });
51
52
53
54
55
56
57
58
59
60
61
62
```






127.0.0.1:3000/login x +

← → ↻ 🏠 ⓘ 127.0.0.1:3000/login

Name :





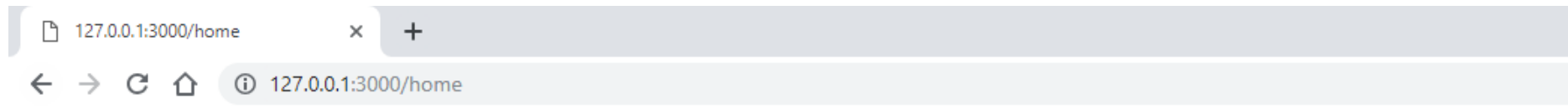
127.0.0.1:3000/login x +

← → ↻ 🏠 ⓘ 127.0.0.1:3000/login

Name :







Hi, Akash [Logout](#)




| Enjoy 😊



# Counter Demo



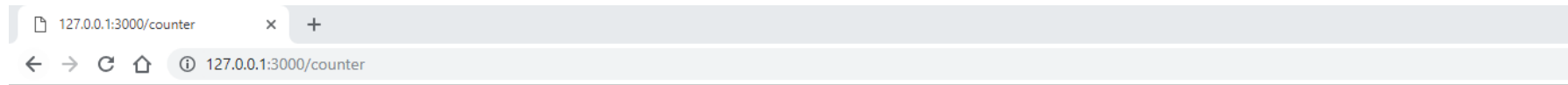


```
File Edit Selection View Go Debug Terminal Help
index.js - myapp - Visual Studio Code

EXPLORER
OPEN EDITORS 1 UNSAVED
  JS index.js routes
MYAPP
  bin
  node_modules
  public
  routes
    JS index.js
    JS users.js
  views
    <%> about.ejs
    <%> contact.ejs
    <%> home.ejs
    <%> index.ejs
    <%> login.ejs
  JS app.js
  package-lock.json
  package.json

51
52 router.get('/counter', function(req, res, next) {
53
54   if (req.session.views) {
55     req.session.views++
56     res.setHeader('Content-Type', 'text/html')
57     res.write('<p>views: ' + req.session.views + '</p>')
58     res.write('<p>expires in: ' + (req.session.cookie.maxAge / 1000) + 's</p>')
59     res.end()
60   } else {
61     req.session.views = 1
62     res.end('welcome to the session demo. refresh!')
63   }
64
65 });
66
67
68
69
70
```



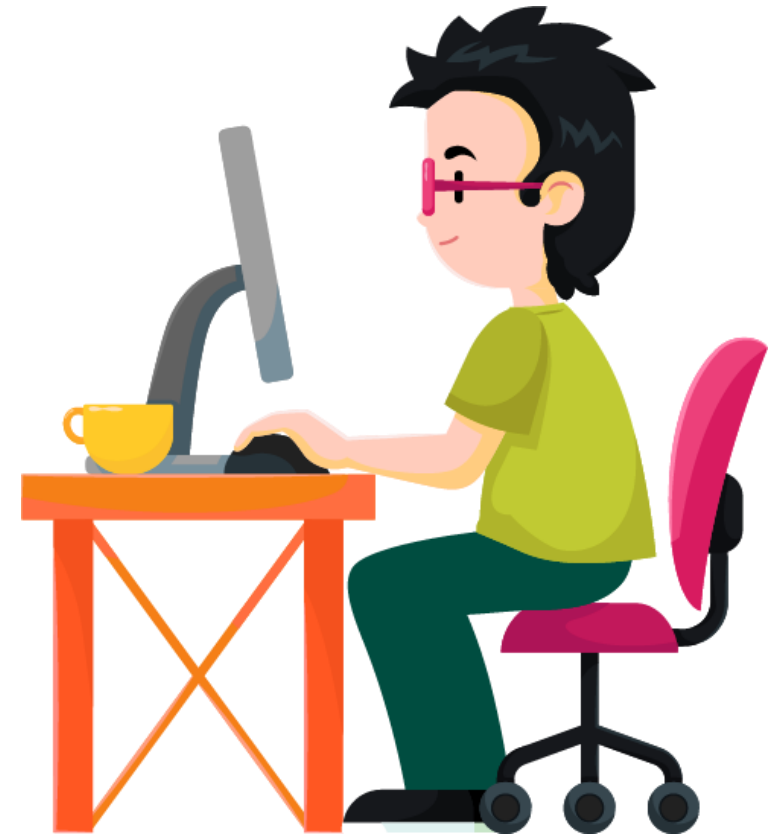


views: 15

expires in: 60s



# Get Exclusive Video Tutorials



[www.apptutorials.com](http://www.apptutorials.com)

<https://www.youtube.com/user/Akashtips>





Get More Details

[www.akashsir.com](http://www.akashsir.com)



# If You Liked It !

## Rating Us Now



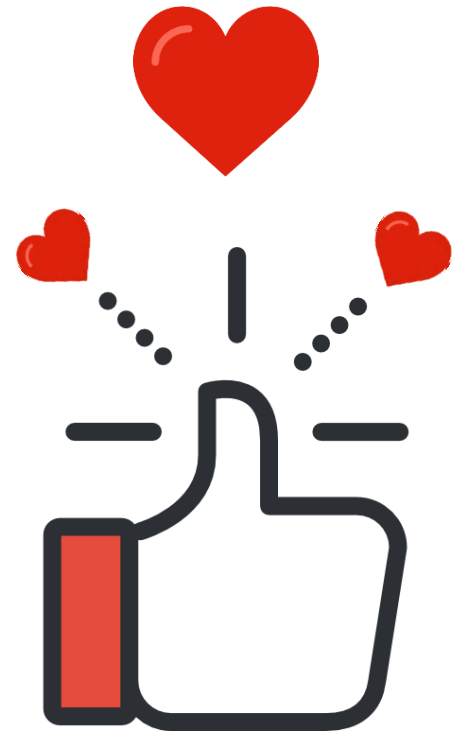
**Just Dial**

[https://www.justdial.com/Ahmedabad/Akash-Technolabs-Navrangpura-Bus-Stop-Navrangpura/079PXX79-XX79-170615221520-S5C4\\_BZDET](https://www.justdial.com/Ahmedabad/Akash-Technolabs-Navrangpura-Bus-Stop-Navrangpura/079PXX79-XX79-170615221520-S5C4_BZDET)



**Sulekha**

<https://www.sulekha.com/akash-technolabs-navrangpura-ahmedabad-contact-address/ahmedabad>





# Connect With Me



Akash Padhiyar  
#AkashSir

[www.akashsir.com](http://www.akashsir.com)  
[www.akashtechlabs.com](http://www.akashtechlabs.com)  
[www.akashpadhiyar.com](http://www.akashpadhiyar.com)  
[www.apptutorials.com](http://www.apptutorials.com)

## # Social Info



Akash.padhiyar



Akashpadhiyar



Akash\_padhiyar



+91 99786-21654



#Akashpadhiyar  
#apptutorials