

## Task1

```
const express = require('express')
const app = express()
const { ROLE, users } = require('./data')
const { authUser, authRole } = require('./basicAuth')
const projectRouter = require('./routes/projects')

app.use(express.json())
app.use(setUser)
app.use('/projects', projectRouter)

app.get('/', (req, res) => {
  res.send('Home Page')
})
//Verify if the user is logged in
app.get('/dashboard', authUser, (req, res) => {
  res.send('Dashboard Page')
})
//Verify if user is Logged in and check if he is admin
app.get('/admin', authUser, authRole(ROLE.ADMIN), (req, res) => {
  res.send('Admin Page')
})

//Giving User ID.
function setUser(req, res, next) {
  const userId = req.body.userId
  if (userId) {
    req.user = users.find(user => user.id === userId)
  }
  next()
}

//Listening the data on 3000 port.
app.listen(3000)
```

## Basic Authentication:

```
//Verify if user is logged in.
function authUser(req, res, next){
  //If not logged in give error 404 and send msg to the user to sign in.
  if(req.user == null){
    res.status(403)
    return res.send('You need to sign in')
  }
}
```

```

    next()
  }
  //Matching User ID with it's Desired roles, if Not match throw error.
  function authRole(role){

    return (req,res,next) =>{
      if(req.user.role !== role){
        res.status(401)
        return res.send('Not allowed')
      }
      next()
    }
  }
  //Getting user and role by using exports export.
  module.exports = {
    authUser,
    authRole
  }

```

## Project Routes:

```

const express = require('express')
const { authUser } = require('../basicAuth')
const router = express.Router()
const { projects } = require('../data')
const { canViewProject,canDeleteProject,scopedProjects } =
require('../permissions/project')

router.get('/', authUser, (req, res) => {
  res.json(scopedProjects(req.user,projects))
})

router.get('/:projectId', setProject, authUser, authGetProject , (req, res) => {
  res.json(req.project)
})

router.delete('/:projectId', setProject, authUser, authDeleteProject,
(req,res)=>{
  res.send('Deleted Project')
})

function setProject(req, res, next) {
  const projectId = parseInt(req.params.projectId)
  req.project = projects.find(project => project.id === projectId)
}

```

```

    if (req.project == null) {
      res.status(404)
      return res.send('Project not found')
    }
    next()
  }

  //Allow Admin and User to View it.
  function authGetProject(req,res,next){
    if(!canViewProject(req.user, req.project)){
      res.status(401)
      return res.send('Not allowed')
    }
    next()
  }

  //Not to Allow the User to Delete it.
  function authDeleteProject(req,res,next){
    if(!canDeleteProject(req.user, req.project)){
      res.status(401)
      return res.send('Not allowed')
    }
    next()
  }
}

```

```
module exports = router
```

## Project Authentication:

```

const { ROLE, projects } = require('../data')

//Verify if the User is admin and then let the view the project route
function canViewProject(user, project){
  return(
    user.role === ROLE.ADMIN ||
    project.userId === user.id
  )
}

//Display user to access the project route
function scopedProjects(user, projects){

```

```

    if(user.ROLE === ROLE.ADMIN) return projects
    return projects.filter(project => project.userId
    === user.id)
}
//Allow User to delete the project.
function canDeleteProject(user,project){
    project.userId === user.id
}

module.exports = {
    canViewProject,
    scopedProjects,
    canDeleteProject
}

```

Users:

```

const ROLE = {
    ADMIN: 'admin',
    BASIC: 'basic'
}

module.exports = {
    ROLE: ROLE,
    users: [
        { id: 1, name: 'Kyle', role: ROLE.ADMIN },
        { id: 2, name: 'Sally', role: ROLE.BASIC },
        { id: 3, name: 'Joe', role: ROLE.BASIC }
    ],
    projects: [
        { id: 1, name: "Kyle's Project", userId: 1 },
        { id: 2, name: "Sally's Project", userId: 2 },
        { id: 3, name: "Joe's Project", userId: 3 }
    ]
}

```

Task2:

Routes:

```

const express = require('express')

```

```

const app = express()
const { ROLE, users } = require('./data')
const { authUser, authRole } = require('./Auth')

app.use(express.json())
app.use(setUser)

app.get('/', authUser ,(req, res) => {
  res.send('Home Page')
})

app.get('/MarkAttendance', authUser, authRole(ROLE.STUDENT), (req, res) => {
  res.send('Mark Attendance Page')
})

app.get('/ViewAttendance', authUser ,(req, res) => {
  res.send('View Attendance Page')
})

function setUser(req, res, next) {
  const userId = req.body.userId
  if (userId) {
    req.user = users.find(user => user.id === userId)
  }
  next()
}

app.listen(3000)

```

## Authentication:

```

function authUser(req,res,next){

  if(req.user == null){
    res.status(403)
    return res.send('You need to sign in')
  }
  next()
}

function authRole(role){

  return (req,res,next) =>{
    if(req.user.role === role){

```

```

        res.status(401)
        return res.send('Not allowed')
    }
    next()
}
}

module.exports = {
  authUser,
  authRole
}

```

Data:

```

const ROLE = {
  ADMIN: 'admin',
  TEACHER: 'teacher',
  STUDENT: 'student'
}

module.exports = {
  ROLE: ROLE,
  users: [
    { id: 1, name: 'Hammad', role: ROLE.STUDENT },
    { id: 2, name: 'Hyder', role: ROLE.STUDENT },
    { id: 3, name: 'Waleed', role: ROLE.STUDENT },
    { id: 4, name: 'Moyu', role: ROLE.STUDENT },
    { id: 5, name: 'Saud', role: ROLE.STUDENT },
    { id: 6, name: 'Daniyal', role: ROLE.ADMIN },
    { id: 7, name: 'Usman Azhar', role: ROLE.TEACHER }
  ]
}

```

Screenshots:

```
test.rest > GET /MarkAttendance
Send Request
1 GET http://localhost:3000/MarkAttendance
2 Content-Type: application/json
3
4 {
5   "userId": 7
6 }
7

1 HTTP/1.1 200 OK
2 X-Powered-By: Express
3 Content-Type: text/html; charset=utf-8
4 Content-Length: 20
5 ETag: W/"14-aDEqNip+5ts3fRRhaktmv2/pjEk"
6 Date: Mon, 16 May 2022 17:34:15 GMT
7 Connection: close
8
9 Mark Attendance Page
```

```
test.rest > GET /MarkAttendance
Send Request
1 GET http://localhost:3000/MarkAttendance
2 Content-Type: application/json
3
4 {
5   "userId": 6
6 }
7

1 HTTP/1.1 200 OK
2 X-Powered-By: Express
3 Content-Type: text/html; charset=utf-8
4 Content-Length: 20
5 ETag: W/"14-aDEqNip+5ts3fRRhaktmv2/pjEk"
6 Date: Mon, 16 May 2022 17:35:25 GMT
7 Connection: close
8
9 Mark Attendance Page
```

```
test.rest > GET /MarkAttendance
Send Request
1 GET http://localhost:3000/MarkAttendance
2 Content-Type: application/json
3
4 {
5   "userId": 5
6 }
7

1 HTTP/1.1 401 Unauthorized
2 X-Powered-By: Express
3 Content-Type: text/html; charset=utf-8
4 Content-Length: 11
5 ETag: W/"b-4DFRMacN7Ydwf4a4DntVvdnCmNc"
6 Date: Mon, 16 May 2022 17:35:44 GMT
7 Connection: close
8
9 Not allowed
```

### Task3:

Permission/project.js:

```
const { ROLE, users } = require("../data");

function canviewattendance(user,viewattendance)
{
    return(user.role === ROLE.ADMIN ||
        viewattendance.userId === user.id )
}

function scopeofviewattendance(user,viewattendences)
{

```

```

    if(user.role == ROLE.ADMIN) return viewattendences
    return viewattendences.filter(viewattendance => viewattendance.userId ==
user.id)
}
function candeleateattendance(user,viewattendences)
{
    return(user.role === ROLE.ADMIN || user.role === ROLE.TEACHER)
}

module.exports = {
    canviewattendance,
    scopeofviewattendance,
    candeleateattendance
}
}

```

## Routes/project.js:

```

const express = require('express')
const router = express.Router()
const { viewattendences } = require('../data')
const { authUser } = require('../Auth')
const { canviewattendance, candeleateattendance, scopeofviewattendance } =
require('../Permissions/project')

router.get('/', authUser, (req, res) => {
    res.json(scopeofviewattendance(req.user, viewattendences))
})

router.get('/:projectId', setProject, authUser, authgetattendance, (req, res) =>
{
    res.json(req.viewattendance)
})

router.delete('/:projectId', setProject, authUser, authdeleting, (req, res) => {
    res.send('Deleted attendance')
})

function setProject(req, res, next) {
    const projectId = parseInt(req.params.projectId)
    req.viewattendance = viewattendences.find(project => project.id === projectId)

    if (req.viewattendance == null) {

```



```

        res.status(404)
        return res.send('Attendance not found')
    }
    next()
}
function authgetattendance(req,res,next){
    if(!canviewattendance(req.user,req.viewattendance))
    {
        res.status(401)
        return res.send('NOT allowed')
    }
    next()
}
function authdeleting(req,res,next){
    if(!candeleteattendance(req.user,req.viewattendance))
    {
        res.status(401)
        return res.send('NOT allowed')
    }
    next()
}
module.exports = router

```

Auth.js:

```

function authUser(req,res,next){

    if(req.user == null){
        res.status(403)
        return res.send('You need to sign in')
    }
    next()
}
function authRole(role){

    return (req,res,next) =>{
        if(req.user.role === role){
            res.status(401)
            return res.send('Not allowed')
        }
        next()
    }
}

```

```
module.exports = {  
  authUser,  
  authRole  
}
```

## Data.js:

```
const ROLE = {  
  ADMIN: 'admin',  
  TEACHER: 'teacher',  
  STUDENT: 'student'  
}  
  
module.exports = {  
  ROLE: ROLE,  
  users: [  
    { id: 1, name: 'Hammad', role: ROLE.STUDENT },  
    { id: 2, name: 'Daniyal', role: ROLE.STUDENT },  
    { id: 3, name: 'Waleed', role: ROLE.STUDENT },  
    { id: 4, name: 'Moyu', role: ROLE.STUDENT },  
    { id: 5, name: 'Saud', role: ROLE.STUDENT },  
    { id: 6, name: 'Hyder', role: ROLE.ADMIN },  
    { id: 7, name: 'Usman Azhar', role: ROLE.TEACHER }  
  ],  
  viewattendances: [  
    { id: 1, name: 'Hammad attendance', userId: 1 },  
    { id: 2, name: 'Daniyal attendance', userId: 2 },  
    { id: 3, name: 'Waleed attendance', userId: 3 },  
    { id: 4, name: 'Moyu attenddance', userId: 4 },  
    { id: 5, name: 'Saud attenddance', userId: 5 },  
    { id: 6, name: 'Hyder attendance', userId: 6 },  
    { id: 7, name: 'Usman Azhar attendance', userId: 7 }  
  ]  
}
```

## Server.js:

```
const express = require('express')  
const app = express()  
const { ROLE, users } = require('./data')  
const { authUser, authRole } = require('./Auth')
```

```

const projectRouter = require('./routes/projects')
app.use(express.json())
app.use(setUser)

app.use('/ViewAttendance',projectRouter)

app.get('/', authUser ,(req, res) => {
  res.send('Home Page')
})

app.get('/MarkAttendance', authUser, authRole(ROLE.STUDENT), (req, res) => {
  res.send('Mark Attendance Page')
})

function setUser(req, res, next) {
  const userId = req.body.userId
  if (userId) {
    req.user = users.find(user => user.id === userId)
  }
  next()
}

app.listen(3000)

```

Output: Slight Changing in Output instead of using /projects use /ViewAttendance ← just to show that route is used for viewing of attendance.... /projects was used to compromise with the editing and testing of certain permissions after tests are done it's route is now:

<http://localhost:3000/ViewAttendance/>

Admin:

```
Send Request
1 GET http://localhost:3000/projects
2 Content-Type: application/json
3
4 {
5   "userId": 6
6 }
7

1 HTTP/1.1 200 OK
2 X-Powered-By: Express
3 Content-Type: application/json; charset=utf-8
4 Content-Length: 331
5 ETag: W/"14b-wK5rRo92Bj0DrLC26tGnTZSNmDg"
6 Date: Mon, 16 May 2022 19:58:55 GMT
7 Connection: close
8
9 [
10  {
11    "id": 1,
12    "name": "Hammad attendance",
13    "userId": 1
14  },
15  {
16    "id": 2,
17    "name": "Daniyal attendance",
18    "userId": 2
19  },
20  {
21    "id": 3,
22    "name": "Waleed attendance",
```

```
Send Request
1 GET http://localhost:3000/projects/1
2 Content-Type: application/json
3
4 {
5   "userId": 6
6 }
7

1 HTTP/1.1 200 OK
2 X-Powered-By: Express
3 Content-Type: application/json; charset=utf-8
4 Content-Length: 46
5 ETag: W/"2e-PiflUmohKY8NYJE+PRL+wMycfpU"
6 Date: Mon, 16 May 2022 19:53:33 GMT
7 Connection: close
8
9 {
10   "id": 1,
11   "name": "Hammad attendance",
12   "userId": 1
13 }
```

```
Send Request
1 DELETE http://localhost:3000/projects/1
2 Content-Type: application/json
3
4 {
5   "userId": 6
6 }
7

1 HTTP/1.1 200 OK
2 X-Powered-By: Express
3 Content-Type: text/html; charset=utf-8
4 Content-Length: 18
5 ETag: W/"12-WM1GU/RIiEL12K0kE3ovzfkbaRI"
6 Date: Mon, 16 May 2022 19:54:09 GMT
7 Connection: close
8
9 Deleted attendance
```

Teachers:

```
Send Request
1 DELETE http://localhost:3000/projects/3
2 Content-Type: application/json
3
4 {
5   "userId": 7
6 }
7

1 HTTP/1.1 200 OK
2 X-Powered-By: Express
3 Content-Type: text/html; charset=utf-8
4 Content-Length: 18
5 ETag: W/"12-WM1GU/RIiEL12K0kE3ovzfkbaRI"
6 Date: Mon, 16 May 2022 19:54:34 GMT
7 Connection: close
8
9 Deleted attendance
```

```
Send Request
1 GET http://localhost:3000/projects/3
2 Content-Type: application/json
3
4 {
5   "userId": 7
6 }
7

1 HTTP/1.1 200 OK
2 X-Powered-By: Express
3 Content-Type: application/json; charset=utf-8
4 Content-Length: 46
5 ETag: W/"2e-QaxD7pNplxhjFxttUsMZLjHHY"
6 Date: Mon, 16 May 2022 19:57:33 GMT
7 Connection: close
8
9 {
10   "id": 3,
11   "name": "Waleed attendance",
12   "userId": 3
13 }
```

```
test.rest > ...
Send Request
1 GET http://localhost:3000/projects
2 Content-Type: application/json
3
4 {
5   "userId": 7
6 }
7

1 HTTP/1.1 200 OK
2 X-Powered-By: Express
3 Content-Type: application/json; charset=utf-8
4 Content-Length: 331
5 ETag: W/"14b-wK5rRo92Bj0DrLC26tGnTZSNmDg"
6 Date: Mon, 16 May 2022 20:00:28 GMT
7 Connection: close
8
9 [
10   {
11     "id": 1,
12     "name": "Hammad attendance",
13     "userId": 1
14   },
15   {
16     "id": 2,
17     "name": "Daniyal attendance",
18     "userId": 2
19   },
20   {
21     "id": 3,
22     "name": "Waleed attendance",

```

## Students:

```
Send Request
1 GET http://localhost:3000/projects/2
2 Content-Type: application/json
3
4 {
5   "userId": 1
6 }
7

1 HTTP/1.1 401 Unauthorized
2 X-Powered-By: Express
3 Content-Type: text/html; charset=utf-8
4 Content-Length: 11
5 ETag: W/"b-VtvSBNh4hHKiTjZFES2EyjmI73A"
6 Date: Mon, 16 May 2022 19:58:02 GMT
7 Connection: close
8
9 NOT allowed
```

```
Send Request
1 GET http://localhost:3000/projects/3
2 Content-Type: application/json
3
4 {
5   "userId": 1
6 }
7

1 HTTP/1.1 401 Unauthorized
2 X-Powered-By: Express
3 Content-Type: text/html; charset=utf-8
4 Content-Length: 11
5 ETag: W/"b-VtvSBNh4hHKiTjZFES2EyjmI73A"
6 Date: Mon, 16 May 2022 19:58:17 GMT
7 Connection: close
8
9 NOT allowed
```

```
Send Request
1 GET http://localhost:3000/projects/1
2 Content-Type: application/json
3
4 {
5   "userId": 1
6 }
7

1 HTTP/1.1 200 OK
2 X-Powered-By: Express
3 Content-Type: application/json; charset=utf-8
4 Content-Length: 46
5 ETag: W/"2e-PiflUmoHkY8NYJE+PRl+wMycfpU"
6 Date: Mon, 16 May 2022 19:58:32 GMT
7 Connection: close
8
9 {
10   "id": 1,
11   "name": "Hammad attendance",
12   "userId": 1
13 }
```

-----END-----

-----