Express JS **(For logical reference only)**

Miscellaneous Tasks

Task: Write an ExpressJS to take a UserName, Password, Textarea for “message” & submit button using get method.

1) After clicking submit button the content of submitted details should be represented on “/login” page along with one “show vowel” link.

2) By clicking “show vowel” link count of vowel used in submitted “message” will display on “/message” page. (Use next() to route page)

Solution:

```
var expr = require("express");
var app = expr();

app.use(expr.static("../public",{index:'index.html'}));

app.get("/login",(req,res,next)=>{
    console.log(req.query);
    res.set("content-type","text/html");
    res.write("<center><h1>Welcome " + req.query.Username + "</h1>");
    res.write("<center><h1> Your Password is " + req.query.Password + "</h1>");
    res.write("<center><h2>Your message id is " + req.query.message+ "</h2>");
    res.write("<h3>You can Find vowel in message by clicking</h3><a
href='/message'>Message</a></center>");
    a=req.query.message
    next();
})

app.get("/message",(req,res)=>{
   res.set("content-type","text/html");
   res.write("<center><h2>Vowel in Your message id : " + a + "</h2>");
    const string =a
    const vowels = ["a", "e", "i", "o", "u"]
    function countVowel(str) {
        let count = 0;
            for (let letter of str.toLowerCase()) {
            if (vowels.includes(letter)) {
                count++;
            }
        }
        return count
    }
    const result = countVowel(string);
    res.write(`${result}`);
    res.send()
})
app.listen(5121,()=>
{
    console.log("server start");
});
```

Task:

Write an express js script to configure the multer middleware. Perform following tasks.

1) Create a html form file named "form.html" in public folder. This file contain centrally oriented heading(h3) "Upload your File" in red color with background-color aqua. Along with choose file option(to browse and select file) and submit button(to upload the file). (Must use external css having name “effect.css” in public folder)

2) Create a js file to show result after uploading any type of file, message should be displayed on "/upload" page "(file original name) has been uploaded".(Css effect must include while running js code)

3) Save uploaded files to specific directory named "File". And in this folder file must be stored in format of "data-file.pdf" where "data" is the field name.

Solution

App.js (In src folder)

```
const express = require('express');
const multer = require('multer');
const path = require('path');

// Initialize the app
const app = express();

// Serve static files from the public folder
app.use(express.static("../public",{index:"form.html"}))

// Set up storage using multer
const storage = multer.diskStorage({
    destination: "File",
    filename: function (req, file, cb) {
        cb(null, file.fieldname+"-file.pdf");
    }
});

// Initialize upload middleware
```

```
const upload = multer({ storage: storage }).single('data')

// Handle file upload
app.post('/upload', upload, (req, res) => {
   if (!req.file) {
      return res.status(400).send('No file uploaded.');
   }
   res.send(`<p>${req.file.originalname} has been uploaded</p><a href="/">Go back</a>`);
});

// Start the server
const PORT = process.env.PORT || 3000;
app.listen(PORT, () => {
   console.log(`Server is running on port ${PORT}`);
});
```

form.js (In public folder)

```
<!DOCTYPE html>
<html lang="en">
<head>
   <meta charset="UTF-8">
   <meta name="viewport" content="width=device-width, initial-scale=1.0">
   <title>File Upload</title>
   <link rel="stylesheet" href="/effect.css">
</head>
<body>
   <div class="container">
      <h3>Upload your File</h3>
      <form action="/upload" method="post" enctype="multipart/form-data">
         <input type="file" name="data" required>
         <button type="submit">Submit</button>
      </form>
   </div>
</body>
</html>
```

effect.css(In public folder)

```
body {  background-color: aqua;}

h3 {  color: red;}
//optional
.container {  text-align: center;}
```

Task: Write a code to set up nodemailer in Express.js. Sender email id: “lju@gmail.com”. Receiver email ids: “student@gmail.com and faculty@gmail.com”. Mail subject should be “LJ University” Mail body contains “Welcome Student” in h3 tag and in table display data Date 28/06/23, Exam name FSD-2.

Solution:

```
const nodemailer = require('nodemailer');
const transporter = nodemailer.createTransport({
   service: 'gmail',
   auth: {
      user: 'lju@gmail.com',
      pass: 'yourpassword'
   }
});

      const mailOptions = {
         from: 'lju@gmail.com',
         to: 'student@gmail.com, faculty@gmail.com',
         subject: 'LJ University',
         html: '<h3>welcome student</h3><table border="1px"><tr><th>date</th><th>Exam
name</th></tr><tr><td> 28/06/23</td><td>FSD-2</td></tr>'
      };
      transporter.sendMail(mailOptions, (error, info) => {
         if (error) {
            console.log(err)
         }
        console.log("mail send")
      });
```

Task: Write an express js script to configure the multer middleware. Perform following tasks. 1) Create a pug file named "file.pug". This file contains heading(h3) "Upload your CV" in red color. And, a form with input type file(to browse and select file) and submit(to upload the file). 2) Create a js file named "file.js" and link this js and pug file to browse pug file on "/home" page. 3) After uploading a file display message on "/upload" page "(file original name) has been uploaded". 4) Save uploaded files to specific directory named "upload". And in this folder file must be stored in format of "lju-file.pdf" where "lju" is the field name.

Solution:

xyz.pug

```
html
body
 form(action="/upload" method="post" enctype="multipart/form-data")
  input(type="file" name="lju")
  input(type="submit" value="upload")
```

xyz.js

```
const express = require('express');
const multer = require('multer');
const app = express();
app.set('view engine','pug')
app.set('views',__dirname)
const storage = multer.diskStorage({
    destination:'file',
    filename: function (req, file, cb) {
        cb(null, file.fieldname+'-file.pdf');
    }
});
const upload = multer({ storage: storage });
app.get("/",(req,res)=>
{
    res.render('xyz')
})

app.post('/upload', upload.single('lju'), (req, res) => {

        res.send(`<h3 style="text-align: center;">${req.file.originalname} has been uploaded</h3>`);

}).listen(3000);
```

Task: Write an expressJS code in which RESTapi is created for json object named Places I love which contains name,country,state,city and rating out of 10(no decimal points) is given.upon passing ratings on the browser it should display the places having that rating.i.e. on localhost:30001/a/10 should display all the places having 10 ratings.

Solution:

api.js

```
const expr = require("express");
const router = expr.Router();
const PlacesIlove=[
{name:’manali’,country:’india’,state:’jammu’,city:’manali’,rating:8},
{name:’kullu’,country:’india’,state:’jammu’,city:’manali’,rating:9},
{name:’kashmir’,country:’india’,state:’jammu’,city:’manali’,rating:10}]
module.exports = router;
router.get("/:rating([0-9]{1,2})",(req,res)=>{
var current_data = data.filter((i1)=>{
if(i1.rating == req.params.rating){
return true;
}
})
if(current_data.length >0){
res.send(current_data);
}else{
res.send("Not Found")
}
})
```

main.js

```
const expr = require("express");
const app = expr();
const api = require("./api")
app.use("/a",api);
app.listen(30001);
```