**JSON Processing**

**Write an express js script to define one JSON array of 3 objects having properties name and age. Short these objects according to age.**

**If user request sort page in url then all names along with age should be printed according to descending order of age.**

**Also, display these sorted values on “Sort page” and display JSON object on “Home page”**

const expr = require("express")

const app = expr();

student = [{name:"abc",age:28},

{name:"def",age:40},

{name:"xyz",age:50}]

**//home page**

app.get ("/",(req,res)=>{

res.set ("content-type","text/html")

res.send (student)

})

**//sort page**

app.get ("/sort",(req,res)=>{

res.set ("content-type","text/html")

**var des = student.sort((a,b)=>b.age-a.age)**

console.log(des)

for (k=0;k<des.length;k++){

res.write ("<center><h2>"+des[k].name+ " = " +des[k].age +"</h2></center>")

}

res.send ()

})

app.listen (5200)

**OR**

const expr = require("express")

const app = expr();

student = [{name:"abc",age:28},

{name:"def",age:30},

{name:"xyz",age:50}]

**//home page**

app.get ("/",(req,res)=>{

res.set ("content-type","text/html")

res.send (student)

})

**//sort page**

app.get ("/sort",(req,res)=>{

res.set ("content-type","text/html")

**for (i=0;i<student.length;i++){**

**for (j=0;j<student.length;j++){**

**if (student[i].age>student[j].age){**

**temp = student[i];**

**student[i] = student[j];**

**student[j] = temp**

**}**

**}**

**}**

for (k=0;k<student.length;k++){

res.write ("<center><h2>"+student[k].name+ " = " +student[k].age +"</h2></center>")

}

res.send ()

})

app.listen (5200)

**Output:**

**On home page**

[{"name":"abc","age":28},{"name":"def","age":40},{"name":"xyz","age":50}]

**On sort page**

## xyz = 50

## def = 40

## abc = 28

**Get Method**

1. **Write express script to get form data using get method and display data in JSON format in next page named “process\_get”**
2. **Write express js script to print message in next line splitting by “.” And use get method to submit the data. HTML file contains form of text area for the message and submit button.**
3. **Write express js script to perform tasks as asked below.**
4. **Create one HTML file which contains two number type input fields, one dropdown which contains options like (select, addition, subtraction, multiplication, division) and one submit button.**
5. **The input fields must contain the value greater than 0 else it will give a message “Please enter the valid number”. Also, user must select any of the formula from the dropdown else give a message “You have not selected any formula”. (Message will be displayed on “/calc” page.)**
6. **If one formula is selected and numbers are entered then respective calculations will be performed on the page “/calc”.**
7. **Use get method to request data.**

**Post Method**

1. **Write express script to get form data using post method and display data in JSON format in next page named “process”**
2. **Write express js script to perform the tasks as asked below.**
3. **Create one HTML file named ljform.html and add one form which contains username, password and submit button. Data should be submitted by HTTP post method. Submit button is of black color with white text. (External CSS).On home page form should be displayed and while submitting the form, on next page named “/login” if username is admin then it will display “Welcome admin” else display “Please login with admin name”.**

**Middleware**

1. **Write express js script to perform following tasks.**
2. **Create one html file which contains one text field for name, email field and checkbox for subscription. Html file will be loaded on home page. Email and name fields are required fields.**
3. **On login page welcome user and email id data should be printed.**
   1. **If user checked the subscription then “Thank you for the subscription” message will be printed and “logout” link will be displayed under the message. If user clicks logout link then he/she will be redirected to the home page.**
   2. **If user has not opted for the subscription then “You can subscribe to get daily updates” message will be printed and “subscribe” link will be displayed under the message. If user clicks subscribe link then he/she will be redirected to the subscription page. In this page “Thank you for the subscription” message will be printed and “logout” link will be displayed under the message. If user clicks logout link then he/she will be redirected to the home page.**

**Use concept of the middleware and you can use any of http methods(get/post).**

1. **Write an express.js script to load an HTML file having username and password and submit button. On clicking submit button. It should jump on "check" page using "POST" method. If username is "admin" , then jump on next middleware to print "welcome… admin" , if username is not "admin" , then stay on same middleware to print "warning msg" in red color.**

**Cookie**

**1.Example:**

**Write an express js script to set cookies of submitted values of form. Perform following tasks.**

* **Create a HTML file which contains a form with fields first name, last name, password and a submit button.**
* **Once form submitted, store all these entered values to the respective cookies on ‘/next’ page.**
* **Then redirect user to “/admin” page and clear the cookie set for the last name. Display remaining set cookie values on this page. (Using post method)**

**2. Example**

**You have been assigned to develop a user feedback form for a website using Express.js and cookies.**

**Implement the following requirements:**

* **Create a form with the following fields:**
  + **Name (input field)**
  + **Email (input field)**
  + **Message (textarea field)**
  + **Rating (radio buttons: Bad, Average, Good, Very Good, Excellent)**
* **When the user submits the form, store their feedback information (name, email, message, and rating) in a cookie named "feedback" that expires in 10 seconds.**
* **Display a confirmation message to the user after successfully submitting the form & Create a link to display the feedback details stored in the "feedback" cookie.**
* **When the user click to the link, retrieve the feedback information from the cookie and display it on the page also include a link on the feedback details page to Logout.**
* **When the user clicks the link, user redirected to home page.**

**Session**

1. **Write express js script using session to display how many times a user visited a website. If user is visiting a website for the first time then display “Welcome! Thank you for visiting our website!” else display the count of user (How many times) for that particular session.**
2. **write a script to meet following requirements:**
   1. **Create index.html file page which contains form(username,password,login button). and open it on localhost.**
   2. **After clicking submit button, it should jump on “savesession” page. Store username and password in session.**
   3. **After saving session, redirect to “fetchsession” page and read value. Put a LOGOUT link button here.**
   4. **Jump on delete session page after clicking LOGOUT button.**
   5. **Destroy the session on this page and redirect to index.html page.**
3. **write a script to meet following requirements:**
   1. **Create session.html file page which contains form(username,password,login button). and open it on localhost.**
   2. **After clicking submit button, it should jump on “save” page. Store username and password in session.**
   3. **After saving session, redirect to “fetchdata” page and read value. On this page check authentication of user. User name and password must be “admin” and “admin@123” respectively.** 
      1. **If this condition is true then display welcome admin and display logout link on this page(fetchdata).**
         1. **By clicking on logout link user should jump to “destroy” page and destroy the session there and display the message “Session destroyed”. And give the link of “login” under that message. By clicking that link user will be redirected to the home page.**
      2. **Else display “Please enter valid username and password” and login link on this page(fetchdata).**