SSL (nodejs)

DB (MySql , Mongodb)

Tokenization

TESTING

Sending mail

ROUTING – pages /views ->

ADMIN SIDE , EMPLOYEE SIDE LOGIN

1. Show all items of array
2. Show array items with keys

DATABASE CONNECTION

ENV FILE –

WHOLE IN CONFIG FILE

1. ~~APPLICATION (app.js)~~
2. ~~AUTHENTICATION~~ ~~-(login , then redirect)~~
3. BROADCASTING
4. CACHE
5. CORS -?
6. Database
7. FILESYSTEMS
8. Hashing
9. Logging
10. Mail
11. Queue
12. API tokens –(personal access token ) without OAuth
13. SERVICES
14. ~~Session - (tracking)- ( basics understood , practice needed)~~
15. View

ROUTING

* Bootstrap – use = to load config , cache (temp. memory)
* Pass data , cookies
* Setting Ip address + localhost to ‘host’
* Form with routing
* Attaching cookies , headers – ( cookies are sent from headers from server to client )
* Routing parameters , basic routing
* Redirecting
  + named routes
  + controlled actions
  + external domains
* FORM validation
* COOKIES , SESSION

Route.get(‘gallery/videos’);

Route.get(‘gallery/photos’);

OR

Route.prefix(‘gallery’)->group(function() {

Route.get(‘photos’);

Route.get(‘videos’);

PROBLEM

* ~~Key-value array output in js – done (Object.keys(arr)); -(DONE) –( D:\WEBSITE DESIGNING\nodejs\BASIC PROGRAMMING\keyarr.js)~~
* ~~Database connection and saving data to database (make same schema name and variables match)~~
* ~~Redirect page from register to login (res.redirect(“login”))~~
* ~~Redirect with id , route.get(‘postid’,1) (D:\WEBSITE DESIGNING\nodejs\NODE WITH PROJECTS\Login page\server\index.js)~~
* Adding authorization middle ware at needed places
* Finding data from database using params

REDIRECT

Route:get(‘/dashboard’, function () ) {

resredirect(‘home/dashboard’);

});

ADDING AUTHORIZATION

Router.get(‘profile’ , [Usercontroller::class , ‘show’] ->middleware(‘auth’);

CONTROLLERS

SHOW PARTICULAR POST- control application

How to create middlewares

COOKIES

setcookie() - function to set cookie

ACCESSING COOKIES

Check if cookies enabled

Modify cookie value

Removing cookies

e.g.-

SETTING UP COOKIE

cookie\_name = ‘adhiraj’

cookie\_value = ‘ lpu student ‘

setcookie(cookie\_name , cookie\_value , time() + 3000)

if( isset ( COOKIE (cookie\_name) ) )

{

Print ( cookie\_name , “set”);

}

Else print ( cookie\_name , “not set “);

SESSION

Deleting data when exit the screen

Session\_destroy() to delete data stored

// SETTING UP LOCAL STORAGE COOKIE , NEVER EXPIRES , SAVED ON BROWSER

localStorage.setItem("name", "adhiraj");

localStorage.getItem("name");

localStorage.removeItem("name");

// ----------------SESSION ----------------

// SESSION STORAGE COOKIE , WORKS TILL TAB IS OPEN , SAVED ON BROWSER THEN GETS DELETED

sessionStorage.setItem("name", "ADHIRAJ");

let data = sessionStorage.getItem("name");

sessionStorage.removeItem("name");

sessionStorage.clear();

// Get the text field that we're going to track

let field = document.getElementById("field");

// See if we have an autosave value

// (this will only happen if the page is accidentally refreshed)

if (sessionStorage.getItem("autosave")) {

  // Restore the contents of the text field

  field.value = sessionStorage.getItem("autosave");

}

// Listen for changes in the text field

field.addEventListener("change", function() {

  // And save the results into the session storage object

  sessionStorage.setItem("autosave", field.value);

});

//------------------COOKIES----------------

// COOKIES SAVED ON BROWSER AND SERVER , SAVES  INFO

document.cookie = 'name=adhiraj ; expires = '+ new Date(2020,1,3).toUTCString();

document.cookie = 'lname=pratap singh; expires = '+ new Date(9999,1,1).toUTCString();

console.log(document.cookie);

FORMS

FORM VALIDATION

COOKIES

O

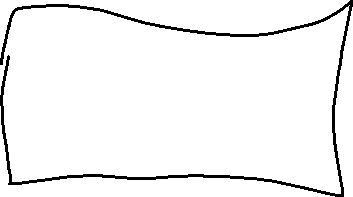


STUDENT MANAGEMENT SSTEM

PROJECT PRESENT ABSENT TOTAL STUDENTS PRESENT



ATTENDANCE FORM DATE



LOGIN PAGE -

KEY TERMS

BODY PARSER – parses data of body , to get access to data we need body parser middleware.

COOKIES - attaching cookies

We can use .env file to make connection with database

Call by value , call by reference

Strict

Params -parameters

REGEX

Preg\_match

Match\_all

Many others ……..

FILTERS

VALIDATING DATA - CHECKING IF DATA IS IN PROPER FORMAT

SANITIZING DATA – REMOVE ILLEGAL CHARACTER FROM DATA

Filter\_var() – in PHP – validate and sanitzation

Sanitize and validate -> email , URL

VALIDATE – integer

Sanitize – String

MIDDLEWARE

Using middleware inside function

Function abc() {

this-> middleware(‘auth’);

this-> middleware(‘log’)->only(‘index’);

this-> middleware(‘subscribed’)->except(‘store’);

}

ERROR HANDLING

If condition satisfy – return some value

If condition fail – return some value

exit() – print and exit the script

if ( file\_exists(“myfile.txt”)) {

file = open(“mytestfile.txt”,”r”);

} else {

Exit(“error: happened”);

}

CUSTOM ERROR HANDLER

Error\_function(error\_level , error\_messaage , -(required parts)

error\_file , error\_line , error\_context ) -(optional parts)

SETTING ERROR HANDLER

OOPS

OBJECT

CLASSS

INHERITANCE

POLYMORPHISM

ABSTRACTION

ENCAPSULATION

PROJECT/APPLICATIONS

1. TODO APP
   1. Add item – delete , edit item