**PRACTICAL-4**

**AIM:** Basic Login System using mongodb as a database and express,body-Parser,cookie and session

**CODE:**

* Model/User.js

const mongoose = require('mongoose')

const bcrpyt = require('bcrypt')

mongoose.connect("mongodb://localhost:27017/LoginDemo",{

    useNewUrlParser:true,

    useUnifiedTopology:true,

    useCreateIndex:true

})

const userSchema=mongoose.Schema({

    username:{

        type:String,

        unique:true,

        required:true

    },

    email:{

        type:String,

        unique:true,

        required:true

    },

    password:{

        type:String,

        required:true

    }

})

// useful function

userSchema.pre("save",function(next){

    if(!this.isModified("password")){

        return next();

    }

    this.password=bcrpyt.hashSync(this.password,12)

    // bcrpyt.hashSync(this.password,12)

    next()

})

userSchema.methods.comparePassword=function(plaintext,callback){

    return callback(null,bcrpyt.compareSync(plaintext,this.password))

        // bcrypt.compareSync(plaintext,this.password))

}

const userModel=mongoose.model("LoginUSer",userSchema) //name of collections

module.exports=userModel

* App.js

const express = require('express')

const bodyParse = require('body-parser')

const cookieParser = require('cookie-parser')

const session = require('express-session')

// database

var User=require("./Model/User")

var app = express();

app.set('port',5555);

// For lower version

// app.use(bodyParser.urlencoded({

//     extended:true

// }))

// Express 4.16 and above

app.use(express.urlencoded({

    extended:true

}))

app.use(cookieParser())

app.use(session({

    key:"user\_id",

    secret:"RandomSecretKey",

    resave:false,

    saveUninitialized:false,

    cookie:{

        expires:600000,

    },

}))

var sessionChecker=(req,res,next)=>{

    if(req.session.user && req.cookies.user\_id){

        res.redirect("/dashboard")

    }  //is there any session created for user ? and is there any cookie created at server sie for that user

    else{

        next()

    }

}

//routes

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

    res.redirect("/login")

})

app.route("/login")

.get(sessionChecker,(req,res)=>{

    res.sendFile(\_\_dirname+"/public/login.html")

})

.post(async(req,res)=>{

    var username = req.body.username,

    password = req.body.password

    try{

        var user=await User.findOne({username:username}).exec(); //Select username where loginuser='Nikita'

        if(!user){

            res.send("User doesn't exist")

            // res.send('<script>alert("USer doesn\'t matched")</script>')

            res.redirect("/login")

        }

        else{

            // res.send("User exists")

            // res.send('<script>alert("USer MAtched")</script>')

            user.comparePassword(password,(error,match)=>{

                if(!match){

                    res.send("User exists but password doesn't matched");

                    // res.send('<script>alert("Password doesnt matched")</script>')

                    res.redirect("/login")

                }

                res.send("Login Successfully");

                // res.send('<script>alert("password Matched")</script>')

            })

            req.session.use=user;

            res.redirect("/dashboard")

        }

    } catch(error){

       console.log(error);

    }

})

app.route("/signup")

.get(sessionChecker,(req,res)=>{

    res.sendFile(\_\_dirname+"/public/Signup.html")

})

.post((req,res)=>{

    var user=new User({

        username:req.body.username,

        email:req.body.email,

        password:req.body.password

    })

    user.save((err,doc)=>{

        if(err){

            console.log(err);

            res.redirect("/signup")

        }else{

            req.session.user=doc

            res.redirect("/dashboard")

        }

    })

})

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

    if(req.session.user && req.cookies.user\_id){

        res.sendFile(\_\_dirname+"/public/dashboard.html")

    }else{

        res.redirect("/login")

    }

})

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

    if(req.session.user && req.cookies.user\_id){

       res.clearCookie("user\_id")

       res.redirect("/")

    }else{

        res.redirect("/login")

    }

})

app.use(function(req,res,next){

    res.status(404).send("Sorry can't found the resources you want")

})

app.listen(app.get('port'),()=>

    console.log(`Login Demo running on port no: ${app.get('port')}`)

)

* dashboard.html

<!DOCTYPE html>

<html lang="en">

<head>

    <meta charset="UTF-8">

    <meta http-equiv="X-UA-Compatible" content="IE=edge">

    <meta name="viewport" content="width=device-width, initial-scale=1.0">

    <title>Login Here</title>

    <link rel="stylesheet" href="https://maxcdn.bootstrapcdn.com/bootstrap/3.3.7/css/bootstrap.min.css">

</head>

<body class="container">

    <div class="page-header">

        <h1>Basic Login System</h1>

    </div>

    <nav class="navbar navbar-default">

        <div class="container-fluid">

            <!-- Collect the nav-links and other content for toggling -->

            <div class="collapse navbar-collapse" id="bs-example-navbar-collapse-1">

                <ul class="nav navbar-nav">

                    <li><a href="/">Home</a></li>

                    <li><a href="/signup">Sign Up</a></li>

                    <li><a href="/dashboard">Dashboard</a></li>

                </ul>

                <ul class="nav navbar-nav navbar-right">

                    <li><a href="/login">Log In</a></li>

                    <li><a href="/logout">Log Out</a></li>

                </ul>

            </div><!--/.navbar-collapse-->

        </div><!-- /.contaner-fluid-->

    </nav>

<div class="container row">

    <h1>Hi,Welcome to your Dashboard</h1>

</div>

</body>

</html>

* Login.html

<!DOCTYPE html>

<html lang="en">

<head>

    <meta charset="UTF-8">

    <meta http-equiv="X-UA-Compatible" content="IE=edge">

    <meta name="viewport" content="width=device-width, initial-scale=1.0">

    <title>Login Here</title>

    <link rel="stylesheet" href="https://maxcdn.bootstrapcdn.com/bootstrap/3.3.7/css/bootstrap.min.css">

</head>

<body class="container">

    <div class="page-header">

        <h1>Basic Login System</h1>

    </div>

    <nav class="navbar navbar-default">

        <div class="container-fluid">

            <!-- Collect the nav-links and other content for toggling -->

            <div class="collapse navbar-collapse" id="bs-example-navbar-collapse-1">

                <ul class="nav navbar-nav">

                    <li><a href="/">Home</a></li>

                    <li><a href="/signup">Sign Up</a></li>

                    <li><a href="/dashboard">Dashboard</a></li>

                </ul>

                <ul class="nav navbar-nav navbar-right">

                    <li><a href="/login">Log In</a></li>

                    <li><a href="/logout">Log Out</a></li>

                </ul>

            </div><!--/.navbar-collapse-->

        </div><!-- /.contaner-fluid-->

    </nav>

    <div class="container row">

        <div class="jumbotron col-sm-4 pull-center">

            <form action="/login" method="POST">

                <div class="form-group">

                    <label>Username:</label>

                    <input class="form-control" required type="text" name="username"/>

                </div>

                <div class="form-group">

                    <label>Password:</label>

                    <input class="form-control" required type="password" name="password"/>

                </div>

                <div class="form-group">

                    <input class="btn btn-primary"  type="submit" value="Log In"/>

                </div>

            </form>

        </div>

    </div>

</body>

</html>

* Signup.html

<!DOCTYPE html>

<html lang="en">

<head>

    <meta charset="UTF-8">

    <meta http-equiv="X-UA-Compatible" content="IE=edge">

    <meta name="viewport" content="width=device-width, initial-scale=1.0">

    <title>Login Here</title>

    <link rel="stylesheet" href="https://maxcdn.bootstrapcdn.com/bootstrap/3.3.7/css/bootstrap.min.css">

</head>

<body class="container">

    <div class="page-header">

        <h1>Basic Login System</h1>

    </div>

    <nav class="navbar navbar-default">

        <div class="container-fluid">

            <!-- Collect the nav-links and other content for toggling -->

            <div class="collapse navbar-collapse" id="bs-example-navbar-collapse-1">

                <ul class="nav navbar-nav">

                    <li><a href="/">Home</a></li>

                    <li><a href="/signup">Sign Up</a></li>

                    <li><a href="/dashboard">Dashboard</a></li>

                </ul>

                <ul class="nav navbar-nav navbar-right">

                    <li><a href="/login">Log In</a></li>

                    <li><a href="/logout">Log Out</a></li>

                </ul>

            </div><!--/.navbar-collapse-->

        </div><!-- /.contaner-fluid-->

    </nav>

    <div class="container row">

        <div class="jumbotron col-sm-4 pull-center">

            <form action="/signup" method="POST">

                <div class="form-group">

                    <label>Username:</label>

                    <input class="form-control" required type="text" name="username"/>

                </div>

                <div class="form-group">

                    <label>Email:</label>

                    <input class="form-control" required type="email" name="email"/>

                </div>

                <div class="form-group">

                    <label>Password:</label>

                    <input class="form-control" required type="password" name="password"/>

                </div>

                <div class="form-group">

                    <input class="btn btn-primary"  type="submit" value="Signup"/>

                </div>

            </form>

        </div>

    </div>

</body>

</html>

* Package.json

{

  "name": "LoginSignupDemo",

  "version": "1.0.0",

  "description": "",

  "main": "index.js",

  "scripts": {

    "start": "nodemon app.js"

  },

  "keywords": [],

  "author": "",

  "license": "ISC",

  "dependencies": {

    "bcrypt": "^5.0.1",

    "body-parser": "^1.19.0",

    "cookie-parser": "^1.4.5",

    "express": "^4.17.1",

    "express-session": "^1.17.2",

    "mongoose": "^5.13.7",

    "nodemon": "^2.0.12"

  }

}

Output:













