**PRACTICAL 12**

# AIM : Building a User Authentication System with Sessions and Cookies without Database (hard-coded username and password)

BACKEND CODE :

* App.js

    const express = require('express');

    const cors=require('cors');

    const session =require('express-session')

    const app = express();

    app.use(express.json())

    app.use(cors({

        origin: 'http://localhost:3000', // your React app's address

        credentials: true,

    }));

    app.use(session({ secret: 'keyboard cat', cookie: { maxAge: 600000 }}))

    app.listen(8000,()=>console.log('connected to port 8000'))

    const [Name,Password]=["om","mypassword"]

    app.post('/login',(req,res)=>{

        const {name,password}=req.body

        console.log(name,password)

        if(name==Name && password==Password){

            req.session.name=name

            res.json("success")

            console.log(req.session.name+"done")

            console.log(req.headers.cookie+"cookie")

            console.log(req.session.id)

        }

        else{

            res.json("please enter correct name and password")

        }

    })

    app.get('/dashboard',(req,res)=>{

        console.log(req.headers.cookie)

        console.log(req.session.id)

        console.log(req.session.name+"dashboard")

        if(req.session.name){

            res.json("welcome to dashboard")

        }

        else{

            res.json("please login first and try again")

        }

    })

    app.get('/logout',(req,res)=>{

        req.session.destroy()

        res.json("deleted")

    })

FRONTEND CODE:

* APP.js

import logo from './logo.svg';

import './App.css';

import './components/styles.css'

import Loginform from './components/Loginform';

import { BrowserRouter,Routes,Route,Link } from 'react-router-dom';

import Dashboard from './components/Dashboard';

function App() {

  return (

  <>

    <BrowserRouter>

    <Link to='/dashboard'>Dashboard</Link>{" "}||{" "}

    <Link to='/'>Login</Link>

    <Routes>

      <Route path='/' element={<Loginform />}/>

      <Route path='/dashboard' element={<Dashboard/>}/>

    </Routes>

    </BrowserRouter>

  </>

  );

}

export default App;

* Dashboard.jsx

import React, { useEffect,useState } from 'react';

import axios from 'axios';

import './styles.css';

import {useNavigate} from "react-router-dom"

function Dashboard() {

    const navigate = useNavigate();

    const [response, setResponse] = useState('');

useEffect(() => {

    axios.get('http://localhost:8000/dashboard', { withCredentials: true })

        .then((response) => {

            setResponse(response.data);

            console.log(response.data + " available");

        })

        .catch((e) => {

            console.log(e.message);

        });

}, []);

    async function logout(){

        const res=await axios.get('http://localhost:8000/logout',{ withCredentials: true })

        console.log("deleted")

        if(res.data=="deleted")

        navigate('/')

    }

    return (  <div className='dashboarddiv'>

        <button onClick={logout}>Logout</button>

    {response}

    </div>

    );

}

export default Dashboard;

* Login.jsx

import React, { useState } from "react";

import axios from "axios";

import './styles.css';

import {useNavigate} from "react-router-dom"

function Loginform() {

  const [name, Setname] = useState("");

  const [password, Setpassword] = useState("");

  const navigate = useNavigate();

  async function handleSubmit(event){

    event.preventDefault();

    const res = await axios.post('http://localhost:8000/login', {name: name, password: password}, {withCredentials: true});

    if(res.data=="success"){

      navigate("/dashboard");

    }

    else{

      console.log(res.data);

    }

}

  return (

    <>

      <div className="logindiv">

        Name :{" "}

        <input

          type="text"

          name="Name"

          onChange={(e) => {

            Setname(e.target.value);

          }}

        />

        Password :{" "}

        <input

          type="text"

          name="Password"

          onChange={(e) => {

            Setpassword(e.target.value);

          }}

        />

        <button type="submit" onClick={handleSubmit}>

          submit

        </button>

      </div>

    </>

  );

}

export default Loginform;

OUTPUT :

   