**Lesson 09 Demo 04**

**react-next-login-app**

**Objective:** To demonstrate the next js to create server side component to do login with file and testing with react component.

**Tools required:** Node JS and React JS

**Prerequisites:** HTML, CSS, JavaScript ES5/ES6, Basic React Concept

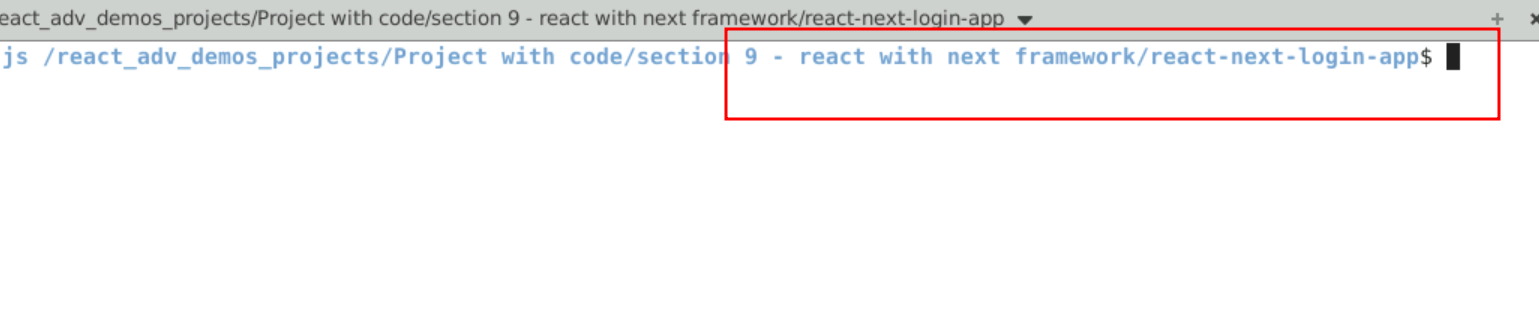
**Note** : All react js project already created with version 18.x with Sample App.js file

**Steps to be followed:**

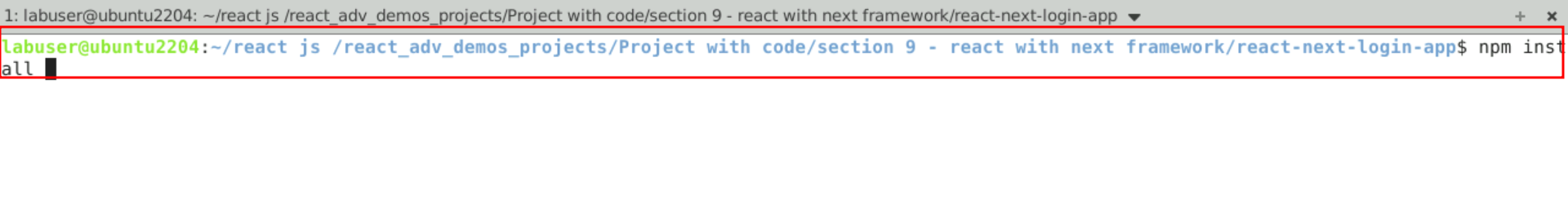
1. Set up for next js project
2. Create login.json file which contains set of login details.
3. Create server folder which contains server side component
4. Create client folder which contains client side component
5. Now we run the application.

**Step 1: Set up for next js project**

1. Open the terminal window inside the **react-next-login-app** folder



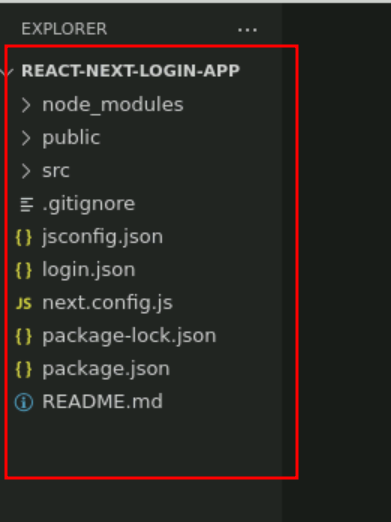
1.2 run npm install to download the dependencies.



* 1. Open the project in VS code

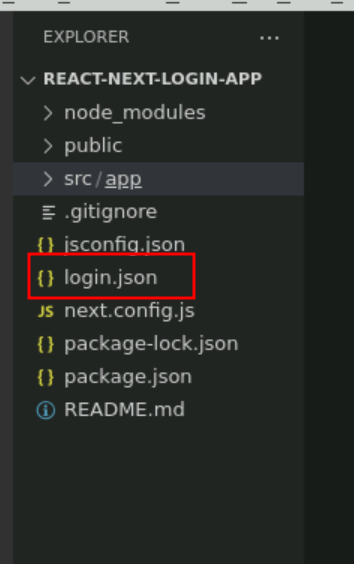
Note: short cut to open write **code .**

* 1. view the project structure



**Step 2 :** Create login.json file which contains set of login details.

2.1 This file contains few static login.json data.

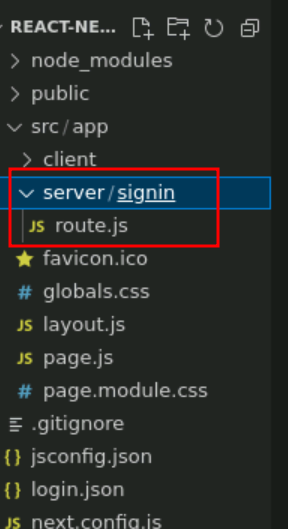
****

2.2 logn.json file

[{"emailid":"steven@gmail.com","password":"steven@123"},{"emailid":"raj@gmail.com","password":"raj@123"}]

**Step 3: create server folder**

3.1 create server folder which contains signin fodler and inside that folder route.js file which is provide POST method to do signup interact with login.json file.



**route.js**

import { NextResponse } from "next/server"

import { promises as fs } from 'fs';

export async function POST(request){

let login = await request.json();

const file = await fs.readFile("login.json","utf-8");

let loginDetails = JSON.parse(file);

let result = loginDetails.find(l=>l.emailid==login.emailid && l.password==login.password);

console.log(result);

if(result!=undefined){

return NextResponse.json({"msg":"Successfully login"})

}else {

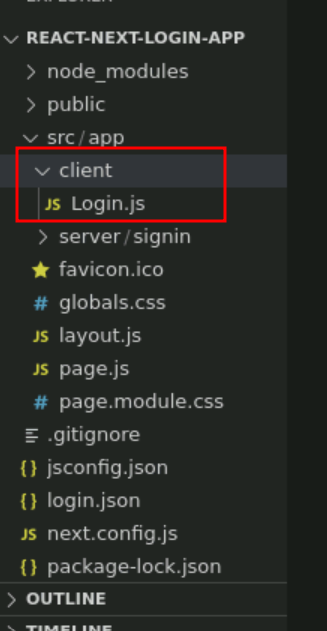
return NextResponse.json({"msg":"In Valid emailid or password"});

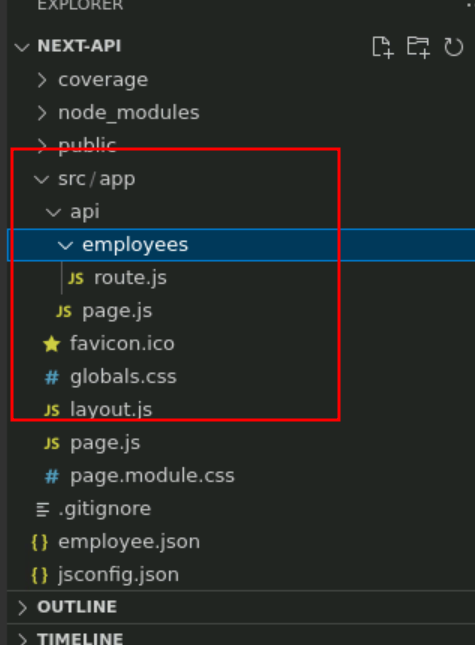
}

}

**Step 4: create client folder**

4.1 create client folder which contains Login.js file component





4.2 This file is responsible to interact with provide login page and interact with server side component with help of axios third party modules.

Login.js

"use client"

import 'bootstrap/dist/css/bootstrap.css'

import { useState } from 'react'

import axios from 'axios';

export default function Login() {

let [emailid,setEmailId]=useState();

let [password,setPassword]=useState();

let [msg,setMsg]=useState("");

const signIn = async (event)=> {

event.preventDefault();

console.log(emailid+" "+password);

let login = {"emailid":emailid,"password":password};

let result = await axios.post(" http://localhost:3000/server/signin",login);

setMsg(result.data.msg);

setEmailId("");

setPassword("");

}

return(<div className='container'>

<form className='form-group' onSubmit={signIn}>

<div className='row offset-4'>

<div className='col-6'>

<h2>Login Page</h2>

</div>

</div>

<div className='row offset-2' >

<div className='col-3'>

<label>EmailId</label>

</div>

<div className='col-3'>

<input type="email" name="emailid" className="form-control"

onChange={(event)=>setEmailId(event.target.value)} value={emailid}

/>

</div>

</div>

<div className='row offset-2' >

<div className='col-3'>

<label>Password</label>

</div>

<div className='col-3'>

<input type="password" name="password" className="form-control"

onChange={(event)=>setPassword(event.target.value)} value={password}

/>

</div>

</div>

<div className='row offset-2' >

<div className='col-3'>

<input type="submit" value="SignIn" className='btn btn-success'/>

</div>

<div className='col-3'>

<input type="reset" value="reset" className='btn btn-info'/>

</div>

</div>

<div className='row offset-4'>

<div className='col-6'>

<span style={{"color":"red"}}>{msg}</span>

</div>

</div>

</form>

</div>)

}

4.3 in Layout.js file provide the client side and server side component details.

Layout.js file

import { Inter } from 'next/font/google'

import './globals.css'

import Login from './client/Login'

const inter = Inter({ subsets: ['latin'] })

export const metadata = {

title: 'Create Next App',

description: 'Generated by create next app',

}

export default function RootLayout({ children }) {

return (

<html lang="en">

<body className={inter.className} suppressHydrationWarning={true}>

<Login></Login>

{children}

</body>

</html>

)

}

Here children props is responsible to provide server side component. Login component is responsible to provide client side components.

3.2 route.js file part of employees folder which contains two method ie Get or post.

Get method is responsible to read the data form employee.json file and produce the data in json format to client. Post method is consume data from frontend technologies and store the data in employee.json file.

route.js

import { NextResponse } from "next/server"

import { promises as fs } from 'fs';

export async function GET(){

const file = await fs.readFile("employee.json","utf-8");

return NextResponse.json(JSON.parse(file));

}

export async function POST(request){

let emp = await request.json();

const file = await fs.readFile("employee.json","utf-8");

let employees = JSON.parse(file);

let result = employees.find(e=>e.id ==emp.id);

console.log(result);

if(result!=undefined){

return NextResponse.json({"msg":"Employee id must be unique"})

}else {

employees.push(emp);

await fs.writeFile("employee.json",JSON.stringify(employees));

return NextResponse.json({"msg":"Record stored successfully"});

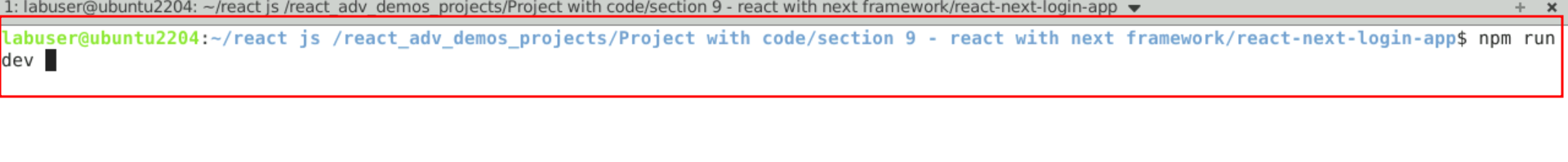
}

}

**Step 5 :Now we run the application.**

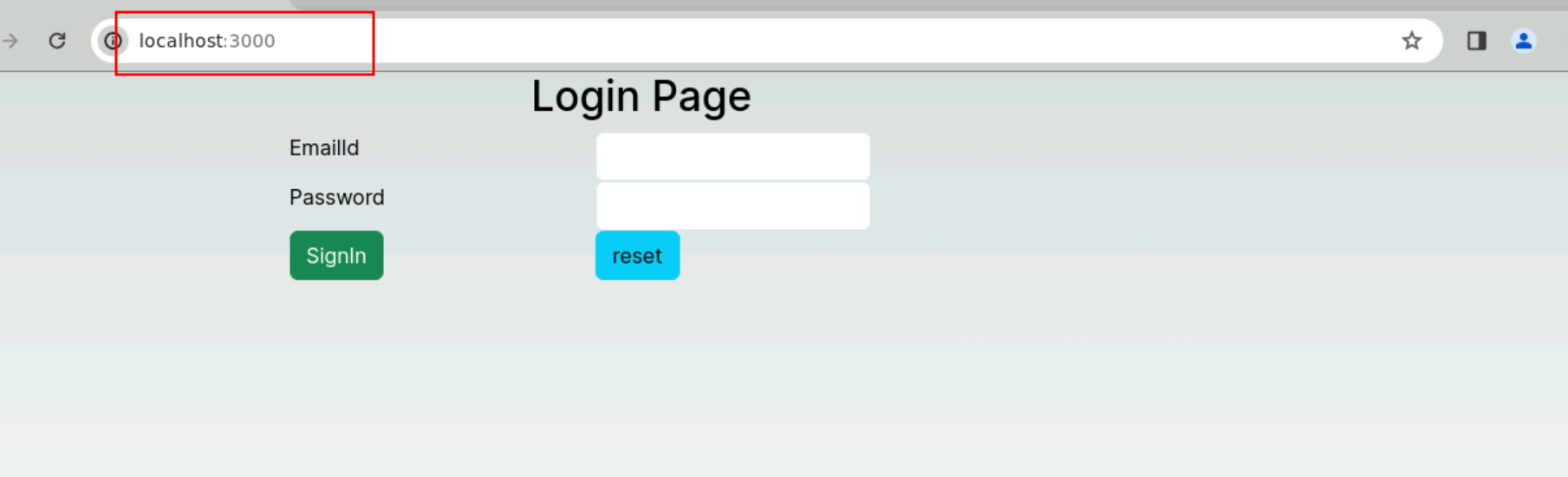
5.1 open the terminal inside the project and run the application we need to use the command as

**npm run dev**

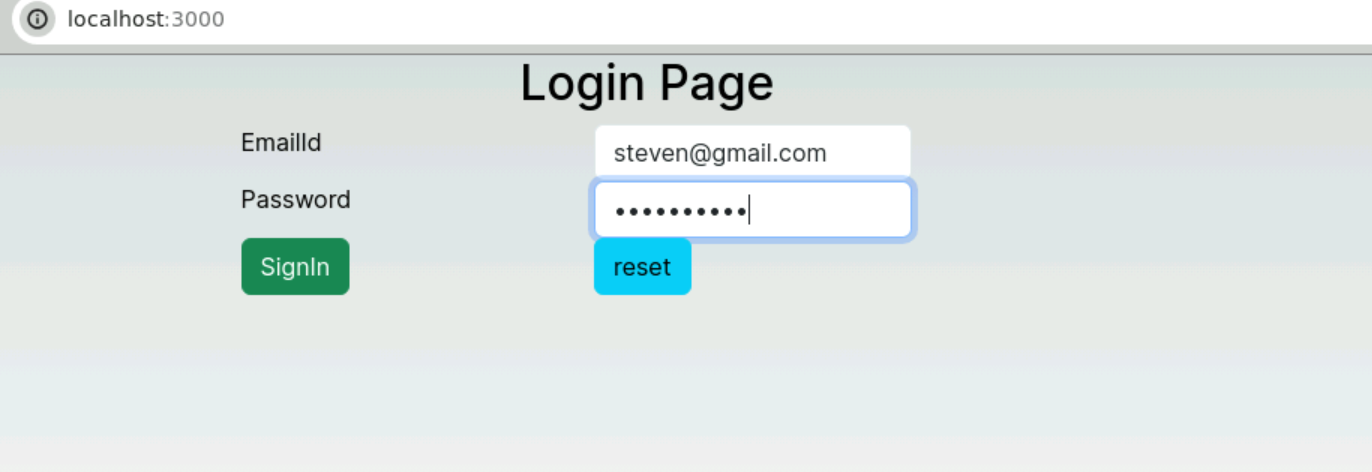
****

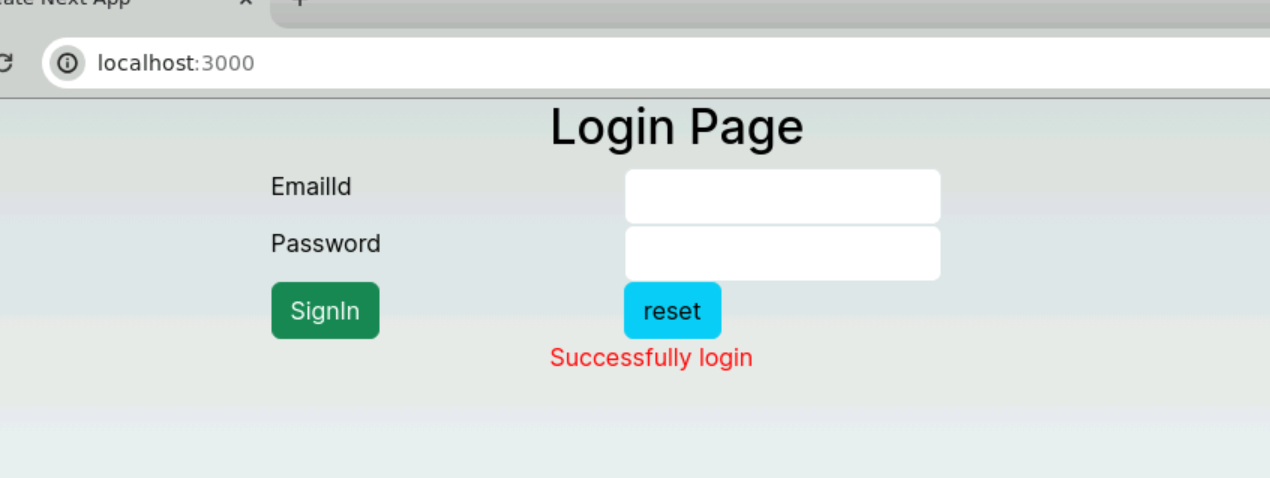
3.2 Now open the browser and write the URL as

<http://localhost:3000>



3.3 Now check the valid emailed and password ie [steven@gmail.com](mailto:steven@gmail.com) and password as **steven@123**





3.4 now check the wrong emailed or password

