Introduction:

React form errors handling in react-hook with useform

APP.JS:

import'./App.css'

import { useForm } from "react-hook-form";

function App(){ const{handleSubmit,register,formState:{errors}}=useForm()

const fun=(data)=>{

alert("successfully login")

}

return(

<div id='app-container'>

<h1><u><center>Form-validation(errors handling)</center></u></h1>

<form onSubmit={handleSubmit(fun)}>

<input type='text' name='us' {...register("us",{required:true,minLength:6})} placeholder=' user name'></input>

<br></br> {errors.us && errors.us.type==="required" &&<font color="red">enter your username</font>}

<br></br> {errors.us && errors.us.type==="minLength" &&<font color="red">invalid username</font>}

<br></br>

<input type='password' name='pw' {...register("pw",{required:true,

pattern:/^(?=.\*\d)(?=.\*[a-z])(?=.\*[A-Z]).{5,12}$/})} placeholder=' password'></input>

<br></br>{errors.pw && errors.pw.type==="required" &&<font color="red">enter your password</font>}

<br></br>{errors.pw && errors.pw.type==="pattern" &&<font color="red">invalid password</font>}

<br></br><input type='number' name='ph' {...register("ph",{required:true,minLength:10})}placeholder=' phone number'></input>

<br></br>{errors.ph && errors.ph.type==="required" &&<font color="red">enter your phone number</font>}

<br></br>{errors.ph && errors.ph.type==="minLength" &&<font color="red">incorrect phone number</font>}

<br></br><input type='email'name='em' {...register("em",{required:true,minLength:12})}placeholder=' email' ></input>

<br></br>{errors.em && errors.em.type==="required"&&<font color="red">invalid email</font>}

<br></br>{errors.em && errors.em.type==="minLength"&&<font color="red">invalid email</font>}

<br></br> <input type='submit' id='bt'></input>

</form>

</div>

)

}export default App

APP.CSS:

#app-container{

  box-sizing: border-box;

  width: 700px;

  height: 500px;

   justify-items: center;

   align-items: center;

   justify-content: center;

   text-align: center;

   position: absolute;

   top: 10%;

   left: 30%;

   border: 2px solid salmon;

   border-radius: 10px;

   background-color:springgreen;

}

#bt{

  background-color: blue;

  width: 100px;

  border-radius: 30px;

}

#bt:hover{

  width: 150px;

  font-weight: bold;

  background-color: red;

  border-radius: 75px

}

input{

  width: 250px;

  height: 20px;

  border-radius: 30px;

  border-color: 2px solid chartreuse;

}

Output:

