import React from "react";

import { Formik, Field, ErrorMessage, Form } from "formik";

import \* as Yup from "yup";

import "bootstrap/dist/css/bootstrap.css";

const LoginSchema = Yup.object().shape({

email: Yup.string()

.email("Invalid email address format")

.required("Email is required"),

password: Yup.string()

.min(3, "Password must be 3 characters at minimum")

.required("Password is required"),

});

class App extends React.Component {

render() {

return (

<div className="container">

<div className="row">

<div className="col-lg-12">

<Formik

initialValues={{

email: "",

password: "",

}}

validationSchema={LoginSchema}

onSubmit={(

values,

{ setSubmitting }

) => {

console.log(values);

// Simulating asynchronous operation, like an API call

setTimeout(() => {

alert(

"Form is validated! Submitting the form..."

);

setSubmitting(false);

}, 1000);

}}

>

{(props) => (

<div>

<div className="row mb-5">

<div className="col-lg-12 text-center">

{!props.isSubmitting ? (

<div>

<h1>

Login

Page

</h1>{" "}

<Form>

<div className="form-group">

<label htmlFor="email">

Email

</label>

<Field

type="email"

name="email"

placeholder="Enter email"

autoComplete="off"

className={`mt-2 form-control ${

props

.touched

.email &&

props

.errors

.email

? "is-invalid"

: ""

}`}

/>

<ErrorMessage

component="div"

name="email"

className="invalid-feedback"

/>

</div>

<div className="form-group">

<label

htmlFor="password"

className="mt-3"

>

Password

</label>

<Field

type="password"

name="password"

placeholder="Enter password"

className={`form-control ${

props

.touched

.password &&

props

.errors

.password

? "is-invalid"

: ""

}`}

/>

<ErrorMessage

component="div"

name="password"

className="invalid-feedback"

/>

</div>

<button

type="submit"

className="btn btn-primary btn-block mt-4"

disabled={

props.isSubmitting

}

>

{props.isSubmitting

? "Submitting..."

: "Submit"}

</button>

</Form>

</div>

) : (

<h1>

Home

Page

</h1>

)}

</div>

</div>

</div>

)}

</Formik>

</div>

</div>

</div>

);

}

}

export default App;