

Exercise 1: Basic Form

Create a simple form with React Hook Form. Add basic validation for required fields and display error messages.

Exercise 2: Custom Validation

Implement custom validation rules using the register method. For example, validate that a password contains at least eight characters and includes both letters and numbers.

Exercise 3: Conditional Validation

Add conditional validation to the form. Create a checkbox that, when checked, requires the user to enter additional information.

Exercise 4: Field Arrays

Build a form that allows the user to add and remove multiple fields dynamically, such as a list of email addresses.

Exercise 5: Integration with UI libraries

Integrate React Hook Form with a UI library like Material-UI. Create a form using their components and apply validation rules.

Exercise 6: Async Validation

Implement asynchronous validation using custom async functions. For instance, validate if an email address is already taken by making an API request.

Exercise 7: Form Submission

Handle form submission using React Hook Form. Add a submit button and display a success message or error message based on the submission result.

Exercise 8: Form Reset and Pre-filling

Implement form reset functionality and pre-fill form fields using the reset and setValue methods respectively.

Exercise 9: Field Validation Dependencies

Create a form where the validation of one field depends on the value of another field. For example, validate that the "confirm password" field matches the "password" field.

Exercise 10: Form Validation with Yup

Integrate React Hook Form with the Yup validation library. Define validation schemas using Yup and use them with React Hook Form to validate your form.