

플랫폼 사업본부 웹 개발팀 박채연 연구원

Form Validation Seminar

React-hook-form / Yup



Overview: 2.0 Form Validation Library(React-hook-form)

1. 세미나 개요

2. Form Validation Library

3. React-hook-form 의 강점

3.1 React-hook-form vs Formik 실습

3.2 Controlled/Uncontrolled

4. React-hook-form

Overview

4 .1 React-hook-form 사용 방법 [실습]



Form Validation

React-hook-form 의

React-hook-form vs

Controlled

React-hook-form

React-hook-form

1. Form Validation 의 향후 사용 방안을 정립 및 내용 공유

2. 상단의 Form 공용 컴포넌트를 제작

React Hook Form

99

✕

Full Name

Vatech.Clever

Email Address

vatech.clever@vatech.com|

Age

23

Start Date

12/12/2022

📅

End Date

12/12/2022

📅

Password

Vatechvictory001#

👁

Confirm Password

Vatechvictory

👁

Incorrect entry.

Normal

✎

🔗

🔗

🔗

🔗

T

Y

🔄

⏮

⏪

⏩

⏭

✎

🔍

Tt

≡

🖼

↶

📄

🔄

Cras mattis consectetur purus sit amet fermentum. Cras justo odio, dapibus ac facilisis in, egestas eget quam. Morbi leo risus, porta ac consectetur ac, vestibulum at eros.

📎 Upload Photo

Terms and Conditions ✕

7.9MB

☐ Terms and Conditions

Must Accept Terms and Conditions.

Submit



Form Validation Library

세미나 개요

Form Validation
Library

React-hook-form 의
강점

React-hook-form vs
Formik 실습

Controlled
/Uncontrolled

React-hook-form
Overview

React-hook-form
사용 방법 [실습]

Form Validation Library

A. Form Validation

P-1. 서버까지 데이터가 다녀온 후 에야 잘못된 데이터임을 파악 가능

**FV: 서버에 데이터를 제출하기 전에
양식이 올바른지 확인해준다.**

P2. 사용자가 의도적으로 이상한 데이터를 넣어 공격

**FV: 사용자가 데이터를 목적지로 보내기 전에 올바른 데이터를 입력했는
지 유효성을 체크해준다.**

B. Library

P-1. 폼을 다루기 위해선 많은 보일러 플레이트 코드 필요

- 유저 입력 값의 상태 관리 및 검증
- 유효하지 않은 에러 메시지 추적
 - 폼 제출 다루기

P-2. Form 필드가 많아지면 코드양 多

Library : 양식의 유효성 검사할 뿐만 아니라 모든 오류에 대한
요약을 만들어 사용자에게 보여준다.



React-hook-form 의 강점 (vs Formik)

세미나 개요

Form Validation
Library

React-hook-form 의
강점

React-hook-form vs

Formik 실습

Controlled
/Uncontrolled

React-hook-form

Overview

React-hook-form
사용 방법 [실습]

| | react-hook-form | formik |
|------------------------|-----------------|-----------|
| Weekly Downloads | 7xx,xxx | 1,4xx,xxx |
| Re-render 효율 | Win | Lose |
| Size | 9.2kb | 13.1kb |
| Typescript | ✓ | ✓ |
| Class components | ✗ | ✓ |
| 부분 Watch | ✓ | ✗ |
| Hook API | ✓ | ✓ |
| Error, Validataion | ✓ | ✓ |
| Yup, Schema | ✓ | ✓ |
| Controlled Component | ✓ | ✓ |
| UnControlled Component | ✓ | ✗ |

Re-rendering 효율성

React-hook-form : 불필요한
재 렌더링을 제거하면서, 작성
해야 하는 코드양을 줄입니다.

빠른 Mount 속도

React-hook-form는 Formik 보
다 mount되는 속도가
빠릅니다.

Uncontrolled

React-hook-form : Controlled
& Uncontrolled,
Formik : Controlled

부분 Watch

React-Hook-Form: 특정 state
만 업데이트 가능
Formik : 전체 업데이트



React-hook-form 의 강점 (vs Formik) : 실습을 통해 확인

세미나 개요

Form Validation

Library

React-hook-form 의

강점

React-hook-form vs

Formik 실습

Controlled

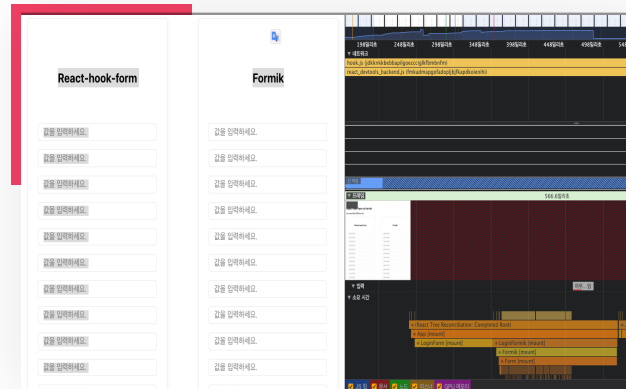
/Uncontrolled

React-hook-form

Overview

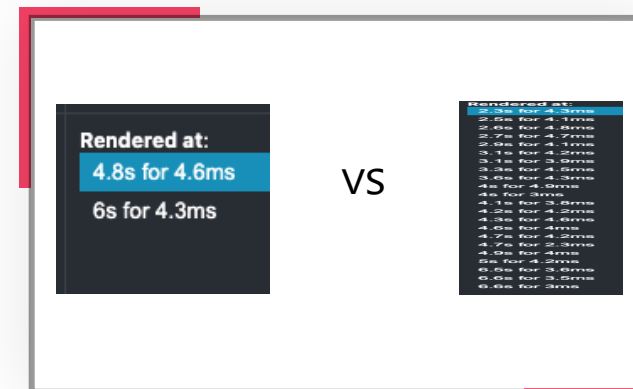
React-hook-form

사용 방법 [실습]



빠른 Mount 속도

React-hook-form은 Formik 보다 mount 완료 속도가 빠릅니다.



Re-rendering 효율성

React-hook-form 은 불필요한 재 렌더링을 제거하면서, 작성해야 하는 코드양을 줄입니다.



세미나 개요

Form Validation

Library

React-hook-form 의

강점

React-hook-form vs

Formik 실습

Controlled

/Uncontrolled

React-hook-form

Overview

React-hook-form

사용 방법 [실습]

Controlled vs Uncontrolled

React

React Component에서 html element 사용?

내부 상태를 신뢰 가능한 단일 소스로 관리하려는 설계 원칙 위배 [Single Source of Truth]

HTML element (<form> <input> <textarea> <select>) 직접 입력 정보 보관 &

React 컴포넌트도 직접 정보 보관

| | Uncontrolled | Controlled |
|-----------------|---|---|
| Concept | 'PULL' the Value from HTML element | 'PUSH' the Value to HTML element |
| Description | React component 가 html element에서 정보를 PULL 하여 사용하는 방식 | component의 state를 HTML element로 PUSH |
| 데이터 정보 저장의 주체 | HTML element | React Component |
| 리랜더링 | 리랜더링 발생하지 않음. (onSubmit 시) | 리랜더링 발생 (onChange 시) |
| 실시간 관리 | 실시간 관리 X | 실시간 관리 O |
| React-hook-form | Input 태그 내부에 부여된 ref를 통해서 받아온 정보에 대한 참조 저장 가능 | 부분 Watch 가능 <부분 Controlled> |
| Formik | 불가능 | 특정 state만 watch를 할 수 없고 전체를 업데이트 |



세미나 개요

Form Validation

Library

React-hook-form 의

강점

React-hook-form vs

Formik 실습

Controlled

/Uncontrolled

React-hook-form

Overview

React-hook-form

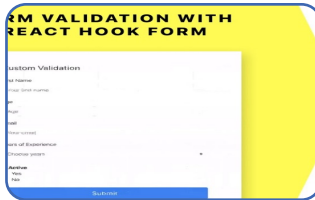
사용 방법 [실습]

React-hook-form Overview



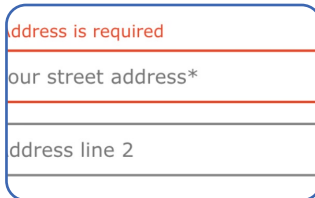
I. Register Fields

구성 요소를 Hook에 등록



II. Apply Validation

(required/min/max/minLength/maxLength/pattern/validate)



III. Handle Errors

React Hook Form 은 유효성 검사 제약 조건에 따라 오류 개체를 제공하여 Form 의 Errors를 표시



IV. Schema Validation

Yup (JavaScript 객체 유효성 검사기) Schema를 Use Form API 에 resolver로 넘겨줄 수 있다.



React-hook-form API Description

세미나 개요

Form Validation

Library

React-hook-form 의

강점

React-hook-form vs

Formik 실습

Controlled

/Uncontrolled

React-hook-form

Overview

React-hook-form

사용 방법 [실습]

API

React Hook Form's API overview

`</> useForm`

A powerful custom hook to validate your form with minimal re-renders.

[Read More ▶](#)

`</> useController`

For Controlled components: interface with the useForm methods and isolate its re-render.

[Read More ▶](#)

`</> useFormContext`

Access your useForm methods and properties from nested components. Great for building larger forms and shared components!

[Read More ▶](#)

`</> useWatch`

Subscribe to individual form input changes without impacting the root component's render.

[Read More ▶](#)

`</> useFormState`

Subscribe to individual form state updates and isolating re-renders at the hook level.

[Read More ▶](#)

`</> useFieldArray`

Manage dynamically generated fields on the fly, shuffle, remove and append fields. Ideal for complex CRUD data entry scenarios.

[Read More ▶](#)