Because when re-rendering occurs, these keys will be different, resulting in React having to recreate your list from scratch.

② Item Index : Indexes are not recommended for keys if the order of items may change, for eg in cases where your list has sorting capabilities or users can either add or remove items.

Why?
Negatively affect performance
User interface glitches.

③ Unique ID : e.g. UUID()

Day-2

✱ Forms in React

✱ What are controlled Components?

→ Controlled components are a set of components that offer a declarative API (Application Programming Interface) to enable full control of the state of form elements at any point in time using React state.

→ How?

Using value prop : → state delegation is performed via value prop.

Value → A special property to determine input content at any time during render life cycle.

→ How to create a controlled component?

(i) Local State (useState)
(2) Value prop
(3) onChange callback

onChange callback → Receives an event parameter, which is an event object representing the action that just took place similar to events on DOM elements.

To get the new value on every keystroke, we need to access the target property from the event and grab the value from that object, which is a string.

```
handleChange (event) {
    setValue(event.target.value)
}
```

→ When form is submitted ? (What to do)
How

```
< form onSubmit={handleSubmit} >
   :
   :
</form>

handleSubmit (event) {
    validate (value) ⟶ validating input values
    event.preventDefault()
}
```

To have control over the form values when form is submitted, use onSubmit prop in <form> tag

onSubmit callback receives event parameter