zh-CN

`Form` 具有自动收集数据并校验的功能，但如果您不需要这个功能，或者默认的行为无法满足业务需求，可以选择自行处理数据。

en-US

`Form` will collect and validate form data automatically. But if you don't need this feature or the default behavior cannot satisfy your business, you can handle form data manually.

```tsx
import React, { useState } from 'react';
import { Form, InputNumber } from 'antd';

type ValidateStatus = Parameters<typeof Form.Item>[0]['validateStatus'];

function validatePrimeNumber(number: number): {
  validateStatus: ValidateStatus;
  errorMsg: string | null;
} {
  if (number === 11) {
    return {
      validateStatus: 'success',
      errorMsg: null,
    };
  }
  return {
    validateStatus: 'error',
    errorMsg: 'The prime between 8 and 12 is 11!',
  };
}

const formItemLayout = {
  labelCol: { span: 7 },
  wrapperCol: { span: 12 },
};

const RawForm = () => {
  const [number, setNumber] = useState<{
    value: number;
    validateStatus?: ValidateStatus;
    errorMsg?: string | null;
  }>({
    value: 11,
  });

  const tips =
    'A prime is a natural number greater than 1 that has no positive divisors other than 1 and itself.';

  const onNumberChange = (value: number) => {
    setNumber({
```

```jsx
      ...validatePrimeNumber(value),
      value,
    });
  };

  return (
    <Form>
      <Form.Item
        {...formItemLayout}
        label="Prime between 8 & 12"
        validateStatus={number.validateStatus}
        help={number.errorMsg || tips}
      >
        <InputNumber min={8} max={12} value={number.value} onChange={onNumberChange} />
      </Form.Item>
    </Form>
  );
};

export default () => <RawForm />;
```